IJDCST @Feb-March-2015, Issue- V-3, I-3, SW-10
ISSN-2320-7884 (Online)
ISSN-2321-0257 (Print)

A Dynamic system for Suspicious URL detection in Twitter to recognize Multiple Redirections

Goli Madhulatha, T. Venkata Sampath Kumar

Abstract: Twitter is a famous social networking and information sharing service that allows users to exchange messages with very limited characters. 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 can contain only a restricted number of characters. Owing to the popularity of Twitter, malicious users often try to find a way to attack it. The most common forms of Web attacks, including spam, scam, 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. A number of static suspicious URL detection schemes have also been introduced in the area of detecting suspicious URLs, which cannot identify the multiple redirections of runtime effectively. In this paper, we proposed A Dynamic system for Suspicious URL detection in Twitter to recognize Multiple Redirections. Instead of investigating the landing pages of individual URLs in each tweet, which may not be successfully fetched, we considered the dynamic (run-time) correlations of URL redirect chains extracted from a number of tweets. We collect numerous tweets from the Twitter public timeline and build a statistical classifier using them. Evaluation results show that our dynamic classifier accurately and efficiently detects suspicious URLs.

Keywords: Twitter, Multiple Redirections, Dynamic URL Redirection, correlations extraction.

I. Introduction

Owing to the popularity of Twitter, malicious users often try to find a way to attack it. The most common forms of Web attacks, including spam, scam, phishing, and malware distribution attacks, have also appeared on Twitter. Many Twitter spam detection schemes have been introduced. Most have focused on how to collect a large number of spam and non-spam accounts and extract the features that can effectively distinguish spam from non-spam accounts. To detect spam accounts, some schemes manually analyze the collected data [1], [2], some use honey-profiles to lure spammers [6], [10], some monitor the Twitter public timeline to detect accounts that post tweets with blacklisted URLs and yet others monitor Twitter's official account for spam reporting, @spam [4].

When Twitter users want to share a URL with friends via tweets, they usually use URL shortening services [3] to reduce the URL length since tweets can contain only a restricted number of characters. bit.ly and tinyurl.com are widely used services, and Twitter also provides a shortening service t.co. Because tweets are short in length, attackers use shortened malicious URLs that redirect Twitter users to external attack servers. Many preliminary studies [6], [7], rely on account features including the numbers of followers and friends, account creation dates, URL ratios, and tweet text similarities, which can be efficiently collected but easily fabricated.

In this paper, we proposed a dynamic suspicious URL detection system for Twitter. Instead of investigating the landing pages of individual URLs in each tweet, which may not be successfully fetched, we considered correlations of URL redirect chains extracted from a number of tweets. Because attacker's resources are generally limited and need to be reused, their URL redirect chains usually share the same URLs. We therefore created a method to detect correlated URL redirect chains using such frequently shared URLs. By analyzing the correlated URL redirect chains and their tweet context information, we discover several features that can be used to classify suspicious URLs. We collected a large

number of tweets from the Twitter public timeline and trained a statistical classifier using the discovered features. The trained classifier is shown to be accurate and has low false positives and negatives.

II. Related work

Many suspicious URL detection schemes have been proposed. They can be classified into either static or dynamic detection systems. Some lightweight static detection systems focus on the lexical features of a URL such as its length, the number of dots, or each token it has [4], and also consider underlying DNS and WHOIS information [6 and 7]. More sophisticated static detection systems, such as Prophiler [8], additionally extract features from HTML content and JavaScript codes to detect drive-by download attacks.

Static detection systems cannot detect suspicious URLs with dynamic content such as obfuscated JavaScript, Flash, and ActiveX content. Therefore, we need dynamic detection systems that use virtual machines and instrumented Web browsers for in-depth analysis of suspicious URLs. Nevertheless, all of these detection systems may still fail to detect suspicious sites with conditional behaviors.

We consider blackraybansunglasses.com, which is a suspicious site associated with spam tweets. We used a one percent of a sample of tweets collected on Jan,2015, to conduct an in-depth analysis of the site blackraybansunglasses.com has a page, redirect.php, which conditionally redirects users to random spam pages.

We also consider a suspicious site 24newspress.net. We used one percent of the tweet samples collected on Jan, 2015, to conduct an in-depth analysis of the page. Unlike blackraybansunglasses.com, 24newspress.net does not perform conditional redirection to avoid investigation. Instead, it uses a number of IP addresses and domain names for cloaking like IP fast flux and domain flux methods.

Multiple redirections: Web pages can embed several external pages and different content. Therefore, some pages can cause multiple redirections. Because our system currently only considers HTTP redirection and does not consider page-level redirection, it cannot catch multiple redirections. Therefore, we need customized browsers to catch and address multiple redirections.

Coverage and scalability: Currently, our system only monitors one percent of the samples from the Twitter public timeline, because our accounts only have the Spritzer access role. If our accounts were to take on the Gardenhose access role, which allows the processing of 10% of the samples, our system could handle this number of samples in almost real time. The current implementation, however, cannot handle 100% of the Twitter public timeline. Therefore, we need to extend WARNINGBIRD to a distributed detection system, for instance, Monarch [9], to handle the entire Twitter public timeline.

III. Dynamic system for Suspicious URL detection in Twitter

Our goal is to develop a dynamic suspicious URL detection system for Twitter that is robust enough to protect against conditional redirections. Consider a simple example of conditional redirections (Fig. 1), in which an attacker creates a long URL redirect chain using a public URL shortening service, such as bit.ly and t.co, as well as the attacker's own private redirection servers used to redirect visitors to a malicious landing page.

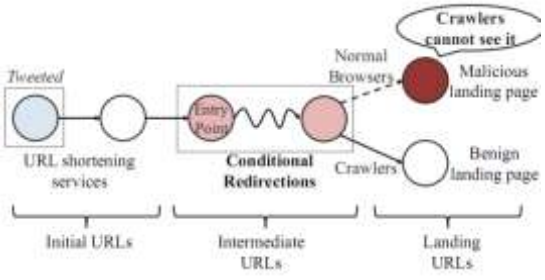

Figure 1. URL Redirection example in twitter

The above example shows that, as investigators, we cannot fetch the content of malicious landing URLs, because attackers do not reveal them to us. We also cannot rely on the initial URLs, as attackers can generate a large number of different initial URLs by abusing URL shortening services.

Proposed Dynamic System Architecture: Our system consists of four components: data collection, feature extraction, training, and classification.

Data collection: 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.

Feature extraction: 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 determine whether a sufficient number of tweets have been collected. Specifically, our system uses a tweet window instead of individual tweets. When more than w tweets are collected, it pops w tweets from the tweet queue. First, for all URLs in the w tweets, this component checks whether they share the same IP addresses.

Training: The training component has two subcomponents: retrieval of account statuses and training of the classifier. Because 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 denied.

Classification: The classification component executes our classifier using input feature vectors to classify suspicious URLs. When the classifier returns a number of malicious feature vectors. This component flags the corresponding URLs and tweet information as suspicious. These URLs, detected as suspicious, will be delivered to security experts or more sophisticated dynamic analysis environments for an in-depth investigation.

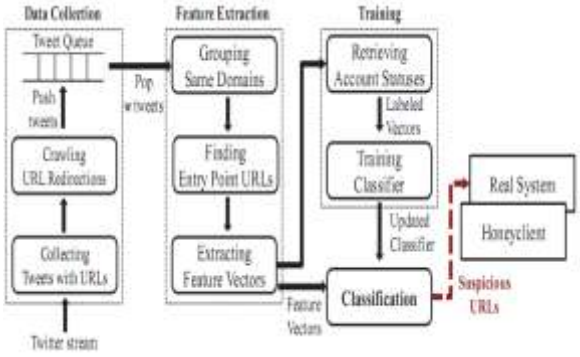

Fig 2.Multi Redirection suspicious URL detection

IV. Experiments

Our system consists of two Intel Quad Core Xeon E5530 2.40GHz CPUs and 24 GB of main memory. To collect the tweets, we used Twitter Streaming APIs[5, 11]. Our accounts have a Spritzer access role, and thus we can collect about one percent of all tweets from the Twitter public timeline as samples. Our system visited all the URLs in the tweets to collect the URL redirect chains.

We used sample tweets collected between September 2011 and October 2011 to train the classification models and sample tweets collected during August 2011 and during November 2011 for testing the classifier using older and newer datasets, respectively. From the training dataset, we found 183,846 entry point URLs that appeared more than once in every 10,000 consecutive sample tweets. Among them, 156,896 entry point URLs were benign and 26,950 entry point URLs were malicious.

We also used two test datasets representing past and future values to evaluate the accuracy of our classifier. Regardless of whether the test datasets represented past or future values, our classifier achieved a relatively high accuracy, and few false positives and false negatives. As a result, we concluded that our features could endure about one month time differences.

We used the F-score to evaluate and compare the features of our scheme [11]. The F-score of a feature represents its degree of discrimination. Features with large F-scores can split benign and malicious samples

better than features with small F-scores. The variations of F-scores and average feature values show that we need to periodically update our classifiers to cope with continuously changing circumstance of Twitter.

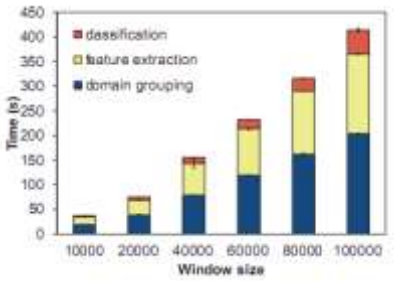

Fig 3. Time allocation chart of main process execution

V. Conclusion

Conventional suspicious URL detection systems are in-effective in their protection against conditional redirection servers that distinguish investigators from normal browsers and redirect them to benign pages to cloak malicious landing pages. In this paper, we proposed a new dynamic suspicious URL detection system for Twitter. Unlike the conventional systems, our system is robust when protecting against conditional redirection, because it does not rely on the features of malicious landing pages 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.

VI. References

[1] S. Lee and J. Kim, "WarningBird: Detecting suspicious URLs in Twitter stream," in Proc. NDSS, 2012.

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

[3] D. Antoniades, I. Polakis, G. Kontaxis, E. Athanasopoulos, S. Ioan-nidis, E. P. Markatos, and T. Karagiannis, "we.b: The web of short URLs," in Proc. WWW, 2011.

[4] D. K. McGrath and M. Gupta, "Behind phishing: An examination of phisher modi operandi," in Proc. USENIX LEET, 2008.

[5] Z. Chu, S. Gianvecchio, H. Wang, and S. Jajodia, "Who is tweeting on Twitter: Human, bot, or cyborg?" in Proc. ACSAC, 2010.

[6] G. Stringhini, C. Kruegel, and G. Vigna, "Detecting spammers on social networks," in Proc. ACSAC, 2010.

[7] C. Grier, K. Thomas, V. Paxson, and M. Zhang, "@spam: The underground on 140 characters or less," in Proc. ACM CCS, 2010.

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

[9] F. Klien and M. Strohmaier, "Short links under attack: geograph- ical analysis of spam in a URL shortener network," in Proc. ACM HT, 2012.

[10] K. Lee, J. Caverlee, and S. Webb, "Uncovering social spammers: Social honeypots + machine learning," in Proc. ACM SIGIR, 2010.

[11] A. Wang, "Don't follow me: Spam detecting in Twitter," in Proc.SECRYPT, 2010.