Leveraging Online Social Networks For a Real-time Malware Alerting System

Isra' Al-Qasem, Sumaya Al-Qasem, and Ahmad T. Al-Hammouri
Jordan University of Science and Technology,
Irbid 22110, JORDAN
Email: {iaelqasem07,sralqasem07}@cit.just.edu.jo, hammouri@just.edu.jo
URL: http://www.just.edu.jo/~hammouri

***Abstract*—Online social networks (OSNs), such as Twitter and Facebook, allow users to post relatively short update messages online. Users usually post messages reporting on their activities, or even on their feelings. With OSNs' users reaching hundreds of millions around the globe, their update messages, although are radically very diverse in topics, represent a fertile environment for mining for useful information. Previous studies exploited OSNs to obtain early alerts about earthquakes, to infer websites and online services availability, and to predict stock market moves.**

In this paper, we propose to utilize OSNs to alarm against the spread of new malware attacks, such as viruses, worms, or Trojan horses. Currently, network administrators and operators use manual and traditional ways of communication, such as phones and e-mails, to warn one another against such attacks. Instead, we propose an automatic platform that mines Twitter posts to provide real-time alerts of malware propagation.

***Index Terms*—Crowd Sourcing, Online Social Networks (OSNs), Real-time Malware Alerting, Web Data Mining, Twitter.**

I. Introduction

Over the past five years, the number of people on social networks has grown extremely rapidly. An interesting list that mentions the world population rankings including social media networks would look like [20]:

1. China
2. India
3. **Facebook**
4. USA
5. **MySpace**
6. Indonesia
7. Brazil
8. **Twitter**

So, if Facebook [4], MySpace [14], and Twitter [20] were countries, they would be the third, the fifth, and the eighth largest countries in the world, respectively.

Online social networks (OSNs), such as Twitter and Facebook, allow users to post relatively short update messages online. Users usually post messages reporting on their activities, or even on their feelings. With OSNs' users reaching hundreds of millions around the globe, their update messages, although are radically very diverse in topics, represent a fertile environment for mining for useful information. Previous studies exploited OSNs to obtain early alerts about earthquakes, to infer websites and online services availability, and to predict stock market moves.

In this paper, we propose to utilize OSNs to alarm against the spread of new malware attacks, such as viruses, worms, or Trojan horses. Currently, network administrators and operators use manual and traditional ways of communication, such as phones and e-mails, to warn one another against such attacks. Instead, we propose an automatic platform that mines Twitter posts to provide real-time alerts of malware propagation.

To this end, we built a system that regularly collects users' posts matching certain queries. The system then filters the data to extract the only relevant ones. Next, the system computes the variability of the data to detect any anomalies pertinent to significant events. We present a representative case study that demonstrates the applicability of our approach, where the scheme is able to trigger alerts earlier than security-related online posts.

The rest of the paper is structured as follows. In Section II, we discuss the research exploiting OSNs and their data. Section III presents our methodology we used in this paper including all steps from data collection to eventually triggering an alert. Section IV presents a case study that demonstrates the benefit of the proposed approach. Future work is presented in Section V, and finally, Section VI concludes the paper.

II. Related Work

Research utilizing OSNs can be categorized into two different directions. First is the social-aspect research of OSNs. This type of research studies the relations and interactions, and the demographic characteristics of the OSNs users; see for example [8], [10]. The second type is concerned with extracting useful information from the posts of users. For example, [18] builds an earthquake reporting system, which is called Toretter. From the time and the location properties associated with each tweet, the system sends warning messages when an earthquake event has occurred. A second example, which inspired us, uses Twitter's data to infer about online services availability and detects when popular services (e.g., Amazon, Gmail, Bing, PayPal, etc.) experience downtime, and it triggers an outage alarm [13]. A third example investigated daily Twitter posts to predict the stock market, whereby the proposed system is claimed to predict the stock market with 87% accuracy [2]. Two researchers at Johns Hopkins University analyzed more than 1.5 billion tweets for health-related terms to predict the spread of diseases and their research showed that Twitter is also a useful tool for medical staff [16], [17]. Last but not least, Twitter was a powerful tool for raising and spreading the awareness of the Arab Spring Revolutions [3].

978-1-4799-0537-9/13/$31.00 ©2013 IEEE

Our research in this paper falls squarely within the theme of the second type of research, in essence, it mimics the work in [16], but instead of dealing with real viruses infecting humans, we here focus on the cyber counterparts: the digital viruses (or malware in general).

III. Methodology

In this section, we elaborate on our methodology and the complete steps from retrieving the tweets until triggering an alert.

The proposed social approach of detecting viruses and malware events treats Twitter's data as signals that a malware is spreading, and Twitter becomes like a sensing device that sends relevant alarms.

However, instead of retrieving tweets directly from Twitter, we query and obtain the data from Topsy [19]. Topsy is a real-time search engine that is scoped for social networks in general and for Twitter in specific. It caches online posts, archives them, and removes noise from the collected data. We chose Topsy over Twitter for the following two reasons:

- Twitter's API does not allow retrieving of data older than a week; whereas, Topsy can give tweets that date back to mid 2008. Topsy's API allows searching and retrieving of data during a specific period of time.
- Instead of dealing and sifting through Twitter's *raw* data that may contain spam or *re-tweets*, we utilize Topsy's *influence* algorithm that purges spam, nonsense, and noise tweets [22], [21].

So, we here use Topsy as a black-box to filter good Twitter's data from the noisy tweets. Note that Topsy crawls twitter in real time and does not compromise the real-time nature of Twitter [19].

A. Data Collection and Parsing

We built a web application that continuously and periodically queries Topsy's APIs for specific keywords, and returns the results in JSON format [9]. The specific keywords we use are **Malware**, **Backdoor**, and **Cyber attack** (as one keyword). Although we experimented with a slightly larger set of keywords, these three terms gave the most stable results (we revisit this issue in Section V). JSON-based results are parsed and the core textual data is stored locally for next steps.

B. Data Filtering

Even if a tweet included one of the three keywords mentioned above, it may not be suitable for detecting malware and virus events happening now. For example, a tweet such as "Could malware kill QR code?", or "Canadian Survey: Malware Attacks Up Because of Social Media" is truly including the keyword malware, but they are not useful for the real-time alert system. Therefore, we implemented a simple, yet an effective filtering algorithm to extract tweets that are appropriate as event triggers. The idea is simple: we extract the tweets that contain one of the following phrases: computer security, new, discover, hit, infect, warn, and watch out.

This filtering step is different from the filtering process performed by Topsy. Topsy would pass through the two examples above, especially if they come from 'credible' users. The filtering step here is based on the context or the 'meaning' of the tweet in reference to our objective.

C. Smoothing the Data

Similar to previous studies [7], [13], we trigger events based on the number of tweets. That is, if the number of tweets on a given topic increases sharply, a significant event must have therefore occurred that ignited Twitter users to get concerned and to tweet voluminously about the corresponding topic. A very straightforward method to trigger an event is then to monitor the number of tweets over time and at a given time, if the number of tweets exceeds the number of tweets at a previously observed instant by some threshold, an event is fired. However, as with any type of sensors, there stems a need to smooth (or low-pass filter) the output signal to base the decision (or the control) on the general trend of the sensor's output rather than on the instantaneous value of the output. As such, we here use an Exponentially Weighted Moving Average (EWMA) [11] to low-pass filter the number of tweets after the data-filtering step of the previous section. In addition, we use an Exponentially Weighted Moving Variance (EWMV) [6] to monitor and detect the increase/rise in the number of tweets. (The tweets here represent the physical process to be observed.) This same approach was also used in [13]. We next elaborate upon the steps we followed.

1) Based on their posting time, we bin the tweets resulting from the filtering step above into 6-hour intervals. Let $X[k]$ be the number of tweets in interval k.
2) We compute the EWMA value, $Y[k]$, in interval k as

$$Y[k] = \lambda \cdot X[k] + (1-\lambda) \cdot Y[k-1] \; , \forall k > 1,$$

where $\lambda \in (0,1)$ is the smoothing factor constant, and is chosen in this paper heuristically to be 0.2.

3) We compute the differential, $D[k]$, between the actual number of tweets, $X[k]$, in interval k and the EWMA value, $Y[k-1]$, in interval $k-1$. If an event occurs, then one would expect $X[k]$ to deviate significantly from the general trend, $Y[k-1]$. $D[k]$ is thus given by

$$D[k] = X[k] - Y[k-1] \; .$$

We then compute the EWMV as [6]

$$(V[k])^2 = \gamma \cdot (D[k])^2 + (1-\gamma) \cdot (V[k-1])^2 \; , \forall k > 1,$$

where $\gamma \in (0,1)$ is another smoothing factor constant, and is also chosen in this paper heuristically to be 0.2. Finally, we compute the threshold that classifies whether the increase in the number of tweets is significant as [13]

$$T[k] = Y[k-1] + \delta \cdot V[k-1] \; ,$$

where δ is a constant that determines how much deviation from the normal behavior, $Y[k]$, is considered abnormal. In this paper, we choose δ to be 3. (Note that we did not research the best way to ascertain the values of λ, γ, and δ, and we leave this for future work.)

D. Triggering a Malware Alert

Finally, a malware alert is triggered when the actual number of tweets, $X[k]$, in a given interval k exceeds the threshold value, $T[k]$, in the same interval.

IV. Results

In this section, we present an example that demonstrates the usefulness of the proposed approach. Figure 1 shows the curves for the number of tweets, the calculated EWMA, and the threshold associated with the three keywords: malware, backdoor, and cyber attack for the period from October 17, 2011 12:00 (noon) to October 23, 2011 00:00 (midnight).

From the figure, the number of tweets exceeds the threshold at five different instants collectively for the three keywords. These five events are summarized in Table I. The table shows when each event is detected based on Twitter's data (i.e., from Figure 1), provides descriptive details about each event, and references the security-specialized articles that first mentioned the given even and the date when the article is published. From the table, there is at least one event corresponding to one keyword that is triggered earlier than the posting of security-related articles reported on the same event.

We obtained other sets of results in other time periods that we do not show in here due to space constraints, but they convey the similar information of the results presented in this paper, i.e., they all validate the proposed approach.

A. Malware Spreading Methods

For some of the results above, we tried to infer from Twitter what ways each malware is spreading via. Criminals always come up with new ways to get malicious software into victims' computers. Therefore, determining the spreading method for a given malware represents a useful piece of information. We here focus on the most pervasive and contemporary ways of spreading, such as via opening an E-mail attachment, visiting a specific website, responding to pop-ups of a given website, and clicking links distributed over and using certain applications within social networking websites, e.g., Twitter or Facebook. A user can then take more care when dealing with warned-against methods, and a network administrator can disable features or block contents for users inside his/her organization accordingly.

For instance, mining the collected data, we could ascertain that the malware related to the Gaddafi's death as reported above was spreading via an E-mail attachment. This is just a preliminary result and further research is left for future work.

V. Future Work

To solidify the proposed simple approach to furnish a highly dependable system for malware alarming based on online social networks, there still exist several issues to be addressed. First, our current system does not differentiate between two or more malwares spreading in the same time—it just alerts that there is a malware appearing. One possible solution to this issue is to apply automatic document clustering and summarizing approaches [1], and document similarity measures algorithms [5] on the tweets. Second, it is important to concretely tag each malware of the way it is spreading via, e.g., spam E-mails, bogus links, etc. Also, it is useful to tell what operating system a malware is specifically targeting. Other directions for future work include using statistical methods that are insensitive to the choice of parameters to detect the variability of the number of tweets, e.g., using enhanced methods to compute the EWMA and the EWMV [6]. Finally, as we mentioned in Section III, we experimented with a larger set of keywords than those we showed results for. For example, we used the keyword 'virus', but we found that it needs special filtering to extract the relevant tweets appropriate for the proposed malware alerting system. So, in the future, we plan to delve into similar issues and to experiment with other keywords.

VI. Conclusion

Computer malware and viruses cause problems to computer users (individuals and organizations) with different levels of severity, and the ability of the malware to pass on between computers is a real threat.

Anti-virus software packages can detect, remove and sometimes prevent such threats, but cannot completely prevent infection of computers by malwares and viruses and the consequent damages and losses, especially for newly appearing ones. Some recent research, which studied the time spent on social networking sites, concluded that an average user spends more than three hours a day on popular networks, such as Twitter. This popularity—which is achieved by the real-time micro blogging service—and the huge amount of information—which is produced by the users reflecting on their daily issues—attracted experts, researchers, and even people in business to build different innovative applications utilizing the power of crowd sourcing.

These interesting phenomena of social networking also attracted us, and so we benefited from malware victims' notes posted on Twitter to warn other people. The system, which uses real-time data from Twitter, triggers malware alerts, based on tweets matching specific queries. As we discussed, we pass these tweets through filtering and smoothing stages to reduce false alarms.

Finally, the proposed real-time malware alerting system, which collects data from human generated notices, can largely reduce the risk of harmful malwares and viruses by early alarming against such threats.

References

[1] R.M. Aliguliyev. Automatic document summarization by sentence extraction. *Journal of Computational Technologies*, 12:5–15, 2007.

[2] Johan Bollen, Huina Mao, and Xiao-Jun Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, March 2011.

[3] Can watching Twitter trends help predict the future? [Online]. Available: http://gigaom.com/2011/10/19/can-watching-twitter-trends-help-predict-the-future/.

[4] Facebook. [Online]. Available: https://www.facebook.com/.

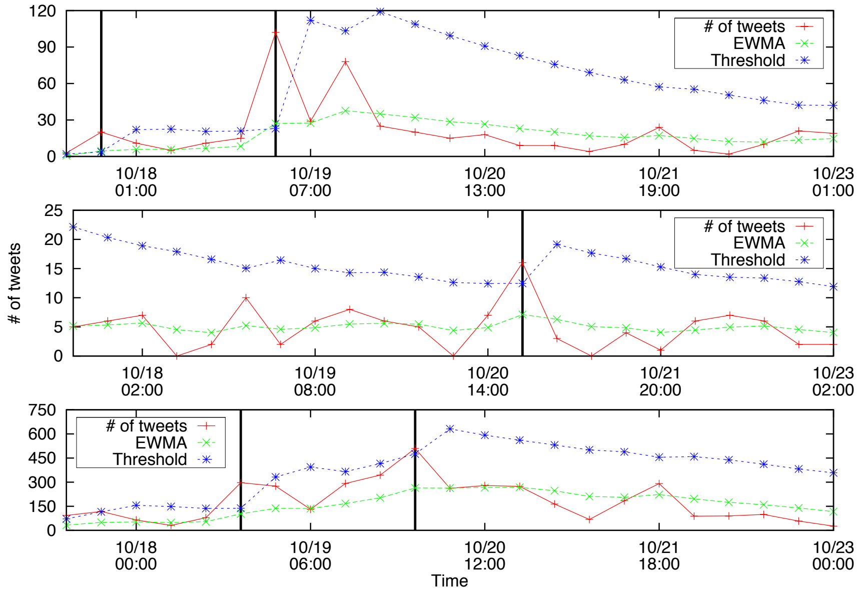

Fig. 1. Number of tweets, the calculated EWMA, and the threshold curves associated with the three keywords: malware (bottom), backdoor (middle), and cyber attack (top) for the period from 10/17/2011 to 10/23/2011. Vertical solid lines highlight when number of tweets exceeds the threshold value and thus an alert is triggered.

Event #	Detected date & time	Associated keyword	Event details	Reported date	Source
1	Oct. 17, 2011 19:00	cyber attack	Mac malware & Malware carrying Stuxnet code	Oct. 18, 2011	[15]
2	Oct. 18, 2011 18:00	malware	Mac malware & Malware carrying Stuxnet code	Oct. 18, 2011	[15]
3	Oct. 19, 2011 01:00	cyber attack	Mac malware & Malware carrying Stuxnet code	Oct. 18, 2011	[15]
4	Oct. 20, 2011 00:00	malware	Malware photos related to Gaddafi's death	Oct. 21, 2011	[12]
5	Oct. 20, 2011 20:00	backdoor	Malware photos related to Gaddafi's death	Oct. 21, 2011	[12]

TABLE I

FIVE TRIGGERED EVENTS BASED ON FIGURE 1. COLUMN TWO GIVES THE DATE AND TIME WHEN THE NUMBER OF TWEETS EXCEEDS THE THRESHOLD IN FIGURE 1. COLUMN THREE GIVES THE KEYWORD BASED ON WHICH THE EVENT IS TRIGGERED. COLUMN FOUR PROVIDES DETAILS AND DESCRIPTION ABOUT THE EVENT. COLUMNS FIVE AND SIX GIVE THE DATE AND THE SOURCE ARTICLES THAT FIRST MENTIONED THE GIVEN EVENT.

[5] A. Huang. Similarity measures for text document clustering. In *Proceedings of the Sixth New Zealand Computer Science Research Student Conference*, pages 49–56, Christchurch, New Zealand, 2008.

[6] Longcheen Huwang, Yi-Hua Tina Wang, Arthur B. Yeh, and Ze-Shiang Jason Chen. On the exponentially weighted moving variance. *Naval Research Logistics (NRL)*, 56(7):659–668, 2009.

[7] Japan Earthquake Shakes Twitter Users ... And Beyonce. [Online]. Available: http://mashable.com/2009/08/12/japan-earthquake/.

[8] Akshay Java, Xiaodan Song, Tim Finin, and Belle Tseng. Why we twitter: understanding microblogging usage and communities. In *Proceedings of the 1st SNA-KDD workshop on Web mining and social network analysis*, 2007.

[9] JSON. [Online]. Available: http://www.json.org/.

[10] Balachander Krishnamurthy, Phillipa Gill, and Martin Arlitt. A few chirps about twitter. In *Proceedings of the first workshop on Online social networks*, WOSN '08, pages 19–24, New York, NY, USA, 2008.

[11] James M. Lucas and Michael S. Saccucci. Exponentially weighted moving average control schemes: Properties and enhancements. *Technometrics*, 32(1):1–12, February 1990.

[12] Malware attack poses as bloody photos of Gaddafi's death. [Online]. Available: http://nakedsecurity.sophos.com/2011/10/21/malware-attack-poses-as-bloody-photos-of-gaddafis-death/.

[13] Marti Motoyama, Brendan Meeder, Kirill Levchenko, Geoffrey M. Voelker, and Stefan Savage. Measuring online service availability using twitter. In *Proc. of conference on Online social networks*, Berkeley, CA, 2010.

[14] Myspace — Social Entertainment. [Online]. Available: http://www.myspace.com/.

[15] New malware appears carrying Stuxnet code. [Online]. Available: http://www.scmagazineus.com/new-malware-appears-carrying-stuxnet-code/article/214707/.

[16] M. Paul and M. Dredze. A model for mining public health topics from twitter. Technical report, Johns Hopkins University, 2011.

[17] Researchers take US temperature via Twitter. [Online]. Available: http://www.bbc.co.uk/news/technology-14059745.

[18] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake shakes twitter users: Real-time event detection by social sensors. In *Proc. of International World Wide Web Conference*, April 2010.

[19] Topsy—Real-time search for the social web. [Online]. Available: http://www.topsy.com/.

[20] Twitter. [Online]. Available: http://twitter.com/.

[21] Understanding Social Signal — Topsy Labs. [Online]. Available: http://topsylabs.com/social-signal/understanding-social-signal/.

[22] Using Influence to Tune Signal to Noise on the Social Web. [Online]. Available: http://searchengineland.com/using-influence-to-tune-signal-to-noise-on-the-social-web-66602.