iJRASET
International Journal For Research in Applied Science and Engineering Technology

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

**Volume:** 2 **Issue:** VII **Month of publication:** July 2014

**DOI:**

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

# Warning Tweet: A Detection System for Suspicious URLs in Twitter Stream

Manjeet Chaudhary1, Prof.H.A Hingoliwala2

*1M.E Computer Science*

*2 Prof. ,Computer department*

*JSPM college*

***Abstract—Twitter is a social networking site where users can exchange messages to other users particularly their followers. Usually the messages sent over twitter are known as tweets. Users can sent messages or tweets to users who do not follow the sender .Tweets are small messages. Thus malicious users can use twitter to send malicious tweets containing malicious URL's for spam or phishing etc. Conventional there are twitter spam detection techniques which uses features like ratio of tweets or date of creation of account but these techniques are ineffective against feature fabrications and consume much time and resources. In this paper, WARNINGTWEET a technique for detecting suspicious URL detection for Twitter is proposed. This system find the correlations of URL redirect chains extracted from several tweets. It uses the fact that the malicious users or attackers have limited resources and thus they need to reuse them. URL redirect chains frequently share the same URLs for the attackers or malicious users. Methods to discover correlated URL redirect chains using the frequently shared URLs and to determine their suspiciousness is developed. On the basis of the results of evaluation we find that our classifier worked accurately and efficiently detects suspicious URLs.***

## INTRODUCTION

Twitter is a social networking Site used to share information between users. Users can send tweets to its followers, to a particular user and also to users who are not the followers of the sender. Twitter tweets can contain only a restricted number of characters thus twitter uses URL shortening services to reduce URL length. As twitter is a famous site so malicious users try to attack it like web attack, spam, scam, phishing . As Tweets are short in length, attackers use shortened malicious URLs that redirect Twitter users to external attack servers

WARNINGTWEET, a 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.

A new suspicious URL detection system for Twitter that is based on the correlations of URL redirect chains, which are difficult to fabricate. The system can find correlated URL redirect chains using the frequently shared URLs and determine their suspiciousness in almost real time. We introduce new features of suspicious URLs: Some of which are newly discovered and while others are variations of previously discovered features. . We present the results of investigations conducted on suspicious URL's.

ISSN: 2321-9653



LITERATURE SURVEY

| Literature Survey | | |
|---|---|---|
| Name/Founder | Features | Disadavatages |
| Twitter | Detection based on informatioa bou account | Time consuming and can easily be fabricated |
| Don't Follow me | Detection based on informatioa bou account | Easily fabricated |
| WarningBird | Detection based on correlated URL's | Cannot detect all types of SPAM and time consuming |
| Lee and Song | Sender and receiver relationship and twitter graph | Time and resource consuming |
| Phishing detection | Uses twitter content and user details | Not able to detect suspicious URl's and less browser compatible |

Table 1: Literature Survey

IMPLEMENTATION

MOTIVATION AND BASIC IDEAS

We have developed a system for twitter which helps in detecting suspicious URL's. This system should be developed such that it protect against conditional redirections. Consider a example of conditional Redirections(Fig.1).In this example the malicious user creates a long URL redirect chain using a public URL shortening service 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

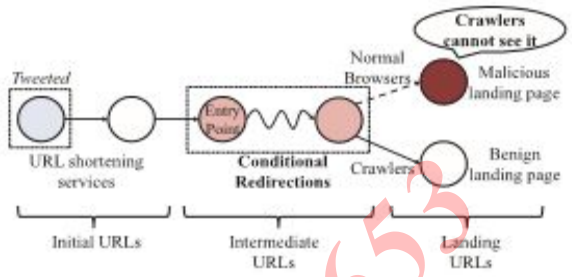



Fig. 1.Conditional redirections

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. 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. Attackers may reuse some of their redirection servers when creating their redirect chains because they do not have infinite redirection servers. Therefore, if we analyze several correlated redirect chains instead of an individual redirect chain, we can find the entry point of the intermediate URLs in these chains.

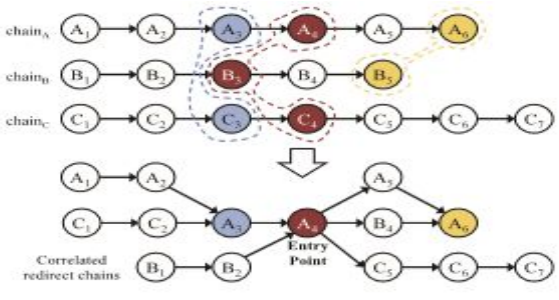



Fig. 2.Conditional redirections

Consider the three redirect chains shown in the top half of Fig. 3, which share some URLs: A3 = C3, A4 = B3

ISSN: 2321-9653



=C4, andA6 = B5. By combining the three redirect chains using these shared URLs, we can generate the correlated redirect chains (the bottom half of Fig. 3) that share the same entry point URL, A4 (because A4 is the most frequent URL in these chains). The correlated redirect chains show that the entry point has three different initial URLs and two different landing URLs, and participates in redirect chains that are six to seven URLs long. These are the characteristics of the suspicious URL's. Even the entry point, A4, does not allow our crawler to visit the latter URLs, we could infer that the chains are suspicious because it has many initial URLs for the same landing (entry point in reality) URLs. Therefore, this correlation analysis can help in detecting suspicious URLs even when they perform conditional redirections.

System details

Our system consists of four components: data collection, feature extraction, training, and classification (Fig. 3).

System details

Our system consists of four components: Collect which consists of data collect phase. Next phase is Extraction phase which helps in grouping same domains and finding entry point URL's. Third phase is Training phase which helps in getting account statuses and training classifiers and the last phase is the classification phase which help distinguishing the malicious and benign URL' s or tweets.(Fig. 3).

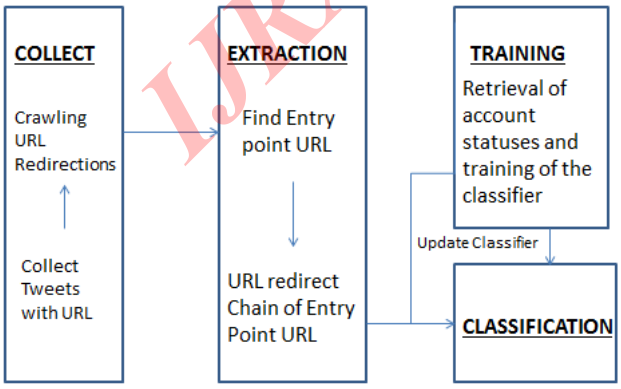



Fig. 3. System Details

COLLECT. 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. 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.

EXTRACTION:. The feature extraction component has two subcomponents: grouping of identical domains, finding entry point URLs. First, for all URLs in the w tweets, this component checks whether they share the same IP addresses. If several URLs share at least one IP address, it replaces their domain names with a list of domains with which they are grouped.This grouping process enables the detection of suspicious URLs that use several domain names to bypass the blacklisting.

This component tries to find the entry point URL for each of the w tweets. First, it measures the frequency with which each URL appears in these tweets. It then discovers the most frequent URL in each URL redirect chain in the w tweets. The discovered URLs, thus, become the entry points for their redirect chains. If two or more URLs share the highest frequency in a URL chain, this component selects the URL nearest to the beginning of the chain as the entry point URL. Finally, for each entry point URL, the component finds URL redirect chains that contain the entry point URL

Finally, for each entry point URL, the component finds

URL redirect chains that contain the entry point URL, and

extracts various features from these URL redirect chains

These feature values are then turned into real-valued feature vectors.

ISSN: 2321-9653



TRAINING:. The training component has two subcomponents: Retrieval of account statuses and training of the classifier. URLs from suspended accounts are considered malicious, whereas URLs from active accounts are considered benign.

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 their 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.

Features

*URL redirect chain length.* Attackers usually use long URL redirect chains to make investigations more difficult and avoid a dismantling of their servers. Therefore, when an entry point URL is malicious, its chain length l may be longer than those of benign URLs.

*Frequency of entry point URL*. The number of occurrences of the current entry point URL within a tweet window is important. Frequently appearing URLs that are not whitelisted are usually deemed suspicious If the size of a tweet window is w and an entry point URL appears n times in the windows, this feature can be computed as n/w

*Relative position of an entry point URL*. Suspicious entry

point URLs are not usually located at the end of a redirect

chain because they have to conditionally redirect visitors to

different landing URLs. Their positions are relative to the

lengths of their redirect chains. Therefore, if the position of

an entry point of a redirect chain of length l is p, this feature

can be computed as p/l

*Relative number of different initial URLs.* :The initial URL is

the beginning URL that redirects visitors to the current entry point URL. Attackers usually use a large number of different initial URLs to make their malicious tweets, which redirect visitors to the same malicious URL, look different. The number of different initial URLs cannot exceed the number of times that their entry point URLs appear. Therefore, if the number of different initial URLs redirecting visitors to an entry point URL that appears n times is i, this feature can be computed as i/n

*Number of different landing URLs*. If the current entry point

URL redirects visitors to more than one landing URLs, we can assume that the current entry point URL performs conditional redirection activities and may be suspicious. Unlike the initial URLs, we use an absolute number of different landing URLs as a feature because the existence of more than one landing URL is a suspicious sign regardless of the frequency of the entry point URL

*Numbers of different domain names and IP addresses*. Some spam sites use a large number of domain names and IP addresses to avoid blacklisting. Therefore, we use the number of different domain names and the number of different IP addresses of the entry point URLs as features.

EVALUATION

System SetUp and Data Collection

Our system consists of two Intel Quad Core Xeon E5530 2.40-GHz CPUs and 24 GB of main memory. To collect the tweets, we used Twitter Streaming APIs . 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. From April 8 to December 8,

ISSN: 2321-9653



2011 (245 days in total), we collected 59,056,761 samples of tweets with URLs. We observed about 240,000 tweets daily, on average. Our system visited all the URLs in the tweets to collect the URL redirect chains. In addition, starting on July 23, our system collected the IP addresses of all URLs for the domain grouping. From the collected tweets, we found 13,261,069 unique Twitter accounts. Among them, 1,339,496 accounts (10.1 percent) were suspended as of January 15, 2012. Twitter announced that it had started to wrap URLs with lengths longer than 19 characters using its URL shortening service t.co from August 15, 2011, and that it started to wrap all URLs regardless of their length from October 10, 2011. We noticed that this additional layer of URL redirections affects our classification results; therefore, from August 15, 2011, we decided to remove the first t.co URLs in redirect chains.

## LABELING THRESHOLD

Labeling is essential for classification. Unfortunately, we were unable to find a suitable source for labeling our data sets, as many of the URLs in our data sets have not been listed on a public URL blacklist, such as the Google Safe Browsing API . Therefore, instead of URL black- lists, we used Twitter account status information to label our data sets. That is, if some accounts had posted the same URLs and Twitter suspended the accounts later, we regarded the URLs as malicious. Otherwise, we regarded them as benign. Since we rely on the results of Twitter's spam account detection system to label the collected data sets, one can argue that it just mimics the Twitter's detection system at most. However, most of our features are independent of the Twitter's rules .Twitter can know whether an account violates the rules or not only after the account have performed a series of activities. However, unlike the rules, we focus on the characteristics of URL redirect chains and the similarity of a group of users who uploaded the same URL redirect chains; our system can immediately check them. We also verified that our system can detect suspicious accounts that Twitter cannot detect even several days later . Therefore, we can say that our system is not a simple mimic of the Twitter's detection system. Because the Twitter's detection system had a time delay for suspicious account detection, we checked the status information of accounts at least one month later from their posting of tweets. The remaining challenge is that of how to reduce the possibility of false labeling. For instance, let us assume that 30 suspended accounts and 20 active accounts distribute the same URL U1, and the other 20 suspended accounts and 30 active accounts distribute the same URL U2. Can we be assured that U1 is malicious and U2 is benign? If we treat a URL as malicious if at least one suspicious account posted it,3 we can capture many suspicious URLs but false positives increase. In contrast, if we treat a URL as benign if at least one active account posted it, we can reduce false positives, but we could miss many real suspicious URLs. Moreover, unlike suspended accounts, we could not guarantee that all of the active accounts are not spam accounts because the Twitter's detection system is not perfect. To solve this problem, we need to define a reasonable threshold value that decides whether a URL is malicious or benign. We used tweets collected between July 23 and August 8, 2011 to ascertain the threshold value. For each group of accounts that distributed the same entry point URLs, we determined what portion were suspended accounts. Approximately 76 percent of the entry point URLs had no relationship with the suspended accounts while another 13 percent of them were distributed solely by suspended accounts. We, thus, need to assess the remaining (approximately) 11 percent of entry point URLs. Fig. 8 shows that the CDF sharply increases when the fraction of suspended accounts is 50 percent. Therefore, intuitively, we can assume that 50 percent is a good candidate. To verify this assumption, we trained an L1- regularized logistic regression algorithm with the data set, where its labels were determined according to the fractions of suspended accounts. We then checked the false-positive and false-negative rates within 10-fold cross validation (Fig. 9). The false-positive rates slightly increased according to the fraction of suspended accounts. In contrast, the false-negative rates substantially decreased in accor- dance with the fraction of suspended accounts, especially

ISSN: 2321-9653



when the fraction was 50 percent. As a result, we choose 50 percent as the value for the labeling threshold.

## TRAINING AND TESTING CLASSIFIERS

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 data sets, respectively. From the training data set, 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 the account status information to label the test data set; the results are shown in Table 2. We used the LIBLINEAR library to implement our classifier. We compared seven classification algorithms, and selected an L2-regularized L1-loss support vector classifica- tion (SVC) algorithm, because it shows the highest AUC and the lowest FP with the training data set, experimen- tally.4 Table 3 shows the results; here, LR is an abbreviation of logistic regression, SVC is support vector classification, AUC is an area under the ROC curve, FP is false positive,

| Dataset | Period | Benign | Malicious | Total |
|---|---|---|---|---|
| Training | September | 78, 982 | 14, 885 | 93, 867 |
| | October | 77, 914 | 12, 065 | 89, 979 |
| Total | | 156, 896 | 26, 950 | 183, 846 |
| $Test_{past}$ | August | 89, 543 | 21, 368 | 110, 911 |
| $Test_{future}$ | November | 81, 742 | 13, 132 | 94, 874 |

Table 2: Training and Test Data Sets

FN is false negative, L1R and L2R are L1- and L2-regularized, and primal and dual represent functions that determine termination of training. Standard deviations of the AUC were 0.0029-0.0032, those of the accuracy were 0.17-0.20 percent, those of the FP were 0.05-0.09 percent, and those of the FN were 0.18-0.19 percent. We could further reduce the value of the FP by increasing the weight value of the benign samples to penalize them (because the number of benign samples is fairly larger than the number of malicious samples). We used a weight value of 1.1 for benign samples; finally, we obtained 0.95 percent FP, 7.33 percent FN, 91.71 percent accuracy, and 0.9027 AUC. All the training and 10-fold cross validation could be done in less than several seconds in our system. Therefore, the training time is negligible. We also used two test data sets representing past and future values to evaluate the accuracy of our classifier (Table 2). Regardless of whether the test data sets represented past or future values, our classifier achieved a relatively high accuracy, and few false positives and false negatives (Table 4). As a result, we concluded that our features could endure about one month time differences.

## FEATURE COMPARISION AND VARIATIONS

We used the F-score to evaluate and compare the features of our scheme . 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. Although the F-score does not reveal the mutual information between features, it is simple and quite effective in many cases. The F-scores show that the similarity of the account creation dates, the relative number of source applications, and the relative number of initial URLs are important features (Table 4). We also verified that the similarity of the number of friends and followers, and the relative number of Twitter accounts are less important because attackers can manipulate the number of their friends and use a large number of bot accounts to distribute URLs. Moreover, we noticed that the number of different domain names is not important because we had already grouped domain names that share the same IP addresses

| | | % | | |
|---|---|---|---|---|
| Dataset | AUC | Accuracy | FP | FN |
| $Test_{past}$ | 0.8937 | 88.00 | 0.83 | 11.16 |
| $Test_{future}$ | 0.8960 | 91.53 | 1.23 | 7.24 |

Table 3 :Classification accuracy of test data sets

ISSN: 2321-9653



| Feature | F-score |
|---|---|
| Similarity of account creation dates | 0.1360 |
| Relative number of sources | 0.1278 |
| Relative number of initial URLs | 0.0659 |
| Similarity of tweet texts | 0.0547 |
| Similarity of follower-friend ratios | 0.0531 |
| Frequency of entry point URL | 0.0393 |
| Number of landing URLs | 0.0277 |
| URL redirect chain length | 0.0142 |
| Number of IP addresses | 0.0091 |
| Relative position of entry point URL | 0.0043 |
| Similarity of the number of friends | 0.0029 |
| Similarity of the number of followers | 0.0008 |
| Number of domain names | 0.0006 |
| Number of accounts | 0.0005 |

Table 4: F scores of training data set

Running Time

We evaluated the running time of our system. First, we compared the running time of each component of the system—domain grouping; feature extraction, including the detection of entry points; and classification—in a single window of collected tweets that varied in size. Even if the window size grew to 100,000, which can contain of about 10 percent of all tweets with URLs per hour, the running time was only 6.9 min Next, we estimated the time required to classify a single URL. Our system currently uses 100 crawling threads to concurrently visit URL redirect chains; on average, each thread requires 2.42 s to visit a single URL redirect chain. For a window size of 100,000, we needed 28.309 ms to process a single URL (Table 6)— indicating that our system can process about 127,000 URLs per hour. Therefore, our system can handle 10 percent of the tweet samples, the level provided by the Gardenhose access role, in real time. By increasing the number of crawling threads, we can process more than 10 percent of the tweet samples. For instance, if we use 1,000 crawling threads, we can process about 576,000 URLs per hour. Even if we do this, the current implementation cannot process all the tweets, because we would have to process a single URL in less than 3.6 ms to handle 1,000,000 URLs per hour.

COMPARISION

We compared the efficiency of WARNINGTWEET with that of Twitter's detection system and WARNING BIRD. For the comparison, we sampled 14,905 accounts detected by our online WARNINGTWEET

To compare their efficiencies, we measured the Time difference between WARNINGTWEET Twitter's suspension of the accounts. We monitored the WARNINGTWEET to obtain newly detected suspicious accounts and then checked the status of each account every 15 s, for one day, until it was suspended. Among the sampled accounts, 5,380 accounts were suspended within a day; 37.3 percent of them were suspended within a minute, another 44.3 percent of them were suspended within 4 hours, and the remaining 18.4 percent of them were suspended within a day The average time difference was 13.5 min, which shows that our detection system is more efficient than that of Twitter.

Among the 14,905 accounts, Twitter had suspended 9,250 accounts. We then randomly selected 500 accounts from the remaining 5,655 active accounts to manually check how suspect they were. Among the 500 accounts, 320 accounts were suspicious. Therefore, the detection accuracy of our system given the sample data is about 86.3 percent.

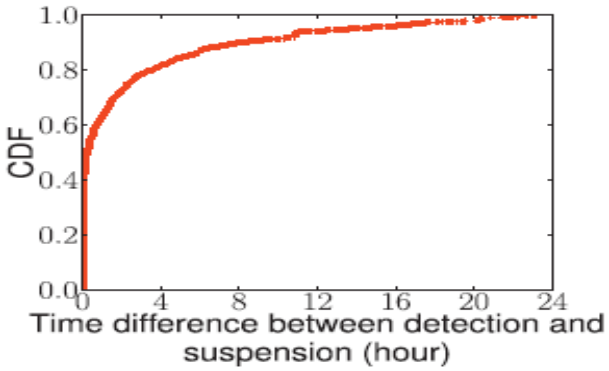



Fig. 5. Time difference between WarningTWEET detection of suspiciou s accounts and Twitter's suspension within a day

CONCLUSION

ISSN: 2321-9653



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 suspicious URL detection system for Twitter, called WARNINGTWEET. Unlike the conventional systems, WARNINGTWEET 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. The evaluation results show that our system is highly accurate and can be deployed as a near real-time system to classify large samples of tweets from the Twitter public timeline. In the future, we will extend our system to address dynamic and multiple redirections. We will also implement a distributed version of WARNINGTWEET to process all tweets from the Twitter public timeline

## FUTURE SCOPE

WARNINGTWEET need to be adapted to other services like Facebook and LinkedIn The scalibilty needs to be enhanced Dynamic and Multiple redirections can be incorporated into the algorithm

## REFERENCES

[1] S. Lee and J. Kim, "WarningBird: Detecting Suspicious URLs in Twitter Stream," Proc. 19th Network and Distributed System Security Symp. (NDSS), 2012.

[2] H. Kwak, C. Lee, H. Park, and S. Moon, "What Is Twitter, a Social Network or a News Media?" Proc. 19th Int'l World Wide Web Conf. (WWW), 2010.

[3] D. Antoniades, I. Polakis, G. Kontaxis, E. Athanasopoulos, S. Ioannidis, E.P. Markatos, and T. Karagiannis, "we.b: The Web of Short URLs," Proc. 20th Int'l World Wide Web Conf. (WWW), 2011.

[4] D.K. McGrath and M. Gupta, "Behind Phishing: An Examination of Phisher Modi Operandi," Proc. First USENIX Workshop Large- Scale Exploits and Emergent Threats (LEET), 2008.

[5] Z. Chu, S. Gianvecchio, H. Wang, and S. Jajodia, "Who Is Tweeting on Twitter: Human, Bot, or Cyborg?" Proc. 26th Ann. Computer Security Applications Conf. (ACSAC), 2010

[6] G. Stringhini, C. Kruegel, and G. Vigna, "Detecting Spammers on Social Networks," Proc. 26th Ann. Computer Security Applications Conf. (ACSAC), 2010.

[7] C. Grier, K. Thomas, V. Paxson, and M. Zhang, "@spam: The Underground on 140 Characters or Less," Proc. 17th ACM Conf. Computer and Comm. Security (CCS), 2010.

[8] S. Chhabra, A. Aggarwal, F. Benevenuto, and P. Kumaraguru, "Phi.sh/$oCiaL: the Phishing Landscape through Short URLs," Proc. Eighth Ann. Collaboration, Electronic Messaging, Anti-Abuse and Spam Conf. (CEAS), 2011.

[9] F. Klien and M. Strohmaier, "Short Links under Attack: Geographical Analysis of Spam in a URL Shortener Network," Proc. 23rd ACM Conf. Hypertext and Social Media (HT), 2012.

[10] K. Lee, J. Caverlee, and S. Webb, "Uncovering Social Spammers: Social Honeypots þ Machine Learning," Proc. 33rd Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, 2010.

[11] A. Wang, "Don't Follow Me: Spam Detecting in Twitter," Proc. Int'l Conf. Security and Cryptography (SECRYPT), 2010.

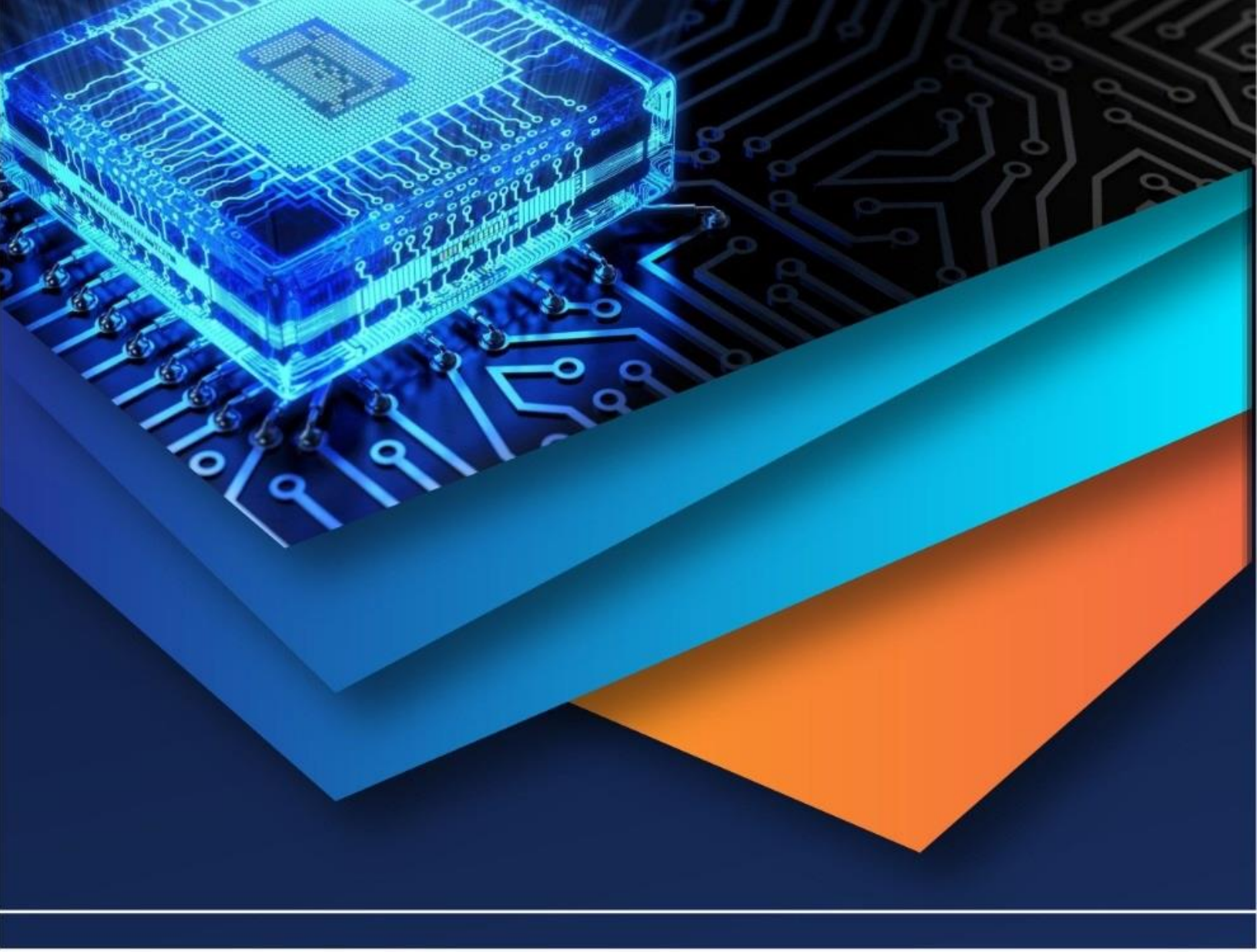
[12] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida, "Detecting Spammers on Twitter," Proc. Seventh Collaboration, Electronic Messaging, Anti-Abuse and Spam Conf. (CEAS), 2010.

[13] J. Song, S. Lee, and J. Kim, "Spam Filtering in Twitter Using Sender-Receiver Relationship," Proc. 14th Int'l Symp. Recent Advances in Intrusion Detection (RAID), 2011.

ISSN: 2321-9653



[14] C. Yang, R. Harkreader, and G. Gu, "Die Free or Live Hard? Empirical Evaluation and New Design for Fighting Evolving Twitter Spammers," Proc. 14th Int'l Symp. Recent Advances in Intrusion Detection (RAID), 2011. [15] H. Gao, Y. Chen, K. Lee, D. Palsetia, and A. Choudhary, "Towards Online Spam Filtering in Social Networks," Proc. 19th Network and Distributed System Security Symp. (NDSS), 2012.N. Kawasaki, "Parametric study of thermal and chemical nonequilibrium nozzle flow," M.S. thesis, Dept. Electron. Eng., Osaka Univ., Osaka, Japan, 1993.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY