Tweet Analysis for Malicious Content using Hybrid System

Aditya Jadhav[1] Shital Salve[2]
[1,2]Department of Computer Engineering
[1,2]Savitribai Phule University of Pune, Modern Education Society College of Engineering, Pune-01

Abstract— Spamming is common in various types of automated communication sources including email, instant messaging, and social networks. To reach more users numerous spammers now use various content-sharing platforms including online social networks— to publicize spam such as twitter. Twitter has become one of the furthermost used social networks. And, as happens with every common media, it is susceptible to misuse. In this environment, spam in Twitter has developed in the last years, becoming a significant problem for the users. In the last years, numerous methods have appeared that are able to determine whether a user is a spammer or not. However, these debarring systems cannot filter every spam message (snippet) and a spammer may make another account and restart sending spam. As there are methods to filter spam text contents, Spammers now using the idea of spam images? So a proficient spam image filter is must. These spam messages contains phish links, so here we need phish link filter. By keeping eye on this problems of twitter, we proposed here a "Twitter Spam Filter" an application to filter spam images, unsolicited tweets and phish links from tweets.

Key words: Phishing, Information Security, Machine Learning, Classifier

I. INTRODUCTION

Twitter being an important media of communications and building relationship on Internet. It is generally being influenced by many factors some of those are, spam images posted on twitter, so the filtering of these spam images is must. The concept of image-based spam is a trick introduced by spammers few years ago. It consists in merging (embedding) all the textual information (i.e. Spam message) to an image attached and posted over twitter. So as to overcome this problem OCR-based modules are proposed against image spam.

Tweets with malicious URLs may have authorized content in the body which are not possible to be detected by content based spam filters. The URLs lead to the actual Phishing sites which are fake of authorised websites and trap the users into entering subtle information. 'Hook' are the actual phishing websites which obtains the private information from the user. The malicious user shows various critical situations such as account suspension, unsuccessful transaction and force user to upgrade the newly installed security feature. The links in the tweets leads to fake phishing Site referred as 'Catch'.

II. PROPOSED SYSTEM

In day today world security is of main concern in the digitalized world of web. There being many hackers' and various attacks through social media. And concern to this there are numerous systems which provide security on various platforms such as E-mail security. But unlike all other security parameter's there is no security developed on social network as twitter. So our system projected Integration of various algorithms for spam and malicious detection on considering Social media (i.e. Twitter).Our system classifies the messages into ham and spam. It makes the efficient use of the various algorithms to provide maximum of correctness in the result.

A. Goals And Objectives

- First we aim at developing a spam Image filter by using Naïve bayes classifier.
- Then we work on Phish links filter using link guard algorithm.
- Next we aim at filtering the unwanted tweets posted by users.

III. URL ANALYSER

Based on the lexical features and host based features of the URL, phishing URLs are analysed. The lexical feature analyses the arrangement of the URL. URLs contain the host name and the link. For example, consider 'www.annauniv.edu/emmrc/emmrc.html', the host name is 'www.annauniv.edu' and 'emmrc/emmrc. html' is the path. The proposed method analyse host based points such as Page rank and how old the domain is, various lexical based features such as URL encoding, containing conscious characters, hexadecimal character or malicious IP addresses to keep out of site them. Then it analyses the word probabilities to determined if the URL contain curious content to avoid end users falling by phishing attacks as described in Fig 1. It is useful as illegitimate users fake their identities, pass tests for authentication and by avoiding spam keywords it may escape content analysis. Some emails may contain only malicious contents or links without any message in the body, urging the user to enter it thus leading them to fraudulent websites.

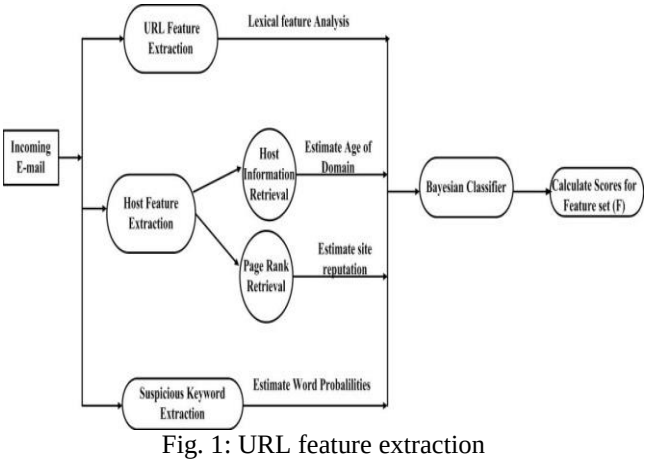


Fig. 1: URL feature extraction

Lexical features analyses the arrangement of the URL. It having the host name size, URL length size, containing of total dots, indulgence of conscious characters such @ symbols, hexadecimal characters and various special binary character such as '.', '=', '$', '^' and etc. Either in the host name or path name. Actual URLs are hidden by

fake IP addresses and hexadecimal characters. The URL can also be given by hexadecimal base values with a '%' symbol. Spoof guard (Neil Chou et al 2004) identified the @ and - symbol most prominent in phishing URLs. A @ symbol in a URL will enable the URL to the right to enter into the phishing site and discard the URL at left which is a legitimate URL. Consider the URL "http://www.sbibank.com@phishingsite.com" will enter into "phishingsite.com" and discards "www.sbibank.com". These kinds of concepts use the actual phishing website to pose as legitimate sites and disguise itself.

A. Host Based Features

The location, owner and how malicious sites are hosted and managed are identified by Host based features. Some of the features are listed below

1) Age Of Domain (F2)

The spontaneous moment at which malicious web sites are hosted such that they have less age or relatively new to obtain the user credentials is determined by Age of domain (F2). They will be newly registered and some domains may not be provided still at the instance of inspection. It obtains the data in the number of months and some may be in years more recently. The WHOIS is used to evaluate the domain registration date, and if the domain registration entry is not exist on the WHOIS server, this feature will simply return -1, to victim.

2) Page Rank (F3)

Page rank determines rank of webpage and decide higher the page rank, if it is very important. Definitely phishing WebPages having less age of domain. Therefore it determine a very degrade page rank or page rank doesn't available.

B. Number of Sensitive Words in URL

1) Individual occurrences (F4) and Co-Occurrences of suspicious phishing keywords (F5)

Garera et al (2007) used a set of eight sensitive words such as Secured, Accounts, Updating, Login, signup, financial, confirmation and decide that frequently appeared in phishing URL. The modules are trained with 1000 phishing examples to give weights to the suspicion words determined in the phishing tweets content.

The total occurrence of most occurring words includes Secured, Accounts, Updating, Login, signup, financial, confirmation and Notify, Click, Inconvenient, password etc and their Co-Occurrences the tweets contents.

C. Approach -Bayes Classifier

Bayes classifier is used in spam filters such that particular features of URLs are distributed not depending on the values of other features. Bayes theorem is used to evaluate the probability of hypothesis for the event B, provided with the training data A,

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

It is often easier to calculate the probabilities, P(A|B), P(A), P(B) for the probability that P(B|A) is required. Extrapolating Bayes rule, assume that legitimate and phishing websites occur equal in number and hence with equal probability, then the posterior probability that the feature vector X belongs to a malicious URL is such that

$$P(B = 1|A) = \frac{P(A|B=1)}{P(B|A=1)+P(B|A=0)}$$

$$P(B|A) = \frac{P(A|B)}{P(A|B)+P(A|B')}$$

$$P(B|A) = \frac{P(A|B)P(B)}{P(A|B)P(B)+P(A|B')P(B')}$$

where, P(A) = Probability of feature F in phishing and legitimate dataset.
P(B')=Legitimate dataset.
P(B)=Phishing Dataset
P(B(Phishing)) = P(B'(Legitimate)) = 0.5

The classifier has a training dataset of malicious phishing URLs and legitimate URLs. The probability occurrence of each feature in the dataset are calculated and their respective scores are obtained (i.e.) Count up occurrence of features in the dataset and calculate the cumulative score. If Cumulative score > Threshold, consider as phishing URL else legitimate URL as illustrated in Fig2.

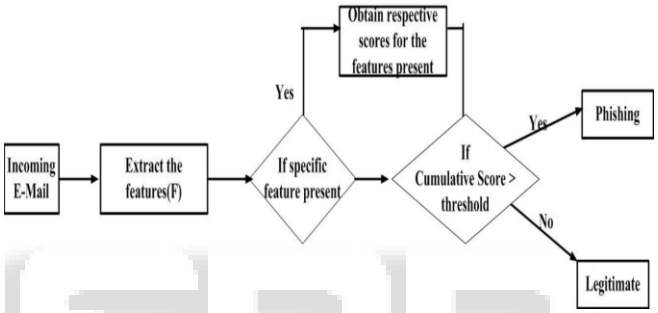


Fig. 2: Phishing URL classifications

a) How many times does feature F(F1,F2,F3,F4,F5,F6) appear in phishing dataset?
b) How many times does feature F(F1,F2,F3,F4,F5,F6) appear in legitimate dataset?

Let F1 = Lexical features ,F2 = Age of the domain factor of URLs, F3 = Occurrence of Page rank< 3 in phishing and legitimate dataset, F4 = Individual Occurrence of suspicious keywords, F5= Co–Occurrences of suspicious keywords, F6 = Login Form detection

IV. PHISHING ALGORITHM

1) Get the hyperlink for verification.
2) Extract the hypertext and anchor text. Check that both are same or not if not then alert the user.
3) If the hyperlink contains any input address, then check the IP Blacklist and IP Whitelist. If IP address found in Blacklist then alert the user and if IP address found in Whitelist then user is safe.
4) If the hyperlink is an encoded one, then the Phish Link detection algorithm will detect it, decode it and then will inform the user.
5) If the hyperlink is shortened then alert the user.
6) Check the domain name of URL in Whitelist and Blacklist and then alert the user respectively.

V. LINK GUARD ALGORITHM

An end-host based anti-phishing algorithm which we call Link Guard, based on the characteristics of the phishing hyperlink. Since Link Guard is character-based, it can detect and prevent not only known phishing attacks but also

unknown ones. We have implemented Link Guard in Windows XP, and our experiments indicate that Link Guard is light-weighted in that it consumes very little memory and CPU circles, and most importantly, it is very effective in detecting phishing attacks with minimal false negatives. Link Guard detects 195 attacks out of the 203 phishing archives provided by APWG without knowing any signatures of the attacks

VI. MORPHISM

Morphism refers to a structure-preserving mapping from one mathematical structure to another. The notion of morphism recurs in much of contemporary mathematics. In set theory, morphisms are functions in linear algebra, linear transformations; in group theory, group homeomorphisms; in topology, continuous functions, and so on.

Morphisms an abstraction derived from structure-preserving mapping between two mathematical structures.

A. Description

There are two processes which are defined on every morphism, the domain (or source) and the co-domain (or target).

If a morphisms has domain X and co-domain Y, we write f: X → Y. Thus a morphism is represented by an arrow from its domain to its co-domain. The collection of all morphisms from X to Y is denoted hom(X, Y) and called the hom-set between X and Y.

For every three objects X, Y, and Z, there exists a binary operation hom(X, Y) × hom(Y, Z) → hom(X, Z) called composition . The composite of f: X → Y and g: Y → Z is written g∘f or gf. The composition of morphisms is often represented by a commutative diagram.

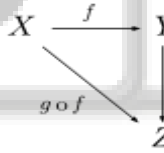


Fig. 5: Commutative Diagram

Morphisms satisfy two axioms:

Identity: for every object X, there exists a morphism IDs: X → X called the identity morphism on X, such that for every morphism f: A → B we have idB∘f = f = f∘idA.

Associativity : h∘ (g∘f) = (h∘g) ∘f whenever the operations are defined.

B. Describing the Morphism In Our System

In our proposed work we are creating a filtering function to filter different type of things from tweeter wall. The source x of this filter consisted of the posted tweets, and the target y of this function is spam images, unwanted tweets and phish links.

So here the morphism can be represented as

f.x → y

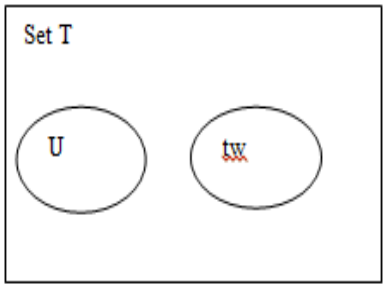
Where x- is a set of tweets and

y – Consisting of spam images, unwanted tweets and phish links

Here we can represent the sets

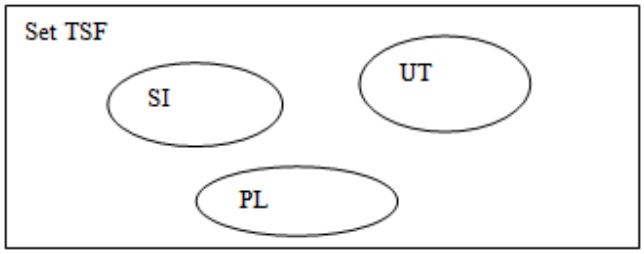
T- Set of user tweets and

TSF- set of filtering functions



Where U – set of users

tw - set of tweets by each user



Where SI- Spam Images

UT – Unwanted tweets

PL – phishing links

VII. MATHEMATICAL MODULE

A. Relevant Mathematics Associated With The Project

With faster increase in users on social networks, there is a corresponding increase in tweets which includes unwanted messages, spam Images, phish links. So here we proposed a system which filters these contents.

Input – {U,T, TSI, Pht }

Where- U Set of users

T – Set of tweets posted by each user

TSI – is a set of training spam images required for naïve Bayes classifier

Pht – Phish tank dataset to filter phish links

Output – { SFC, Alerts }

SFC - Spam Filtered Content after gone through three functions first Spam Image filter, second Phish link filter and third function unwanted message filter

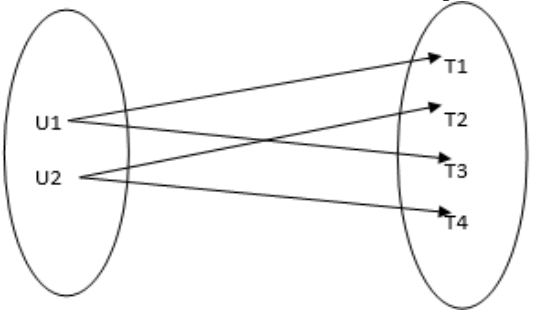
Consider a set U which consists of users registering with the application. This set can be represented as follows

U = {u1.u2,u3….Un }

Each user post number of tweets on his social wall, which may consist of text messages, links or URLs, Images. So we present here a set T as follows

T = {T1,T2,T3…..Tn}

The relation between Set U and set T can be represented as



Here ti that is a tweet posted by user can be represented as ti= {m, l, im}

Where m - textual messages

L – Url in message
im – Image uploaded by user
So here in our proposed work we are going to filter these tweet content, In order to filter textual content posted by different users we are using a user created filtering patterns fp. These filtering patterns can be used to filter the tweets posted on that user wall.

B. Functions

fm← Unwanted Tweet Filter (fp,m)
Where fm – filtered message
Next we are checking the images uploaded by users, whether it is a spam image or not. We are using here a naïve Bayes algorithm to detect the spam images.
fi ← Check Spam Ham (im)
Where fi – filtered image
Another processing that we are working on this tweet to check whether it contains phish links or not. Consider a tweet Ti contains Link li Here we are using Phish Link detection algorithm
Phi ← Check Phishing Link (li)
Where phi ← Filtered phishing links
After filtering these tweet contents user will alerted that someone posted unwanted contents on your wall.
SFC ← (fm, fi, phi)
Where SFC-Spam Filtered Content
Based on these three filtering functions we get the final output of our system, i.e. the Spam filtered Content, and the alerts related to spam images, phish links.

VIII. ARCHITECTURE DIAGRAM

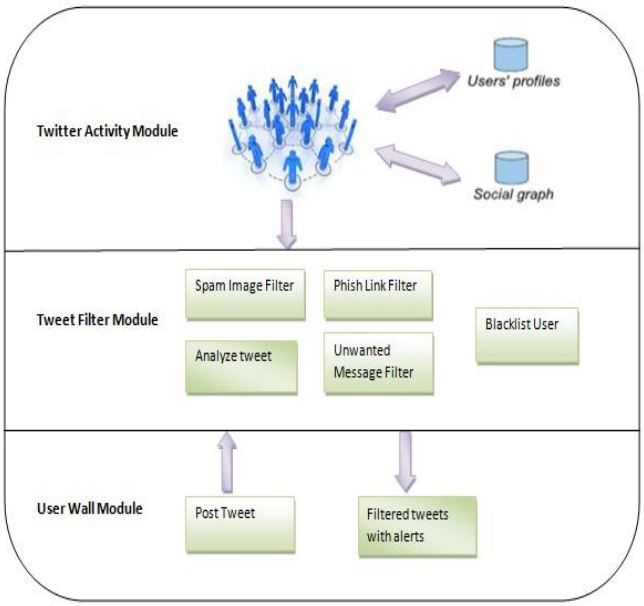


Fig. 3: Architecture Diagram

In this diagram we have represented our system in the form of modules. The first module is twitter activity module. It represents all the activities a user can perform while using the system. The second module is Tweet filter module, it shows what all the filtering procedures we are going to apply on incoming data such as Image filter, Phish link filter, Analysis of filter. The last module is User wall modules which notify users about suspicious content.

IX. TEST RESULTS

The testing was performed to degrade the software failures and to increase the fault tolerance capability of the system. It was taken into consideration that if there are any flaws in the software that were uncovered in the testing process, the logical errors were detected and corrected. The various testing types that were carried out for checking the accuracy of this system are: unit testing, integration testing, black box and white box testing, validation testing. The testing period for this system was carried out for 20 days after the system was implemented. In the initial phase, unit testing was carried out for modules such as filtering rules, phish link detection and spam images detections on twitter wall. The integration testing was done on all the modules by running the project on more than one system. Accordingly, the results were noted and the improvements were made. As the system got habituated to fault tolerance the number of systems were increased. The below table predicts the results of testing carried out:

No. of Systems used	Images	Messages	Phish Links	Fault Tolerance	Accuracy
1	2	10	5	Messages-10 Links-5 Images-2	99%
5	5	10	15	Messages-8 Links-13 Images-3	80%
10	15	150	95	Messages- 8 Links-13 Images-11	89.61%.
Total Accuracy for each	72.72 %	88.23%	93.91 %		

Table 1: Test Case for Accuracy of the System for Fault Tolerance

System Accuracy = 89.25%

The above table shows the results for the testing carried out on this system. In an heterogeneous environment the use of this system is considerable. It can tolerate more faults on a network. The table shows that when the tendency of the system is increased i.e number of system is increased by 10 then the overall accuracy of the system is also increased by 83.61 %, thus increasing its fault tolerance capability.

X. RESULTS

The login page that is displayed at the beginning when the user logins into the system.

The registration page is used to add new users to the system. Here, the user can provide their credentials

XI. CONCLUSION

Hackers bypass anti-spam filtering techniques by embedding malicious URL in the body of the messages. So the URL analyzer method with the help of minimized phishing feature set identifies the malicious URL in the Tweeter.

Differentiation of Spam content from Ham content is an important consideration for most people. Such a differentiation helps others to detect unwanted information and even threats to their cyber security. As a result many Internet researchers have a particular interest in finding the best classification algorithm to remove Spam.

Phishing has becoming a viral network security problem, causing financial loss of billions of dollars to both consumers and e-commerce companies. And perhaps more fundamentally, phishing has made e-commerce non trusting and unattractive to normal consumers. In this paper, we have studied the characteristics of the hyperlinks that were embedded in phishing contents. We then designed an anti-phishing algorithm, Link Guard, based on the derived characteristics. Since Phishing Guard is characteristic based, it not only detects known attacks, but also is effective to the unknown ones.

Our experiment showed that Link Guard is light-weighted and can detect up to 96% unknown phishing attacks in real-time. We believe that Link Guard is not only useful for detecting phishing attacks, but also can shield users from malicious or unsolicited links in Web pages and Instant messages. Our future work includes extending the Link Guard algorithm, so that it can handle CSS (cross site scripting) attacks.

REFERENCES

[1] Dhanalakshmi Ranganayakulu, Callappan C., "Detecting Malicious URL in E Mail implementation"- AASRI Conference of Intelligent system and control- 2013
[2] Mohammad-Ali Oveis-Gharan Faculty of engineering University college of Nabi Akram, Tabriz, Iran,- "Multiple Classifications for Detecting Spam email" by Novel Consultation Algorithm ,"- 2014 IEEE Salwa Adriana Saab, Nicholas Mitri, Mariette Hawed Faculty of Electrical and Computer Engineering.
[3] American University of Beirut, Lebanon," Ham or Spam? A comparative study for some Content-based Classification Algorithms for Email Filtering" -7th IEEE Mediterranean Electro technical Conference, Beirut, Lebanon, 13 April 2014
[4] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo,"Tweet Analysis for Real-Time Event Detection and Earthquake Reporting System Development¬" IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING VOL: 25 NO: 4 YEAR 2013.

