A methodology to detect malicious web sites using Classification algorithm

Dr.P.Suresh[1], K.N.Nithya[2], B.Manivannan[3]

Head, Dept of Computer Science, Salem Sowdeswari College - Govt. Aided, Salem[1]

Asst. Professor, Department of Computer Science, Sri Sakthi Kailaash Women's College, Salem [2]

Asst. Professor in Research Dept.of Computer Science and Applications Govt. Thirumagal Mills College, Gudiyattam, Vellore[3]

Abstract: **Malicious codes have been a major ad hock initiating from a local computer destruction which perpetuates even in malfunctioning of Internet by causing invincible attacks to the IP addresses. An attempt is made to pre-study the URL sites before the end users visit. In this paper, we implement an automated URL classification algorithm using Bayesian methods that determines malicious and benign websites from the list of URLs provided. The method is very effective at the server side as detection process is done beforehand and after letting the websites to be used by the end users.It is found to be very efficient in detecting malicious websites with only modest false positives.**

Keywords: **URL, Malicious Code Detection System, Naïve Bayes Algorithm.**

1. INTRODUCTION

Uniform Resource Locators (URLs), sometimes known as "Web links," are the primary means by which users locate resources on the Internet. Our goal is to detect malicious Web sites from the lexical and host-based features of their URLs. Criminal Web sites support a wide range of socially undesirable enterprises, including spam-advertised commerce (e.g., counterfeit watches or pharmaceuticals), financial fraud (e.g., via phishing or 419-type scams) and malware propagation (e.g., so-called "drive-by downloads"). Although the precise commercial motivations behind these schemes may differ, the common thread among them is the requirement that unsuspecting users visit their sites. These visits can be driven by email, Web search results, or links from other Web pages, but all require the user to take some action, suchas clicking, that specifies the desired Uniform (URL)[1].

Thus, each time a user decides whether to click on an unfamiliar URL that user must implicitly evaluate the associated risk. Is it safe to click on that URL, or will it expose the user to potential exploitation? Not surprisingly, this can be a difficult judgment for individual users to make.

Aware of this difficulty, security researchers have developed various systems to protect users from their uninformed choices. By far the most common technique, deployed in browser toolbars, Web filtering appliances, and search engines, is "blacklisting."

In this approach, a third-party service compiles the names of "known bad" Web sites (labeled by combinations of user feedback, Web crawling, and heuristic analysis of site content) and distributes the list to its subscribers. A user may click on a malicious URL before it appears on a blacklist (if it ever does).

URL: Just as we use file names to locate files on a local computer, we use Uniform Resource Locators (URLs) to locate Web sites and individual Web resources. One way users visit a site is by typing a URL into the browser's address bar

An arguably easier way is to click a link which is contained within a page that is already rendered by the browser, or an email message rendered by an email client.URLs has the following standard syntax.

<protocol>://<hostname><path>

The <protocol> portion of the URL indicates which network protocol should be used to fetch the requested resource.

The <hostname> is the identifier for the Web server on the Internet.

The <path> of a URL is analogous to the path name o-f a file on a local computer

We now provide further background on the mechanisms for constructing, resolving, and hosting a Web site's URL. We relate these mechanisms to the goal of detecting malicious URLs, hinting at their applications as useful indicators for automated detection and deferring detailed discussion about the features of URL.

A. DOMAIN NAME SYSTEM RESOLUTION:

The Domain Name System (DNS) is a hierarchical network of servers responsible for translating domain names into IP addresses and other kinds of information [1].

During the DNS resolution process, the to kens in the domain name are traversed from right-to-left to direct the client's DNS resolver to query the appropriate DNS name servers.

B. DOMAIN NAME REGISTRATION:

Besides the IP addresses associated with a domain name, there is useful information associated with domain name registration. Registration establishes which name servers are associated with a domain name[1].

Typically, the registrant registers the primary domain name (a term we define shortly) with the registrar; the registrant is the owner of the domain name, and the registrar is the organization responsible for hosting the NS record that points to the primary domain's servers

II.MALICIOUS CODE DETECTION SYSTEM

Malware the malicious software is used to gather sensitive information, disrupt computer operation or to have access to secure computer systems. It can be appear in the form of coding, scripts, active contents and other software. Malware is known as computer pollution, as in the legal rules of several United States[9].

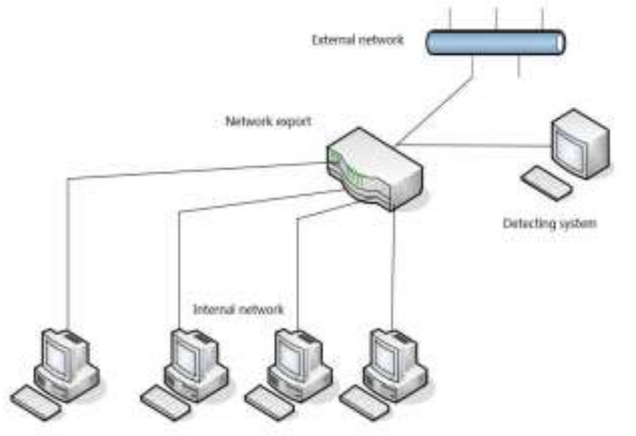


Fig 1. Operation Environment of Unknown Malicious Code Detection System

Malware is different from unusable software, which is legitimate software but having harmful bugs that were not removed before release. Malware is the term used to refer a variety of forms of intrusive software [8]. Malware mainly includes different computer viruses, ransom ware, Trojan horses, worms, root kits, key loggers, adware, dialers, spyware, rogue security software and some other malicious programs; the majority of active malware threats are normally Trojans or warm rather than viruses

III.CLASSIFICATION ALGORITHM

The introduction of data mining technology, applied to the malicious code detection system, completes the process of automatically extracted from a large number of data. In the process of establishing the attack detection system, it can categorize the malicious and benign websites, which can develop a set of automatic tools to generate the attack detection model from various audit data[2].

By using correlation analysis and sequential pattern analysis, we find the relation between the features and the time sequence, so as to complete the collection process of the user's network behavior information data.

Naive Bayes algorithm:

Naïve Bayes Classifier technique is mostly preferred when the dimensionality of the inputs is high. In spite of simplicity of Naive Bayes, it can handle and perform better than more complicated classification methods. It is a classification technique based on Bayes theorem with an assumption of independence among predictors [3][7]. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

Bayes theorem provides a way of calculating posterior probability P(h|D) from P(D), P(h) and P(D|h). Look at the equation below:

P(h/D)= P(D/h) P(h) / P(D)

P(h) : Prior probability of hypothesis h
P(D) : Prior probability of training data D
P(h/D) : Probability of h given D
P(D/h) : Probability of D given h

IV. PROPOSED METHOD

Bayesian reasoning is applied to decision making and inferential statistics that deals with probability inference. The knowledge of prior events is used to predict future events[7]. This prediction probability can be applied here to detect the presence of malicious codes. Assuming that the benign and malicious websites are of equal probability, we categorize as below;

Posterior probability:
A probability that the fed website is malicious or being website.
Prior probability:
The fed website is taken for the process.
Class:
The web server maintains a collection of classes, under which, a set of acknowledged website lists are stored.

Steps:

1. Once the website is given by the client it is pre operated by the web server.
2. The web server finds the proximity of the URL by applying the above formulae.
3. The result is compared with the threshold value by setting the equal probability between benign and malicious websites.
4. The highest posterior probability is the outcome of the result.

Thus, Naive Bayes classifier calculates the likelihood that a program is having malevolent code given the features that are present in the program [8].

This approach used both string and byte sequences data for computing a probability of a binary's malicious code having some features.

V. RESULT AND DISCUSSION

Step1: We generate a data set table containing information related to benign websites (W1,….,Wn) and equivalent websites(E1,….,En)to be sent to the clients when requested and suspicious websites(M1,....,Mn) that are to be blocked as follows

Web Sites	Acknowledgement
W1	Yes
E1	Yes
W2	Yes
M1	No
W3	Yes
M2	No
E2	Yes
W4	Yes
E3	Yes
W5	Yes
E4	No
W6	Yes
M3	No
W7	Yes

Step2:

Now convert the data set in to a frequency table.

Web Sites	Yes	No
W	7	-
M	-	3
E	4	

The total number of transaction is found to be 14.

Step 3: Create a Likelihood table by finding the probabilities like Overcast probability = 0.21 and probability of playing is 0.78.

Website	Yes	No		
W	7	-	=6/14	=0.42
M	-	3	=3/14	=0.21
E	4	-	=4/14	=0.28
All	11	3		
Total Probability	=11/14=0.78	=3/14=0.21		

Likelihood Table

Step 4: Now we use Naive Bayesian equation to calculate the posterior probability for each class. The class with the highest posterior probability is the outcome of prediction.Therefore the procedure to permit acknowledgement for a website access is;

P (Yes | W) = P (W | Yes) * P(Yes) / P (W)

Here we have P (W |Yes) = 7/11 = 0.63, P(W) = 7/14 = 0.5, P(Yes)= 11/14 = 0.78

Now, P (Yes | Sunny) = 0.63 * 0.78/ 0.5 = 0.98, which has a higher probability.

VI.CONCLUSION

The above discussion gives a clear exposure on the efficiency of Naïve Bayes classification algorithm towards detection of malicious web sites. The most vital fact is that a precautional maintenance of the data sets denoting the class similarity has to be done. By using this methodology at the web server side, the network administrator who plays a major role in delivering the services to the client, can detect with the highest probability. We hope that by using the above algorithm, we can also generate the list of malicious sites accurately.

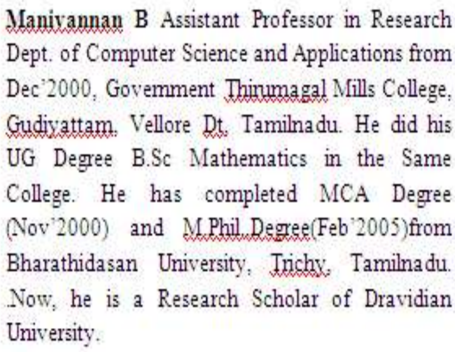
VII. REFERENCES

1. "Security Applications for Malicious Code Detection Using Data Mining ", International Journal of Computer Science Trends and Technology (IJCST) – Volume 3 Issue 1, Jan-Feb 2015
2. "Learning to Detect Malicious URLs", ACM Transactions on Intelligent Systems and Technology, Vol. 2, No. 3, Article 30, Publication date: April 2011.
3. "Detection of Malicious Data using hybrid of Classification and Clustering Algorithms under Data Mining", International Journal of Computer Applications (0975 – 8887) Volume 104 – No 11, October 2014
4. "Malicious URL Detection and Identification", International Journal of Computer Applications (0975 – 8887) Volume 99 – No.17, August 2014
5. "A Study of Encryption Algorithms (RSA, DES, 3DES and AES) for Information Security",International Journal of Computer Applications (0975 – 8887) Volume 67– No.19, April 2013
6. "Research on Intrusion Detection Systems and Unknown Malcode Detection based on Network Behavior", International Journal of Security and Its Applications Vol. 10, No. 5 (2016) pp.315-326
7. "Data Mining for Malicious Code Detection System ", Multidisciplinary Journal of Research in Engineering and Technology, Volume 2, Issue 3,July 2015.
8. "Learning to Detect and Classify Malicious Executables in the Wild", Journal of Machine Learning Research 7 (2006) 2721-2744, June 2012.
9. "Malicious Code Detection through Data Mining Techniques ", International Journal of Computer Science & Engineering Technology,ISSN : 2229-3345 Vol. 5 No. 05 May 2014.

BIOGRAPHY

Dr. P. Suresh is the Head, Department of Computer Science, Salem Sowdeswari College [Govt. Aided], Salem. He received the M.Sc., Degree from Bharathidasan University in 1995, M.Phil Degree from Manonmaniam Sundaranar University in 2003, M.S (By Research) Degree from Anna University, Chennai in 2008, PGDHE Diploma in Higher Education and Ph.D., Degree from Vinayaka Missions University in 2010 and 2011 respectively in Computer Science. He is an Editorial Advisory Board Member of Elixir Journal. His research interest includes Data Mining and Natural Language Processing. He is a member of Computer Science Teachers Association, New York.

Mrs.K.N.Nithya is an Assistant Professor in the Department of Computer Science, Sri SakthiKailaash Women's College. She received the B.C.A degree and M.C.A degree from Periyar University. Her research area of interest includes Data Mining and Mobile Computing.



Manivannan B Assistant Professor in Research Dept. of Computer Science and Applications from Dec'2000, Government Thirumagal Mills College, Gudiyattam, Vellore Dt, Tamilnadu. He did his UG Degree B.Sc Mathematics in the Same College. He has completed MCA Degree (Nov'2000) and M.Phil Degree(Feb'2005)from Bharathidasan University, Trichy, Tamilnadu. .Now, he is a Research Scholar of Dravidian University.