Mentions of Security Vulnerabilities on Reddit, Twitter and GitHub

Sameera Horawalavithana
CSE, University of South Florida
sameera1@mail.usf.edu

Abhishek Bhattacharjee
CSE, University of South Florida
abhishekb1@mail.usf.edu

Renhao Liu
CSE, University of South Florida
renhaoliu@mail.usf.edu

Nazim Choudhury
CSE, University of South Florida
nachoudhury@mail.usf.edu

Lawrence O. Hall
CSE, University of South Florida
lohall@mail.usf.edu

Adriana Iamnitchi
CSE, University of South Florida
anda@cse.usf.edu

ABSTRACT

Activity on social media is seen as a relevant sensor for different aspects of the society. In a heavily digitized society, security vulnerabilities pose a significant threat that is publicly discussed on social media. This study presents a comparison of user-generated content related to security vulnerabilities on three digital platforms: two social media conversation channels (Reddit and Twitter) and a collaborative software development platform (GitHub). Our data analysis shows that while more security vulnerabilities are discussed on Twitter, relevant conversations go viral earlier on Reddit. We show that the two social media platforms can be used to accurately predict activity on GitHub.

CCS CONCEPTS

• **Networks → Online social networks**; • **Theory of computation → Social networks**.

ACM Reference Format:
Sameera Horawalavithana, Abhishek Bhattacharjee, Renhao Liu, Nazim Choudhury, Lawrence O. Hall, and Adriana Iamnitchi. 2019. Mentions of Security Vulnerabilities on Reddit, Twitter and GitHub. In *IEEE/WIC/ACM International Conference on Web Intelligence (WI '19), October 14–17, 2019, Thessaloniki, Greece.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3350546.3352519

1 INTRODUCTION

The volume and variety of cyber attack strategies are continuously growing. In 2018, the number of distributed denial of service attacks increased by 500% and affected 75% of the health care industry [15]. According to [7], a hacker attack takes place every 39 seconds.

Cybersecurity attacks are partially enabled by the fact that many applications nowadays share open-source libraries, and thus a vulnerability in one of these shared libraries can have wide-spread impact. When an open-source vulnerability is discovered, security experts tend to announce and publicly discuss it on a variety of platforms, including forums, blogs, and social media [13, 24]. Sapienza et al. [21] discovered that attackers explore public information channels (such as security web blogs and social media) along with more private discussion threads (such as email chains or the DarkWeb) to discover security vulnerabilities and identify potential attack targets. Thus, complementary information is publicly available on multiple social platforms and such information describes software vulnerabilities and ways to exploit them.

While different studies have quantified the mentions of software vulnerabilities on individual platforms [16, 19], it is little understood how this information is present on multiple social platforms. Do some platforms post more information and faster than others or are they equal contributors to public knowledge? Moreover, how does the information related to software vulnerabilities affect the related software development activities? The answer to such questions may inform the design of security alert tools based on information from multiple social media platforms.

This paper provides a quantitative analysis of software vulnerability information on three different social media platforms to address questions such as the ones listed above. We chose three social media platforms for this study: Twitter, due to its ubiquity; Reddit, due to its focused topic-based conversations structured around subreddits; and GitHub, a popular collaborative open-source software development platform. We analyzed the security vulnerability mentions in these three online data sources to answer the following two research questions: *What is the relationship between mentions of security vulnerabilities as posted on Twitter, Reddit and GitHub? (RQ1)*; and *Can the software development activities in GitHub be predicted from the discussions on Reddit and Twitter? (RQ2).*

Our analysis shows that, while more vulnerabilities are mentioned on Twitter, more engaging conversations that mention security vulnerabilities take place on Reddit. Moreover, most of the Reddit vulnerability conversations end before the time of its public disclosure, while on Twitter information cascades tend to form only after public disclosure. In addition, we developed a prediction model that is able to accurately predict the patterns of popularity and engagement level activities in GitHub using activity information extracted from Reddit and Twitter.

2 RELATED WORK

Software vulnerabilities and potential security attacks are commonly discussed on various social media platforms, such as Twitter, security blogs, and even the dark web [13, 20, 21, 24]. Sauerwein et al. [22] showed that security vulnerabilities are discussed on Twitter even before they are officially announced. Syed et al. [24]

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
WI '19, October 14–17, 2019, Thessaloniki, Greece
© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-6934-3/19/10…$15.00
https://doi.org/10.1145/3350546.3352519

characterized the software vulnerability tweet messages which drive retweeting of these tweets.

Many studies looked at the potential of mining these sources of data—tweets [11], blogs, and other online texts [1]—for automatically discovering security threats [6, 23]. For example, Sapienza et al. [20] developed tools to detect security threats by using discussions on Twitter and the dark web. Mittal et al. [13] analyzed tweets to detect cybersecurity vulnerabilities and alert users. Nunes et al. [16] used a combination of supervised and semi-supervised machine learning approaches on data from sites in darknet and deepnet to detect cybersecurity threats. Sabottke et al. [19] used Twitter and other complementary data sources for exploit detection using supervised machine learning techniques. Other studies [21] used Twitter and other data sources, such as security related blogs and the dark web, to implement a threat detection mechanism which could distinguish between various classes of security violations, such as data breaches, malware, and DDoS attacks. Mulwad et al. [14] proposed a framework for vulnerabilities, threat, or attack detection by passing texts from the web through a SVM classifier and verified the results using databases like Wikitology.

In this paper, we quantitatively compare the volume and the timing of discussions related to security vulnerabilities on two social media platforms, Twitter and Reddit. Moreover, we show that information from these two social media platforms can be used to predict the activity on various open source software repositories in GitHub that relate to these vulnerabilities or are potentially affected by them. Such predictions can have multiple practical applications: for example, the popularity of a vulnerable software repository (measured/predicted by GitHub Fork and Watch activities) tracks the spread of underlying exploits.

3 DATASETS

For our investigations we used three types of datasets related to software vulnerabilities and their mention in social media over a period of 1.5 years. These datasets were privately released under the DARPA SocialSim project for its December 2018 Challenge.

First, we selected the subset of software vulnerabilities described by their Common Vulnerability Exposure (CVE) identifiers as recorded in the *National Vulnerability Database (NVD)* between January 2015 and May 2018. Second, we filtered posts from two social media platforms (Reddit and Twitter) made between March 2016 and August 2017, and considered only those conversations that included at least one mention of a vulnerability identifier (i.e., CVE) from our NVD subset. Third, we selected a subset of software repositories in GitHub that are related to the CVEs in our NVD dataset and considered all user interactions with those repositories between January and August 2017. These datasets are presented below. The longer period for the NVD dataset allows us to analyze the timing of conversations on Reddit and Twitter in comparison with the official time of vulnerability disclosures, as recorded in NVD.

3.1 The National Vulnerability Database

Common Vulnerability Exposure (CVE) identifiers are standard identifiers for security vulnerabilities, heavily used in related public postings on the web [12]. The assignment of CVE identifiers is usually done by a numbering authority (e.g., MITRE[1]) once a vulnerability is discovered. The National Vulnerability Database (NVD) serves as the standard body to accept CVE identifiers for publicly declared security vulnerabilities.

The life-cycle of a software vulnerability, as presented in [2, 8], is shown in Figure 1. The period between the first discovery and the public disclosure of the vulnerability is defined as the *Black Risk Phase* [22]. During this time, the vulnerability is known by a small community, it can be discussed on public and private forums, and its exploitation risk is the highest. The day of the public disclosure is often referred to as Day 0. Once NVD accepts the vulnerability, the *Grey Risk Phase* follows until the vulnerability is resolved with an official patch release. Thus, the *grey risk phase* ends with the availability of official countermeasures, while the *white risk phase* covers the time needed for deploying the countermeasures (patch).

Attackers often start exploiting vulnerable software products during the *Black risk phase*, and thus early awareness (i.e., before the official disclosure time) is vital for protection. This is the reason why we focus some of the analysis in this study on the timing of actions before and after the official public disclosure.

The NVD fields available for our study are the published date of the particular CVE identifier, the score (0-10) and the class (i.e., critical, high, medium, low) of attack severity according to *Common Vulnerability Scoring System v3.0*[2].

Figure 1: The life-cycle of a vulnerability.

3.2 Social Media Datasets

Reddit is a popular website where users engage with a bulletin board system by posting content, commenting on each other's posts, and voting messages up or down. Content is organized into topic-specific subreddits. Users can post content, comment, or vote once they are logged into their account. Users can also befriend each other (similar to an online social network), in which case they get updates on their friends' actions on Reddit. Users can also subscribe to subreddits to personalize what content they see. Many subreddits are on technical topics, such as */r/linux*, */r/security*, */r/hacking*, and */r/sysadmin*.

Twitter is a micro-blogging platform where users broadcast messages (i.e., tweets) publicly or share privately to their follower network. Twitter allows tweets to be tagged with hashtags, and users can post messages under one or multiple hashtags.

Data Processing. Our social media data from these two platforms consists of all Reddit conversations and tweets between March 2016 and August 2017 that include at least one CVE identifier from the NVD dataset described above. We used the regular expression pattern of `CVE-\d{4}-\d{4}\d*` to match CVE IDs that appeared in posts, comments and tweets.

[1] https://cve.mitre.org/cve/
[2] https://www.first.org/cvss/specification-document

We included the complete conversation thread when a CVE identifier appears in a post or a comment. We included all tweets, retweets, and replies when a CVE identifier was found in the original tweet. Table 1 presents the size of the social media dataset.

Table 1: Size of dataset. Activities represent posts (18%) and comments (82%) in Reddit, tweets (76%), re-tweets (19%) and replies (5%) in Twitter, and events (push, issue, pull-request, watch, fork) in GitHub. Communities are represented by subreddits in Reddit, hashtags in Twitter, and software repositories in GitHub.

Dataset	Activities	Users	Communities
Reddit	170,486	65,059	384
Twitter	105,596	8,766	1,685
Github	7,240,398	567,925	13,837

In addition, we augmented these datasets as follows:

- *Re-tweet cascades*: We used publicly available code [17] to reconstruct the re-tweet cascades given a set of original tweets in Twitter. This code implements the re-tweet construction algorithm proposed in [25]. The algorithm uses the Twitter's follower graph to construct the retweet cascade. We used the Twitter follower API to get the list of followers for all users who appeared in our Twitter dataset.
- *Sentiment analysis*: We used multiple measures of sentiment as follows. For *sentiment scores (polarity and subjectivity)*, we used VADER [10], a Python library of a natural language toolkit. The polarity sentiment score reflects the level of controversy in the text, while the subjectivity sentiment score classifies the text by the subjective/objective opinion of the author. For *Reddit uncertainty analysis*, we inferred the uncertainty of a Reddit comment towards the subject in the conversation based on a trained machine learned model presented in [26]. Note that this model is the best-performing uncertainty detector in the literature with a reported F1-score of 65.06% on standard benchmarks. For identifying *Twitter reaction types*, we used the trained machine learned model presented in [9]. Based on replies to tweets, we inferred the type of reply as one of the following six types of discourse: answer, elaboration, question, appreciation, negative reaction, agreement. From each tweet/reply message, we extracted Linguistic Inquiry and Word Count (LIWC) features [9] to feed into the model.
- *Bot detection*: We used BotHunter [5] to detect bot accounts in Twitter. We manually identified subreddits in Reddit that are driven by a bot using the textual description that appears in subreddits. For example, `CVEWatch` [3] is a bot-driven subreddit that automatically posts new CVE description from NVD. Because we are interested in the reaction to human-curated messages, we eliminated bot-driven messages from our datasets in many instances of our analysis.

[3]https://www.reddit.com/r/CVEWatch

3.3 GitHub Dataset

Github is an open-source software collaboration platform where users contribute to Github repositories via code commits, pushes, pull-requests, and raising issues. Users can also watch repositories to receive alerts on updates, and can fork (i.e., copy) public repositories to make their own software modifications.

GitHub is home to over 100 million public repositories and 30 million users. The dataset of GitHub events on the public repositories is publicly available [3]. In this study we focus on a subset of repositories that have one of the CVE identifiers from our NVD dataset in the repository description or are the target of a user action that includes a CVE identifier in the text description associated with the action (such as a commit message, for example). We use the same regular expression `CVE-\d{4}-\d{4}\d*` to match CVE identifiers that appeared in Github event descriptions. We restricted the time window for the event description selection to January 2017–August 2017. This interval has a significant overlap with both the NVD and the social media dataset (47% and 3% overlap with CVE ids observed in Twitter, and Reddit respectively), and is also manageable in size. During this interval, over half a million users contributed to over 13 thousand software repositories via more than 7 million actions (Table 1).

4 CVE MENTIONS ON PUBLIC INFORMATION CHANNELS

Previous work [20] suggested that messages shared in public social media channels can be used as early signals to detect security vulnerabilities. Syed et al. [24] found that such messages in Twitter usually contain vulnerability alerts, advisory messages, patch information, exploit details, and root-causes. In this study, we analyze the reaction of social-media platforms after perceiving security vulnerability messages. We show how such information disseminates across two social-media platforms, Reddit and Twitter, and present the software development activities in related GitHub repositories. These activities usually represent a collective response from the public or security experts to flaws in software repositories. Our empirical investigation shows how multiple social-media communities respond to security vulnerabilities.

This section answers two questions related to (*RQ1*). First, *how do social media platforms compare in terms of these signals on security vulnerabilities?* Second, *to what extent are named vulnerabilities discussed on public channels before the official disclosure day?* To answer these questions, we compare the volume of discussions on Reddit and Twitter before and after the public disclosure and we analyze the GitHub activity with respect to CVE mentions.

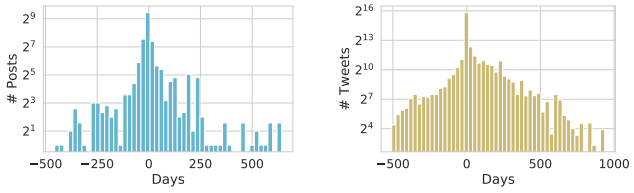
4.1 Social Conversations

We first characterize social-media platforms based on the appearance of CVE identifiers. We analyze the number of distinct CVE identifiers that are discussed only on Twitter, only on Reddit and, respectively, on both platforms. Out of 10,257 CVE IDs from our social-media dataset, the vast majority (95%) are mentioned only on Twitter. 0.5% CVE IDs are mentioned only on Reddit, with the remaining 4.5% mentioned on both platforms.

The timing relative to the public disclosure is paramount for describing early signals. There are 10,209 CVE identifiers discussed

on Twitter, and 17% are mentioned before their public disclosure. Within the same period, of the 460 CVE identifiers mentioned in Reddit posts, 51% were mentioned before their public disclosure. Of the 412 CVE identifiers that appeared on both platforms, 225 (54%) were first discussed on Twitter, while the rest of 46% appeared on Reddit before Twitter.

Figure 2 shows the daily volume of posts and tweets as related to the Day 0 (public disclosure day). We discovered both Reddit and Twitter have mentions of CVEs more than a year before their public disclosure dates. As an example, mentions of `CVE-2016-0898` occur on both platforms 456 days before its NVD public disclosure. Another observation from Figure 2 is the peak in CVE mentions during Day 0 on both platforms. This behavior is confirmed by a previous study [22] on Twitter, where the same phenomenon was noticed on data from a different period.



(a) Reddit (b) Twitter

Figure 2: The early time advantage to receive alerts on application vulnerabilities in Reddit and Twitter. Bot-driven messages were removed from both platforms.

Tables 2 and 3 present the number of discussions before the public disclosure for the two platforms, classified by topics as suggested by subreddits for Reddit and by hashtags for Twitter. Remarkably, the majority of posts that refer to CVEs in Reddit happen before the public disclosure, while the majority of tweets that include CVEs are after the public disclosure.

Table 2: Top 10 subreddits by the total number of posts published during 2016/03–2017/08 that mention CVEs. The proportion of posts that mention CVEs before their public disclosure is presented in parentheses.

Subreddit	Number of CVE Posts	
	Total	Before public disclosure
CVEWatch	1,884	0 (0%)
netsec	304	174 (57%)
fulldisclosure	149	110 (74%)
phpAdvisories	117	35 (30%)
linux	89	71 (80%)
security	66	41 (62%)
hacking	59	32 (54%)
sysadmin	50	31 (62%)
InfoSecNews	33	8 (24%)
Android	25	14 (56%)

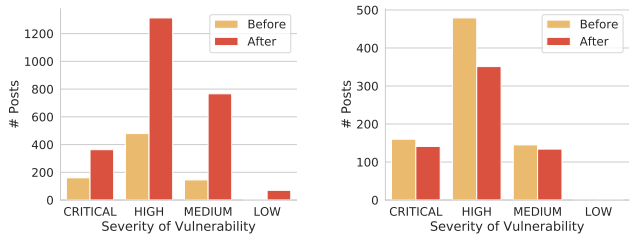
Table 3: Top 10 hashtags by the total number of tweets published during 2016/03–2017/08 that mention CVEs. The proportion of tweets that mention CVEs before their public disclosure is presented in parentheses.

Hashtag	Number of CVE Tweets	
	Total	Before public disclosure
#security	38,977	71 (0.1%)
#infosec	29,762	4,883 (16%)
#cybersecurity	26,681	8,610 (32%)
#cve	11,728	162 (1%)
#securityaffairs	2,298	1,051 (46%)
#vuln	2,026	492 (24%)
#0daytoday	1,525	328 (22%)
#linux	986	274 (28%)
#0day	856	503 (59%)
#exploit	656	398 (60%)

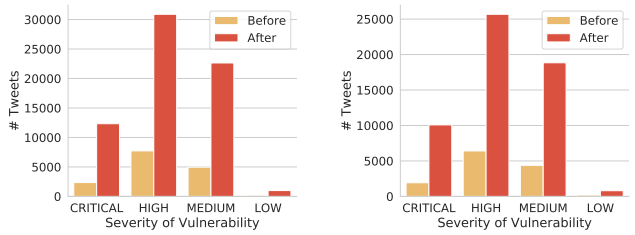
The next question we asked was *how the severity of the security vulnerabilities affects the timing of vulnerability mentions on the two platforms*. When looking at the volume of posts/tweets by the severity of the CVE (as described in the National Vulnerability Database), we observed that, independent of severity, mentions of particular vulnerabilities are more likely to happen after their public disclosure (Figure 3). This apparent contradiction of the observation from Tables 2 and 3 is due to the large volume of posts in the CVEWatch subreddit. This subreddit is a bot-driven channel that automatically posts new CVEs from the National Vulnerability Database. When we remove the identifiable posts/tweets by bots, we see a significant difference between the two platforms (Figures 3b and 3d). While on Twitter the pattern of higher number of CVE mentions after the public disclosure holds, in Reddit it appears that more CVE mentions happen before public disclosure. We observe the same behavior in Figures 4a and 4b, which present the distribution of CVE identifiers mentioned in the top 5 most popular subreddits/hashtags, respectively.

How do CVE mentions spread over social-media platforms? Another way to look at the timing of discussions is to structurally analyze the information cascades that mention CVEs. Table 4 presents the statistics of information cascades on the two platforms. We consider three types of cascades based on the relative timing of the start and end of a cascade. The cascades that start and end before the public disclosure day are represented as *Before (completed)*. That is, the date of the last comment or retweet of a cascade that at any point mentions a particular CVE is before the date when that CVE was inserted into the NVD. Surprisingly, despite the much larger volume of tweets compared to posts on Reddit, the proportion of cascades completed before Day 0 is much larger in Reddit: 80% of the discussions with the initial post before Day 0 end before Day 0. In Twitter, only 0.5% of retweet cascades end before Day 0.

What types of sentiments fuel these discussions? Figure 5 presents sentiment metrics related to the content of the cascades on both platforms. Figure 5a shows that Reddit users are more certain than uncertain both before and after the public disclosure date—a trait common, perhaps, to web-mediated conversations. As Figure 5b

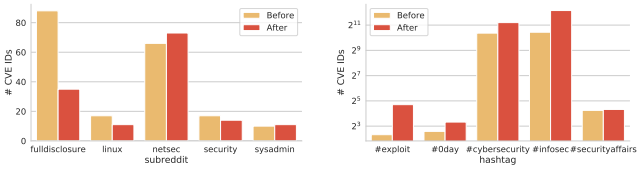


(a) Reddit (including Bots) (b) Reddit (without Bots)



(c) Twitter (including Bots) (d) Twitter (without Bots)

Figure 3: The severity of application vulnerability as published in Reddit posts and Tweets before and after the public disclosure (NVD) of the vulnerability.



(a) Reddit (b) Twitter

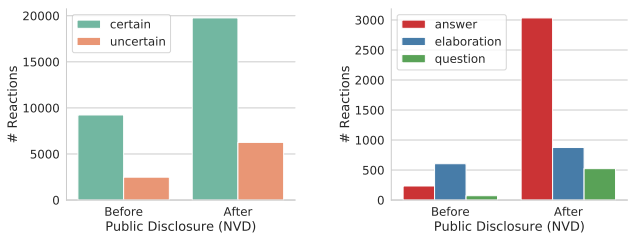
Figure 4: The discussion of security vulnerabilities in Reddit and Twitter communities is presented showing the number of CVE identifiers discussed before and after the public disclosure (NVD).

shows, the amount of replies to tweets containing CVE mentions is higher after the public disclosure and such replies are classified by the tools we are using as "answers". However, before public disclosure, most replies were classified as "elaborations".

4.2 Software Development Activities

During the same observation period of other social media channels (March 2016 to August 2017), there are 10,502 distinct CVE identifiers that are included in the text descriptions attached to GitHub events. As shown in Table 5, most CVE identifiers appear in GitHub commit messages. Out of the distinct CVE identifiers that appeared in the GitHub dataset, 40% are also on Twitter, while only 3% appear on Reddit.

How does GitHub activity depend on the public disclosure of security vulnerabilities? Figure 6 presents the number of GitHub events



(a) Reddit Reactions (b) Twitter Reactions

Figure 5: Reaction in Reddit and Twitter during and after Black Risk phase. This only includes reaction towards posts/ tweets done by humans.

that refer to a CVE. While the majority of CVEs are mentioned in only one GitHub event, some vulnerabilities are mentioned in many GitHub actions. As an example, we highlight the pattern of GitHub activity over time that mentions an issue in the Linux kernel (see the zoom-in plot in Figure 6). The red dashed-line represents the public disclosure of the issue in the NVD. Interestingly, the burst of activity occurs eight months after the public disclosure of the vulnerability. This might be due to a passive software development life-cycle where developers do not address a vulnerability until a major exploit. In this case, a massive exploit to the Android operating system based on this vulnerability (known as the *Iovyroot exploit*) affected many Android mobile devices in early 2016. There are 1,265 GitHub repositories that addressed this issue, where 2,911 users performed 3,098 software development activities in total.

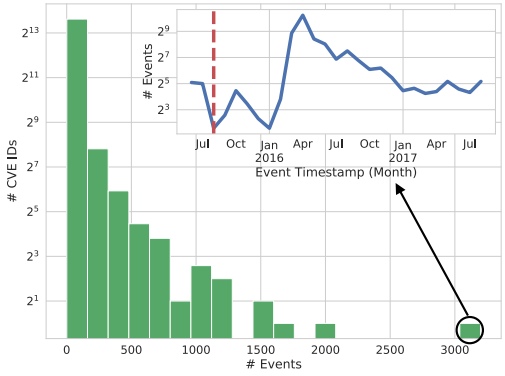


Figure 6: The distribution of GitHub events associated with CVE IDs. The insert presents the number of GitHub events over time that are related to `CVE-2015-1805`, a vulnerability in the Linux kernel.

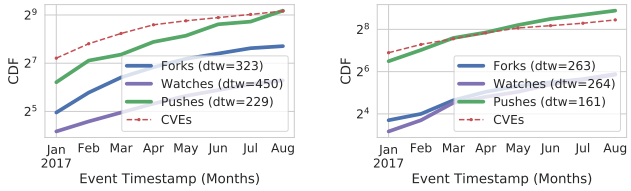
How does GitHub activity correlate with the number of CVEs for the most vulnerable repositories? We looked at the two repositories with the largest number of distinct CVE identifiers mentioned during the eight months of 2017, with 573 and, respectively, 349 CVEs. Figure 7 shows the activities that correlate best with the number of CVEs: fork, watch and push. We calculated *dynamic time warping (dtw)* [4]

Table 4: Properties of cascades in Reddit before and after public disclosure. For the cascade properties, we only consider posts or tweets with at least one comment/retweet. We compare the properties of cascades started during the Black Risk phase but completed before and after the public disclosure of the vulnerability.

Measurement	Reddit - Black Risk Phase			Twitter - Black Risk Phase		
	Before (completed)	Before (not completed)	After	Before (completed)	Before (not completed)	After
Number of cascades	122	30	187	7	1,479	5,541
Number of adoptions	2,581	1,334	4,181	20	5,504	19,537
Number of unique users	635	277	1,186	8	1,355	2,455
Cascade size (Mean/Max)	21/ 336	44/ 342	22/ 814	3/ 8	3/ 51	3/ 424
Cascade breadth by level (Mean/ Max)	10/ 21	12/ 30	20/ 73	1/ 1	2/ 34	2/ 106

Table 5: The distribution of distinct CVE identifiers as they appeared in GitHub event texts during 2016/03—2017/08.

Event Type	Entity	#CVE IDs
Push	commit message	8,181
Issue	description	2,762
Pull-Request	description	2,606
Issue-Comment	comment	2,344
Pull-Request	title	1,320
Release	description	1,269
Issue	title	557
Fork	description	171
Create	description	141
Commit-Comment	comment	119
Pull-Request Review Comment	comment	53



(a) Repository with 573 CVEs **(b) Repository with 349 CVEs**

Figure 7: The time-series of GitHub events in 2017 for two repositories found to have the most CVE identifiers associated with them in 2017. We also present the timeline of associated CVEs as they published in NVD (red dashline).

to measure trend similarity between two time-series: the monthly time-series of events and the number of associated CVE identifiers. As expected, the pattern of GitHub pushes shows a similar trend with the number of CVE mentions. This is expected, since software patches are contributed to the repository code through GitHub pushes. However, it turns out that the time-series of fork and watch activities, which are measures of popularity, are the next most similar events with respect to the CVE mentions (dtw is 323 and 263). This unexpected popularity in a vulnerable software repository may be due to interest in learning about bug fixes, developing bug fixes via development on a forked repository, or developing exploits to the vulnerable software. This analysis suggests that only specific types of GitHub activity seem to be influenced by the volume of CVEs mentioned.

5 PREDICTING GITHUB ACTIVITY FROM SOCIAL MEDIA

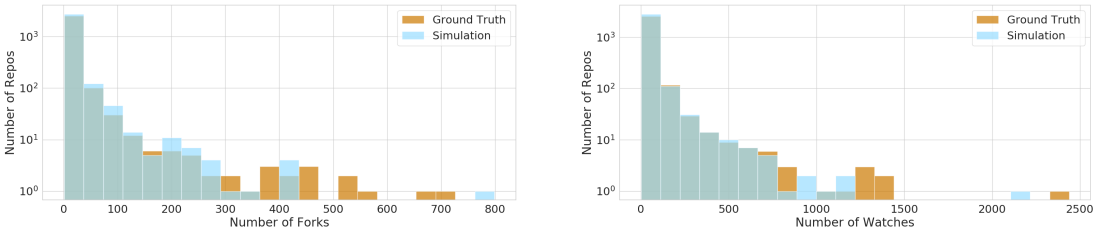
The question we aim to answer in this section is: *Can Twitter and Reddit CVE mentions help predict the actual activity on relevant repositories on GitHub? (RQ2)* This question is relevant for multiple reasons. First, it closes the loop from discussions (on Twitter and Reddit) to actions (on GitHub). To our knowledge, this is the first such study related to software vulnerabilities that studies interrelated events on these three online platforms. Second, this study acts as a proof-of-concept scenario, in which activity from two platforms can reveal the activity on a different platform on a related topic. Third, it strengthens the case that diverse online platforms are part of an ecosystem in which signals travel cross-platform. This evidence stresses the need for cross-platform studies for various research problems, such as information diffusion in online media.

5.1 Machine Learning Setup

We trained two machine-learning models to predict GitHub events. A GitHub event consists of the type of action E_p (as listed in Table 5), associated GitHub repository R, the identity of a user who performed the action U and the event time-stamp T_h in hours.

We first we train a neural network to predict the total number of GitHub events that occurred in a day. For this model, we only used the activity-driven features from Reddit and Twitter. These features include the *daily count of posts, daily count of active authors, daily count of active subreddits, and daily counts of comments* on Reddit; and *daily count of tweets, daily count of tweeting users, daily count of retweets, and daily count of retweeting users* on Twitter. During training and testing, the feature vector X_1 contained these 8 features, and the target value Y_1 contained the total number of GitHub events for the respective day.

The second step is to distribute the prediction output of the first machine learning model (i.e., total events in a day) across users and repositories. For this task we trained a recurrent neural network for each GitHub event type to predict the likelihood of a user action to a given GitHub repository in a particular hour. The feature vector X_2 contained 24 values, where each value represents the likelihood



(a) Repository Liveliness

(b) Repository Popularity

Figure 8: GitHub Popularity: the distribution of a) Fork events and b) Watch events across GitHub repositories.

of user performing an action to a repository in a particular hour of the day. The target vector Y_2 is in a similar format of X_2, but describes the user-repository actions in the following 24 hours after the last hour captured in X_2. Once this model predicts the likelihood of user actions in an hour, it multiplies with the prediction output of first machine learning model (i.e., total events in a day) to derive the events $< E_p, R, U, T_h >$.

The first machine-learning model is a simple 2-layer fully connected neural network with the layers of 100 and 10 hidden neurons. The second model consists of a long short term memory (LSTM) neural network of 10 hidden units. The model loss function was *mean absolute error* and the optimizer was *ADAM*. Each model was trained 100 epochs and parameters corresponding to the lowest validation loss in these 100 epochs were chosen for the model. We used GitHub data from January 2017 to May 2017 (five months) as training data, the next two months as validation data, and the last month (August) as test data.

5.2 Prediction Results

We predicted GitHub events across the days in August 2017 and compared the predictions with ground-truth data. We only report our results on two particular GitHub events, Fork and Watch, as they measure the popularity of repositories. Intuitively, a repository that contains a particular security vulnerability will be watched by users interested in notifications about possible patches. Similarly, users who are not contributors to that repository might want to develop their own patches and thus fork (copy) that repository.

Figure 8a presents the distribution of Fork events across GitHub repositories, and Figure 8b presents the distribution of Watch events. Both distributions of predicted events are close to the equivalent ground truth distributions: the *js* divergence scores are 0.0029 and 0.0020, and the r2 scores are 0.6300 and 0.6067, respectively. The Jensen-Shannon (js) divergence is a finite measure applicable to finite random variables. It can be used to quantify how "distinguishable" the predicted distribution is from the ground truth distribution. It ranges from 0, indistinguishable (our goal), to 1. The low *js* divergence scores we obtained suggests that we got the distributions of the number of Fork and Watch events very close to the ground truth. The coefficient of determination (r2 metric) is a statistical measure that ranges from 1, a perfect fit, to $-\infty$ and quantifies the goodness of fit of a model. In our case, the r2 metric (close to 1) shows our predictions fit well the ground truth.

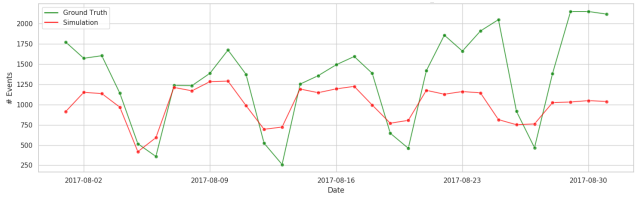


Figure 9: The growth of the most active GitHub repository by the number of daily events occurring in 2017/08.

Figure 9 presents the timeseries of GitHub events on the most active repository in August 2017. The figure shows the results of long-term simulation: that is, unlike typical prediction tasks, the daily predicted counts of events do not use the ground truth of the previous day. Instead, the prediction of the next data point is based on the predicted value of the previous day. The simulations follows accurately the ground truth data for the first week. The distance between simulated and ground truth data increases significantly in the third and fourth week. These results show, on one hand, the predictive power over two weeks, and on the other hand, the limitations for longer intervals.

6 CONCLUSIONS

This paper compares the volume and timing of software vulnerability mentions on three social platforms over a period of 18 months, from March 2016 to August 2017. In addition, this work presents machine-learning models that predict the patterns of popularity and engagement level activities in GitHub using activity information from Reddit and Twitter. The set of security vulnerabilities considered was recorded officially in the National Vulnerability Database between January 2015 and May 2018.

Our data analysis leads to the following observations. First, it appears that the volume of security vulnerability mentions is significantly higher on Twitter than on Reddit. Moreover, of the vulnerabilities mentioned on both platforms, the ones on Twitter typically appear slightly earlier. This suggests that Twitter is a better platform to monitor for early vulnerability alerts.

Second, however, of the vulnerability mentions on Reddit, most occur before public disclosure. Moreover, these early mentions generate viral and deeper information cascades than the early mentions on Twitter do. In contrast, cascades on Twitter emerge only after

the official public disclosure. This distinction is probably motivated by the different use of the two platforms in this particular context. While Twitter is a broadcast medium, Reddit is a discussion forum. Consequently, broadcasting activities become energized by public disclosures of vulnerabilities, while discussions on Reddit seem to be driven by a professional community.

Third, the majority of GitHub activities in response to vulnerabilities takes place (sometimes long) after the public disclosure. Yet the signals from Reddit and Twitter are useful for accurately predicting the overall volume of user activities on repositories likely affected by security vulnerabilities. We stress that we do not suggest that activity on Twitter or Reddit drive the activity on GitHub. Instead, our study shows that the three platforms respond in some coordinated way to the same external events, namely the identification of software vulnerabilities.

The practical implications of this study are multiple. For example, an early cyber-security alert tool could be developed that monitors multiple social media channels and is calibrated to the specifics of each such channel. Specifically, such a tool would be sensitive to the fact that Twitter covers more vulnerabilities but Reddit sends earlier signals. Another practical application is in designing tools for knowledge collection: while the volume of tweets and retweets may reflect the risk level of a vulnerability, the Twitter replies and Reddit conversations may provide clues on mitigation approaches. Our analysis revealed that comments in Reddit provide more certitude both before and after the public release of software vulnerabilities. Similarly, Twitter reply messages, while late in the vulnerability life cycle, often consist of answers.

Another practical application of this work is related to predicting the GitHub activity using the Twitter and Reddit signals. According to [18], 73% of software vulnerabilities targeted by Advanced Persistent Threat groups have available public exploits hosted on GitHub. Thus, predicting the activity on GitHub can be useful as a monitoring tool to identify unusual behavior such as increased activity on particular GitHub repositories or a DoS attack. Another use scenario could be creating a realistic trace of activity to replace missing real workloads.

Future work directions include the study of the evolution of CVE-specific actions across the three platforms. Specifically, we would like to understand whether signals from one platform affect the activity on other platforms, as it has been shown in different contexts [27]. This would enable us to track the evolution of a cyber-threat and develop better mitigation strategies. Also, we plan to extend our analysis to other types of social media content (e.g., multi-media content) related to security vulnerabilities.

ACKNOWLEDGMENTS

This work is supported by the DARPA SocialSim Program and the Air Force Research Laboratory under contract FA8650-18-C-7825. The authors would like to thank Leidos and Netanomics for providing data.

REFERENCES

[1] Mohammed Almukaynizi, Eric Nunes, Krishna Dharaiya, Manoj Senguttuvan, Jana Shakarian, and Paulo Shakarian. 2017. Proactive identification of exploits in the wild through vulnerability mentions online. In *2017 International Conference on Cyber Conflict (CyCon US)*. IEEE, 82–88.

[2] William A Arbaugh, William L Fithen, and John McHugh. 2000. Windows of vulnerability: A case study analysis. *Computer* 33, 12 (2000), 52–59.

[3] Github Archive. 2018. GH Archive. http://www.gharchive.org/.

[4] Donald J Berndt and James Clifford. 1994. Using dynamic time warping to find patterns in time series.. In *KDD workshop*, Vol. 10. Seattle, WA, 359–370.

[5] David M Beskow and Kathleen M Carley. 2018. Bot-hunter: A Tiered Approach to Detecting & Characterizing Automated Activity on Twitter. (2018).

[6] Orcun Cetin, Carlos Ganan, Maciej Korczynski, and Michel van Eeten. 2017. Make notifications great again: learning how to notify in the age of large-scale vulnerability scanning. In *Workshop on the Economy of Information Security*.

[7] Michel Cukier. 2018. Study: Hackers Attack Every 39 Seconds. https://eng.umd.edu/news/story/study-hackers-attack-every-39-seconds.

[8] Stefan Frei, Bernhard Tellenbach, and Bernhard Plattner. 2008. 0-day patch exposing vendors (in) security performance. *BlackHat Europe* (2008).

[9] Maria Glenski, Tim Weninger, and Svitlana Volkova. 2018. Identifying and Understanding User Reactions to Deceptive and Trusted Social News Sources. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 176–181.

[10] Clayton J Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Eighth international AAAI conference on weblogs and social media*.

[11] Yang Liu, Armin Sarabi, Jing Zhang, Parinaz Naghizadeh, Manish Karir, Michael Bailey, and Mingyan Liu. 2015. Cloudy with a chance of breach: Forecasting cyber security incidents. In *24th {USENIX} Security Symposium*. 1009–1024.

[12] Peter Mell and Tim Grance. 2002. *Use of the common vulnerabilities and exposures (cve) vulnerability naming scheme.* Technical Report. National Inst. Of Standards and Technology Gaithersburg Div.

[13] Sudip Mittal, Prajit Kumar Das, Varish Mulwad, Anupam Joshi, and Tim Finin. 2016. Cybertwitter: Using twitter to generate alerts for cybersecurity threats and vulnerabilities. In *Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. IEEE Press, 860–867.

[14] Varish Mulwad, Wenjia Li, Anupam Joshi, Tim Finin, and Krishnamurthy Viswanathan. 2011. Extracting information about security vulnerabilities from web text. In *Proceedings of the 2011 IEEE/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*. IEEE Computer Society, 257–260.

[15] Nexusguard. 2018. DDoS Threat Report 2018 Q2. https://www.nexusguard.com/.

[16] Eric Nunes, Ahmad Diab, Andrew Gunn, Ericsson Marin, Vineet Mishra, Vivin Paliath, John Robertson, Jana Shakarian, Amanda Thart, and Paulo Shakarian. 2016. Darknet and deepnet mining for proactive cybersecurity threat intelligence. *arXiv preprint arXiv:1607.08583* (2016).

[17] PNNL. 2018. Pacific Northwest National Laboratory, Socialsim. https://github.com/pnnl/socialsim.

[18] RFSID. 2016. Running for Office: Russian APT Toolkits Revealed. https://www.recordedfuture.com/russian-apt-toolkits/.

[19] Carl Sabottke, Octavian Suciu, and Tudor Dumitras. 2015. Vulnerability Disclosure in the Age of Social Media: Exploiting Twitter for Predicting Real-World Exploits.. In *USENIX Security Symposium*. 1041–1056.

[20] Anna Sapienza, Alessandro Bessi, Saranya Damodaran, Paulo Shakarian, Kristina Lerman, and Emilio Ferrara. 2017. Early warnings of cyber threats in online discussions. In *Data Mining Workshops (ICDMW), 2017 IEEE International Conference on*. IEEE, 667–674.

[21] Anna Sapienza, Sindhu Kiranmai Ernala, Alessandro Bessi, Kristina Lerman, and Emilio Ferrara. 2018. DISCOVER: Mining Online Chatter for Emerging Cyber Threats. In *Companion of the The Web Conference 2018 on The Web Conference 2018*. International World Wide Web Conferences Steering Committee, 983–990.

[22] Clemens Sauerwein, Christian Sillaber, Michael M Huber, Andrea Mussmann, and Ruth Breu. 2018. The Tweet Advantage: An Empirical Analysis of 0-Day Vulnerability Information Shared on Twitter. In *IFIP International Conference on ICT Systems Security and Privacy Protection*. Springer, 201–215.

[23] Ben Stock, Giancarlo Pellegrino, Frank Li, Michael Backes, and Christian Rossow. 2018. Didn't You Hear Me?—Towards More Successful Web Vulnerability Notifications. *The Network and Distributed System Security Symposium (NDSS)* (2018).

[24] Romilla Syed, Maryam Rahafrooz, and Jeffrey M Keisler. 2018. What it takes to get retweeted: An analysis of software vulnerability messages. *Computers in Human Behavior* 80 (2018), 207–215.

[25] Soroush Vosoughi, Mostafa'Neo' Mohsenvand, and Deb Roy. 2017. Rumor gauge: Predicting the veracity of rumors on Twitter. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 11, 4 (2017), 50.

[26] Ning Yu and Graham Horwood. 2018. Veracity Enriched Event Extraction. In *2018 International Workshop on Social Sensing (SocialSens)*. 3–3. https://doi.org/10.1109/SocialSens.2018.00010

[27] Savvas Zannettou, Tristan Caulfield, Emiliano De Cristofaro, Nicolas Kourtellis, Ilias Leontiadis, Michael Sirivianos, Gianluca Stringhini, and Jeremy Blackburn. 2017. The web centipede: understanding how web communities influence each other through the lens of mainstream and alternative news sources. In *Proceedings of the 2017 Internet Measurement Conference*. ACM, 405–417.