Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 1, January 2014, pg.69 – 74

RESEARCH ARTICLE

WarningBird MailAlert Based Malicious URLs Blocker System in Twitter

MS. SARANYA.S[1], MR. UDHAYA KUMAR.V[2]

[1]M.TECH (Computer Science &Eng), PRIST UNIVERSITY, Pondicherry
[2]Assistant Professor (Computer Science &Eng), PRIST UNIVERSITY, Pondicherry
Email: [1] sarancomp10@gmail.com, [2] udhaya_kurinji@yahoo.com

***Abstract*—Twitter is prone to malicious tweets containing URLs for spam, phishing, and malware distribution. Conventional Twitter spam detection schemes utilize account features such as the ratio of tweets containing URLs and the account creation date, or relation features in the Twitter graph. These detection schemes are ineffective against feature fabrications or consume much time and resources. Conventional suspicious URL detection schemes utilize several features including lexical features of URLs, URL redirection, HTML content, and dynamic behavior. However, evading techniques such as time-based evasion and crawler evasion exist. In this paper, we propose WARNINGBIRD, a suspicious URL detection system for Twitter. Our system investigates correlations of URL redirect chains extracted from several tweets. Because attackers have limited resources and usually reuse them, their URL redirect chains frequently share the same URLs. We develop methods to discover correlated URL redirect chains using the frequently shared URLs and to determine their suspiciousness. We collect numerous tweets from the Twitter public timeline and build a statistical classifier using them. Evaluation results show that our classifier accurately and efficiently detects suspicious URLs.WARNINGBIRD as a near real-time system for classifying suspicious URLs in the Twitter stream. In this project I proposed block the malicious URLs and providemailalert for malicious URLs occur in the twitter stream.**

***Keywords*—Twitter; correlation; share URLs; spam; reciprocity; crawl**

I. Introduction

Twitter is a microblogging service less than three years old, command more than 41 million users as of July 2009 and is growingfast. Twitter users tweet about any topic within the 140-characterlimit also know as tweets and follow others to receive their tweets. Twitter it is a new medium of information sharing.We have crawled the entire Twitter site and obtained 41:7 millionuser profiles, 1:47 billion social relations, 4; 262 trending topics, and 106 million tweets. In its follower-following topology analysiswe have found a non-power-law follower distribution, a short effectivediameter, and low reciprocity, which all mark a deviation fromknown characteristics of human social networks. When a user Alice updates (or sends) a tweet, it will be distributed to all of her *followers* who have registered Alice as one of their friends. Instead of distributing a tweet to all of her followers, Alice can also send a tweet to a specific twitter user Bob by mentioning this user by including *@Bob* in the tweet. Unlike status updates, mentions can be sent to users who do not follow Alice. Twitter, we have ranked users by the numberof followers and by PageRank and found two rankings to be similar.When Twitter users want to share a URL with friends via tweets, they usually use URL shortening services to reduce the URL length since tweets cancontain only a restricted number of characters.bit.lyand tinyurl.com are widely used services, and Twitteralso provides a shortening service t.co.

Social networking sites have become one of the main ways for users to keep track and communicate with their friends online. Sites such as Facebook,MySpace, and Twitter are consistently among the top 20most-viewed web sites of the Internet.

All current Online Social Networks(OSNs) adopt the client-server architecture.The OSN service provider acts as the controlling entity. Itstores and manages all the content in the system.OSN is using online spam filtering is deployed at the OSN service provider sideOnce deployed, it inspects every message beforerendering the message to the intended recipients and makesimmediate decision on whether or not the message underinspection should be dropped.If it is illegal message mean immediately dropped the message otherwise it is forward to the corresponding receiver.

OSN users form a huge social graph, where each noderepresents an individual user. In Facebook-like OSNs, a sociallink would connect two nodes if the two correspondingusers have mutually agreed to establish a social connection.Two users without a social link between them cannot directlyinteract with each other. Twitter-like OSNs imposelooser restrictions, where a user can "follow" anyone to establish directed social link, so that he can receive allthe updates.Recent studies suggest that the majorityof spamming accounts in OSNs are compromised account. The below fig1 Cumulative distribution of the social degree ofspamming and legitimate accounts, respectively.

The popularity of Twitter, malicious users often try to find a way to attack it. The most common forms of Web attacks, including spam, phishing, and malware distribution attacks, have also appeared on Twitter. Because tweets are short in length, attackers use shortened malicious URLs that redirect Twitter users to external attack servers.

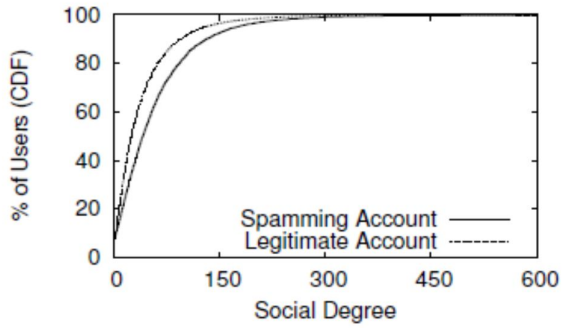

Fig. 1 Cumulative distribution of the social degree of spamming and legitimate accounts, respectively.

To cope with malicious tweets, several Twitter spam detection schemes have been proposed. These schemes can be classified into account feature-based relation feature based and message feature based schemes. Account feature-based schemes use the distinguishing features of spam accounts such as the ratio of tweets containing URLs, the account creation date, and the number of followers and friends. However, malicious users can easily fabricate these account features.The relation feature-based schemes rely on morerobust features that malicious users cannot easily fabricatesuch as the distance and connectivity apparent inthe Twitter graph. Extracting these relation features froma Twitter graph, however, requires a significant amount of time and resources as a Twitter graph is tremendousin size. The message feature-based scheme focused onthe lexical features of messages. However, spammers caneasily change the shape of their messages.

A number of suspicious URL detection schemes have also been introduced. They use staticor dynamic crawlers, and they may be executed invirtual machine honeypots, such as Capture-HPC,HoneyMonke, and Wepawet, to investigatenewly observed URLs. These schemes classify URLsaccording to several features including lexical featuresof URLs, DNS information, URL redirections, and theHTML content of the landing pages. Nevertheless, maliciousservers can bypass an investigation by selectivelyproviding benign pages to crawlers. For instance, becausestatic crawlers usually cannot handle JavaScript.or Flash, malicious servers can use them to deliver malicious content only to normal browsers. Malicious servers can also employ temporalbehaviors—providing different content at different times to evadean investigation.

II. Proposed Algorithm

Here we present the proposed OFFLINE SUPERVISED LEARNING ALGORITHM **(OSLA)** supervised algorithms require categorized examples. After presenting these examples to the algorithm, adaptations are made to the configuration such that the different categories are recognized correctly in the future. With non-supervised learning, there is no explicit set of good and bad examples. In our project, we use an offline supervised learning algorithm, the feature vectors for training are relatively older than

feature vectors for classification. To label the training vectors, we use the Twitter account status; URLs from suspended accounts are considered malicious whereas URLs from active accounts are considered benign.
we periodically update our classifier using labeled training vectors.

In this section discusses issues related to algorithm. Three steps used in our offline supervised learning algorithm

- Case-A: Frequent URL with similar domain names and from same IP address.
- Case-B: Reoccurrences of redirect chains in URLs (entry points)
- Case-C: Check whether same URL is Posted to other users(followers) from same IP .

III. System Design

Fig 2 Our system consists of six components: data collection, feature extraction, training, classification, detecting suspicious URLs, and MailAlert.

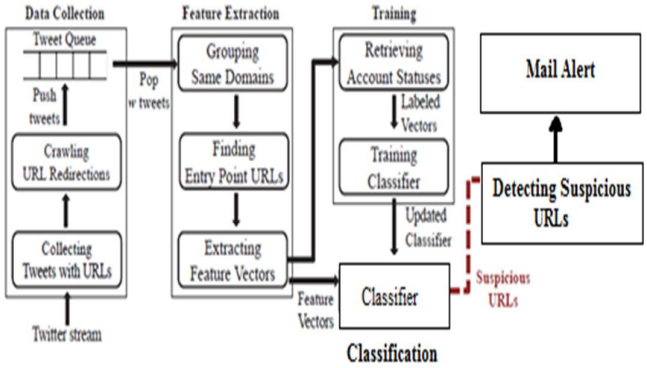

Fig. 2 System Architecture.

A. Data collection

It is important to notice that there is one important limitation imposed by the Twitter API. The number of requests could not exceed 350 per hour, which limits considerably the possibility to retrieve a large amount of samples, so we had to use several accounts to gather them. Our Java-based collecting method obtained, from the selected profiles, the users' ID and the timeline tweets, until having at least 100 genuine tweets. A genuine tweet is the tweet that is generated by the user itself (i.e., written by itself) and is not one re-tweet of another user's tweet. The data collection component has two subcomponents: the collection of tweets with URLs and crawling for URL redirections. To collect tweets with URLs and their context information from the Twitter public timeline, this component uses Twitter Streaming APIs . Whenever this component obtains a tweet with a URL, it executes a crawling thread that follows all redirections of the URL and looks up the corresponding IP addresses. The crawling thread appends these retrieved URL and IP chains to the tweet information and pushes it into a tweet queue. As we have seen, our crawler cannot reach malicious landing URLs when they use conditional redirections to evade crawlers. However, because our detection system does not rely on the features of landing URLs, it works independently of such crawler evasions.

B. Feature Extraction

Our dataset contains the following features extracted from each of the profiles the tweets, time of publication, language, geoposition and Twitter client. The first feature, the tweet, is the text published by the user, which gives us the possibility of determine a writing style, very characteristic of each individual. The time of publication helps determining themoments of the day in which the users interact in the social network. The language and geoposition also help filtering and determining the authorship because users have certain behaviors which can be extrapolated analyzing these features. Finally, despite being possible that users have several devices from where they tweet (e.g., PC, Smartphone or tablet), they usually choose to do it using their favorite Twitter client, which gives us another filtering mechanism.

The feature extraction component has three subcomponents: grouping of identical domains,finding entry point URLs, and extracting feature vectors.This component monitors the tweet queue to determinewhether a sufficient number of tweets have

beencollected. Specifically, our system uses a tweet windowinstead of individual tweets. When more than w tweetsare collected (w is 10,000 in the current implementation),it pops w tweets from the tweet queue. First, for allURLs in the w tweets, this component checks whetherthey share the same IP addresses. If several URLs share.at least one IP address, it replaces their domain nameswith a list of domains with which they are grouped. For instance, when http://123.com/hello.html and http://xyz.com/hi.html share the same IP address, this replaces these URLs with http://[‘123.com’,‘xyz.com’]/hello.html and http://[‘123.com’,‘xyz.com’]/hi.html. This grouping process enables the detection of suspicious URLs that use several domain names to bypass the blacklisting, in which each URL appears in these tweets. It thendiscovers the most frequent URL in each URL redirectchain in the *w* tweets. The discovered URLs thus becomethe entry points for their redirect chains. If two or moreURLs share the highest frequency in a URL chain, thiscomponent selects the URL nearest to the beginning ofthe chain as the entry point URL. Finally, for each entry point URL, the component findsURL redirect chains that contain the entry point URL, and extracts various features from these URL redirectchains along with the related tweet information. These feature values are then turned into real-valuedfeature vectors.

When we group domain names or find entry pointURLs, we ignore whitelisted domains to reduce falsepositiverates. Whitelisted domains are not grouped withother domains and are not selected as entry point URLs.

C. Training

The training component has two subcomponents:retrieval of account statuses and training of theclassifier. Because we use an offline supervised learningalgorithm, the feature vectors for training are relativelyolder than feature vectors for classification. To labelthe training vectors, we use the Twitter account status;URLs from suspended accounts are considered maliciouswhereas URLs from active accounts are considered benign.We periodically update our classifier using labeledtraining vectors.

D. Classification

The classification component executes ourclassifier using input feature vectors to classify suspiciousURLs. When the classifier returns a number ofmalicious feature vectors, this component flags the corresponding URLs and their tweet information as suspicious.These URLs, detected as suspicious, will be deliveredto security experts or more sophisticated dynamicanalysis environments for an in-depth investigation.

E. Detecting Suspicious URL

In this module, we proposed a new suspicious URL detection system for Twitter, called WARNINGBIRD. Unlike the conventional systems, WARNINGBIRD is robust when protecting against conditional redirection, because it does not rely on the features of malicious landingpages that may not be reachable. Instead, it focuses on the correlations of multiple redirect chains that share the same redirection servers. We introduced new features on the basis of these correlations, implemented a near real-time classification system using these features, and evaluated the system’s accuracy and performance.

F. MailAlert

In this module, we enhance our system by providing mail alert system.Though the suspicious URLs are detected in an efficient way, it is unknown to the twitter users. Thus a MailAlert system is generated for providing an alert before the usage of the malicious URLs.

IV. Implementation

Our goal is to develop a suspicious URL detection system for Twitter that is robust enough to protect against conditional redirections. Consider a simple example of conditional redirections, in which an attacker creates a long URL redirect chain using a public URL shortening service, such asbit.lyandt.co, as well as the attacker’s own private redirection servers used to redirect visitors to a malicious landing page. The attacker then uploads a tweet including the initial URL of the redirect chain to Twitter. Later, when a user or a crawler visits the initial URL, he or she will be redirected to an entry point of the intermediate URLs that are associated with private redirection servers. Some of these redirection servers check whether the current visitor is a normal browser or a crawler. If the current visitor seems to be a normal browser, the servers redirect the visitor to a malicious landing page. If not, they will redirect the visitor to a benign landing page. Therefore, the attacker can selectively attack normal users while deceiving investigators. (Fig 3) shows the implementation framework of our proposed system. Thus shows how the benign URL and the malicious URL are classified which leads to the detection of attacker and block the malicious URL and prevents system disaster.

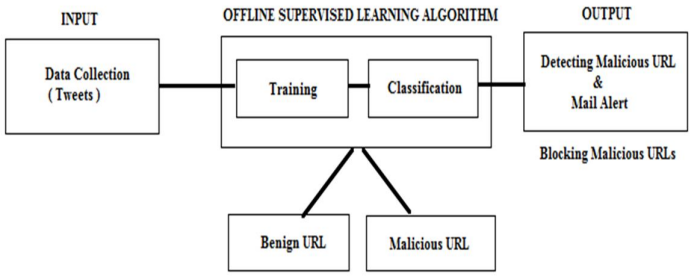

Fig. 3 Implementation framework

V. Discussions

Efficiencies, we measured the time difference betweenWARNINGBIRD's detection and Twitter's suspensionof the accounts.Among the sampled accounts, 5; 380 accountswere suspended within a day; 37:3% of them wassuspended within a minute, another 42:5% of them wassuspended within 200 minutes, and the remaining 20:7%of them was suspended within a day.

VI. Result Set

Previous suspicious URL detection systems are weak atprotecting against conditional redirection servers that distinguish investigators from normal browsers and redirectthem to benign pages to cloak malicious landing pages its disadvantage is time consuming and less detection accuracy (fig 4) our proposed system less time and high detection accuracy.

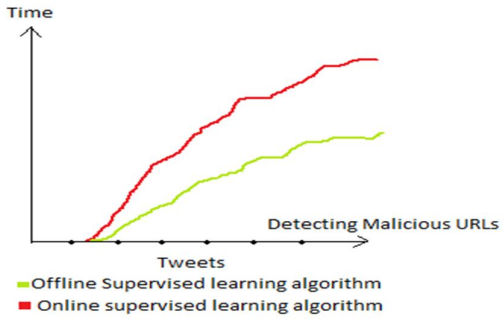

Fig. 4 Time consuming between offline and online supervised algorithm

VII. Conclusion

We proposed a new a new suspicious URL detection system for Twitter, called WARNINGBIRD. Unlike the conventional systems,WARNINGBIRD is robust when protecting against conditionalredirection, because it does not rely on the featuresof malicious landing pages that may not be reachable.Instead, it focuses on the correlations of multiple redirectchains that share the same redirection servers. We introducednew features on the basis of these correlationsimplemented a near real-time classification system using these features, and evaluated the system's accuracy and performance. The evaluation results show that oursystem is highly accurate and can be deployed system to classify large samples of tweets fromthe Twitter public timeline. Using offline supervised learning algorithm to detect the suspicious URLs in Twitter stream then immediately block that URLs and also provide alert to user through Mail. We present Malicious URLs blocker system provide high accuracy.

Our main future objective is to extend these ideas to address to address dynamic and multiple redirections.We will also implement a distributed version ofWARNINGBIRD to process all tweets from the Twitterpublic timeline.

REFERENCES

[1] S. LEE AND J. KIM, "WARNINGBIRD: DETECTING SUSPICIOUS URLS INTWITTER STREAM," IN *PROC. NDSS*, 2012.

[2]H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a socialnetwork or a news media?" in *Proc. WWW*, 2010.

[3] G. Stringhini, C. Kruegel, and G. Vigna, "Detecting spammers onsocial networks," in *Proc. ACSAC*, 2010.

[4]S. Chhabra,A. Aggarwal, F. Benevenuto, and P.Kumaraguru,"Phi.sh/$oCiaL: the phishing landscape through short URLs," in*Proc. CEAS*, 2011.

[5]F. Klien and M. Strohmaier, "Short links under attack: geographicalanalysis of spam in a URL shortener network," in *Proc. ACMHT*, 2012.

[6]F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida, "Detectingspammers on Twitter," in *Proc. CEAS*, 2010.

[7]H. Gao, Y. Chen, K. Lee, D. Palsetia, and A. Choudhary, "Towardsonline spam filtering in social networks," in *Proc. NDSS*, 2012.

