International Journal of Engineering & Technology, 7 (4.6) (2018) 476-479

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Malicious Website Collection System Using Machine Learning

Jayakumar Shriya[1]*, S Rajendran[2]

[1]M.Tech(ISCF) student, Department of Information Technology, SRM Institute of Science and Technology, Kattankulathur
[2] Professor, Department of Information Technology, SRM Institute of Science and Technology, Kattankulathur
**Corresponding author*

Abstract

Malicious websites are those sites which have malicious content or files in it. It lures the user when they click on it either by going to some other irrelevant site or downloading some malicious content in the user system without the user's knowledge. These websites appear to be legitimate websites but they are malicious sites. It contains various content such as spam, phishing, driven-by-download, virus, ransomware and other etc. These malicious sites even cause huge losses to a particular organization or to an individual user. Typically a blacklisting mechanism is used to detect malicious websites. But these blacklisting mechanism doesn't work efficiently to find all kinds of malicious sites. This blacklisting mechanism can be easily evaded by the attacker. To overcome this blacklisting mechanism a machine learning approach is used to detect and tackle all kind of malicious contents in the web pages. This machine learning approach can't be evaded by the attacker. Supervised and Unsupervised machine learning approaches are used to detect malicious websites. [1] The supervised approach is used to detect known attacks were Unsupervised learning is used to detect unknown malicious websites. Unsupervised learning is done using a machine learning approach. For classification of websites, we use Hidden Markov Model(HMM) which is safe and reliable for operating on the internet. This model works efficiently to find inter-dependencies among the resources. A fast feature extraction is used to find the attributes, the Baum Welch algorithm and Viterbi algorithm in the Markov model used to detect malicious URLs more accurately and precisely. This shows that the application of HMM enhances the performance to classify the data sets and gives more accurate results. This model is applied on all social media.

Keywords: *Internet, Malicious websites, blacklist, Machine learning, Hidden Markov Model(HMM),legitimate websites.*

1. Introduction

Nowadays internet has become mandatory of all kind of needs in our daily life. It has been used in various fields like education, business, banking, and financial sectors. Based on the needs of the user's many websites has been increased which is most important stage where miscreants attack user's..[2] Download attacks are the most frequently happening attack. In this, the attacker uses various techniques on the web pages to make it as malicious sites or to link a benign site with the malicious site. Once the victim clicks on these sites they are taken to malicious sites without their notice. It makes the attacker's to gain victim information that is stored on the host systems, which leads to grave financial loss. Most of the malign websites has driven by download exploits. An attacker may also use tricks on the web page to make it look legitimate. Consider an example, when you visit a webpage that asks the user to install a fake video player that necessary to show a video(in fact, it is malware binary). Another example includes virus program injected in the website that takes the victim to several sites. Since rapid growth of websites, it is a challenge for us to prove the malicious and benign sites.

There are many changes and stages in the history of malicious software. Which has been exposed and detected in hosts and networks. These are self-replicating adware but not self-transporting. Malware attacks are sharply increasing with increase in complexity and interconnection of rising information systems. The user are being target by clicking on some unknown URLs. To prevent the users from visiting such URLs research generated by security industry is going on.

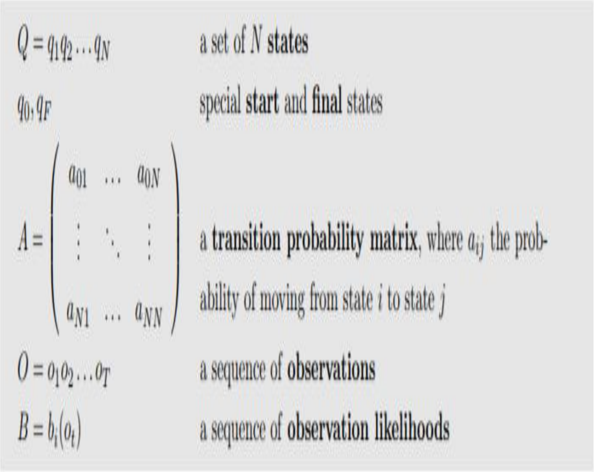
Signatures with various short and exclusive strings in the program are used to organize scrupulous threats in executable files and records of a boot. The disadvantage, these signature technique is not effective to customize and find malign content executions. The heuristic method is more complex than signature techniques and it also consumes more time but it fails to detect new malicious executables.

Traditionally websites were detected through domain names using, reverse technologies and data mining techniques, as new network technologies are applied, malicious domain names creation and usage has become more flexible, so the method cannot effectively detect these malicious domains. Number of registration and system deployments of the global domain name system are increasing along with that the complexity of DDOS attack scale and technology used for the attacking the domain name system are also increased. This Hidden Markov model finds all hidden contents in the particular URL. We can also use Naive Bayes algorithm for finding hidden contents but it doesn't work well for identifying for all types of malicious content in the URL and couldn't able to classify large datasets. These malicious URLs may be shared in any social media but the victim doesn't know that it is malicious or legitimate URL. The victim just clicks on the URL present in the social media. As soon as the URL is clicked it takes the victim to some other webpage. These webpages may be malicious webpages. Some video links may be malicious links such as when the video has been played by the victim behind the video there may be some malicious content which gathers the victim's information or inject the malicious files in the victim's system. Hidden Markov model works well to identify hidden contents in videos, images and other etc. These videos appear to be legitimate so it is not any blacklisting method. This machine learning approach can't be evaded by an attacker. So it detects well to identify all types of malicious content in the URL which contains video or images or any other content.

2. Related Work

Strider Honey Monkey which is Microsoft research project, this project tells use how to detect website exploiting on internet explorer. In this project Yi-Ming et. al described man unknown and zero day attacks.They installed Internet explorer on various patched Windows machines and analyzed the transitions of each machine. Rohit Shukla and Maninder Singh explained the concept of using Python for detecting malicious Urls in their paper "PythonHoneyMonkey: Detecting Malicious Web URLs on Client Side Honeypot Systems"[3] they presented similar project related to Microsoft research project they used snort tool to blacklist all malicious Urls since snort has predefined signatures and A python based utility Beautiful Soup is used on windows OS as web Crawler whereas Lynx is used for Linux based OS which is web Crawler tool. IP blacklist file is Created which stores the IP address by using Snort IDS tool. The snort tool runs in background which maintains logs on activity going in the system using internet. [4] The blacklist based methods are collection of malicious URLs which blacklists the URL when the page is being visited it happens by querying the blacklist database.The links for blacklist are collected from various websites such as Phishtank, Safe browsing websites, Site Advisor websites and Websense ThreatSeeker Network. The blacklists contains various malicious websites those websites are collected from user feedbacks and by crawling the websites by crawlers. In some cases honeypots are used to find out malicious websites which are also included in the blacklists. This blacklisting method is correct and simple to find out malicious websites but the drawback of this method are they produce slow results because of direct verification process and results are not accurate as they don't assure that every new malicious URL will be in the blacklist database. The text based method check the matching text of the web page of a URL to detect whether it is malicious or not.This method is very useful and of much consequence. For example, Provos et al.discovered malicious URLs using features from the content of the equivalent URLs,as the presence of definite javascript and iFrames are inappropriate.Moshchuk et al. used the anti-spyware tools to check downloaded trojan executables to find malicious URLs. Byung-ik Kim et al. malicious websites are detected based on the entropy of each characteristic of the website, entropy, frequency, density, content of thejavascript. The advantage of this content based methods is which an offline detection and analysis; they don't work as good as for online detection. The challenges in online detection is that they often incur major latency,because examining and check page text repeatedly costs much computation time and resource. AbubakrSirageldin, Baharum B. Baharudin, and Low Tang Jung their paper "Malicious Web Page Detection: A Machine Learning Approach"[5] explains about a framework to detect malicious webpages using artificial neural network techniques.This framework reduced high false positive rate. This framework is partial method used for URL feature collection. Frank Vanhoenshoven∗, Gonzalo N´apoles∗, Rafael Falcon, Koen Vanhoof∗ and Mario K¨oppen they used a machine learning approach in their paper "Detecting Malicious URLs using Machine Learning Techniques" they overcome the problems in the blacklisting method and used various algorithms in machine learning such as random forest and multilayer perception. These algorithm requires more calculations and it is also time-consuming process. The random forest method can be for URL classification it can't be used for other methods. This method only predicts numerical features which are used for training set. By using Naive Bayes method there is a problem that larger number malicious URLs are undetected. At the time of classifying the URLs many URLs are incorrectly classified. The prediction rate is very low in Naive Bayes algorithm. Toshiki Shibahara et al. paper on detecting malicious website by integrating various webpages using redirection subgraph, it uses various subgraphs to detect malicious, benign and compromised URLs. A redirection subgraph is used to detect redirection site whether the redirection sites are malicious or benign. But this method can analyze only some website. It can also be easily evaded by an attacker. The processing the datasets is also very slow even though it improvised true positive rate by malicious subgraph. Honeyclients are not suitable to classify all types of malicious URLs. It used subgraph instead of Content Management system (CMS) which is vulnerable and easily evaded by the attacker. L.Vu et al. proposed FirstFilter in 2016 which is the cost sensitive approach which outperform the binary classifier in the multi-layer systems but it can't be used at client side it can be used for large organization or enterprises.

In this paper we use Hidden Markov Model(HMM) we can also use Naive Bayes algorithm for classification but it is static process it is not used for all types of classification it requires more computational time. By using HMM we can easily classify the malicious sites and we also find interdependence among various clusters. Hidden Markov model is used for statistically determining the behaviour of the pattern. They are used in recognizing the speech of a person or animal, detecting the malware in the files and in biological analysis fields. Markov model has various states with known probabilities of transitions in states these states are clearly visible.[6] Whereas a Hidden Markov Model (HMM) has invisible states and it is a machine learning approach. HMM behaves like state transition machine where every state has its own probability distribution. The probabilities are fixed for variation in states.[7]



$Q = q_1 q_2 \dots q_N$	a set of N **states**
q_0, q_F	special **start** and **final** states
$A = \begin{pmatrix} a_{01} & \dots & a_{0N} \\ \vdots & \ddots & \vdots \\ a_{N1} & \dots & a_{NN} \end{pmatrix}$	a **transition probability matrix**, where a_{ij} the probability of moving from state i to state j
$O = o_1 o_2 \dots o_T$	a sequence of **observations**
$B = b_i(o_t)$	a sequence of **observation likelihoods**

Fig1: The five parameters used by Hidden Markov Model

The three basic issues in Hidden Markov model are:

1. How to predict output sequence probability of a model?
2. How to predict the output state when model and output sequence is provided?
3. How to evaluate the parameter?

Hidden Markov model is the mathematical model for classification of the malicious URLs in the training phase. The output sequence probability is predicted by baum welch algorithm. To analyze state sequence we are going to use Viterbi algorithm.[8]

The algorithms used in HMM to solve the issues in it are [9]

- Filtering the state of the observed sequence done by using Forward algorithm.
- Hidden states are predicted by using Forward and Backward algorithm.
- State transition are predicted by Viterbi algorithm.
- Baum Welch algorithm as known as EM algorithm used evaluating HMM parameters.

3. System Architecture

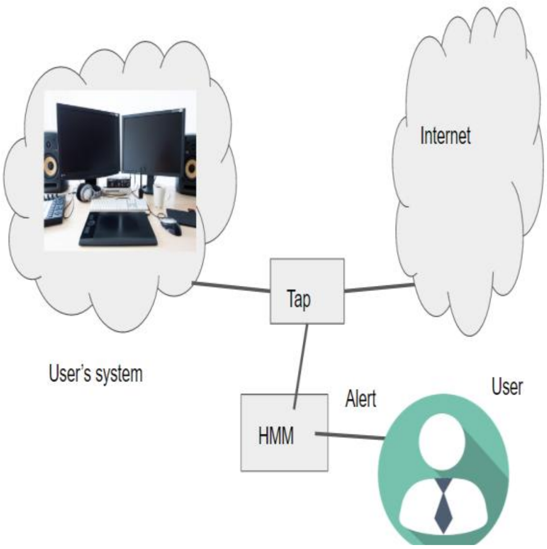


Fig 2: The Framework of proposed system

When the client start browsing the internet , tap is the software which continuously monitors the users activity on online. Tap monitors continuously at background in the user system. Tap sends all the URLs which the client is accessing on the internet to HMM(Hidden Markov Model) this mathematical model helps to classify malicious Urls from normal Urls. After classifying malicious Urls and benign Urls the HMM shows alert if it is malicious Urls to the users. This HMM works more efficiently for analyzing malicious contents in the websites.

3.1 Working of Hidden Markov Model:

Malign domain names are abstract and concrete attributes are extracted from DNS logs may have variation states at different times. Some domain names behave well with defined signatures. Some behave abnormally due to having some malicious signatures which could damage at various levels. At continuous intervals, the transition between various malicious states is linked with the Markov character. These states are hidden, the domain names which are related to extracted domains from DNS logs are observed and their characteristics are also measured. The observation variables show the malicious strength of the websites and this malicious strength is measured by the state of the domain. The probability is calculated for each input based on the variations in the output we predict whether the site malicious or not. If the probability of the particular input is very low it detects it as malicious.

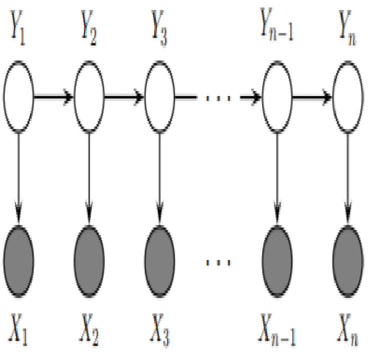


Fig3: graphical representation of a hidden Markov model

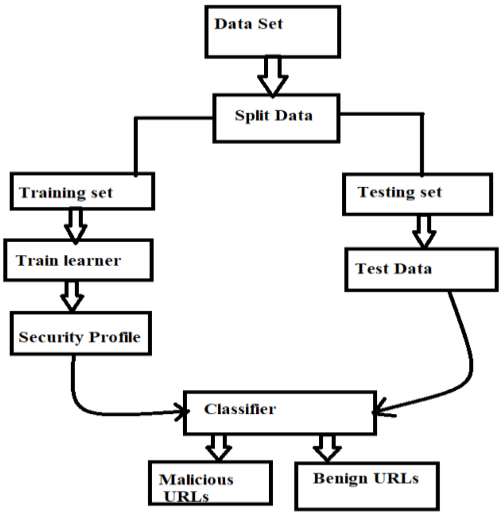


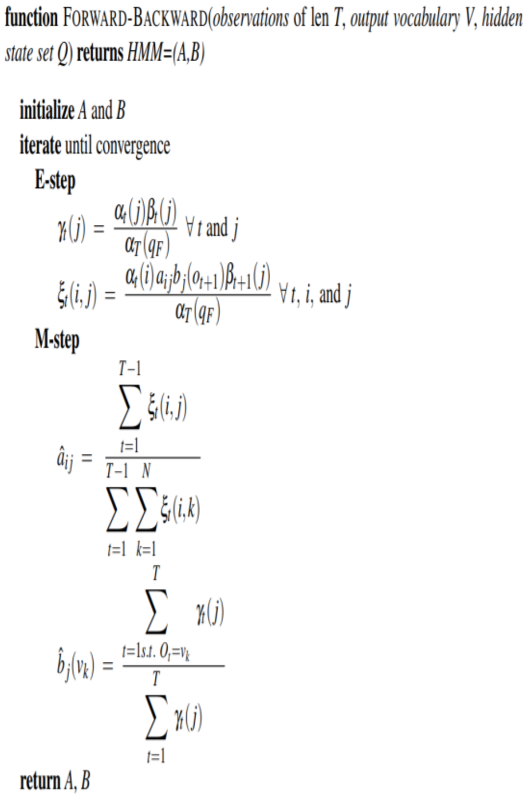
Fig4: Machine Learning Classifier

The Data set which contains set of URLs. By using HMM we are splitting the data into two sets one as training and other as testing. The training dataset contains 80% and testing dataset contains 20%. In order to evaluate the most efficient mechanism to detect malicious URLs.

It uses EM algorithm, which iteratively calculates and finally obtain a model with evaluated parameters. Baum-Welch algorithm are not global solutions for HMM algorithm they local solutions. Therefore, different local solutions are obtained for given training data using Baum-Welch algorithm.

For training the machine learning algorithm Baum-Welch algorithm is used. For testing the data Viterbi algorithm is used. A security profile is made on training data. Compared training data and testing data Pearson correlation algorithm is used.[10]

3.2 Baum-Welch algorithm:



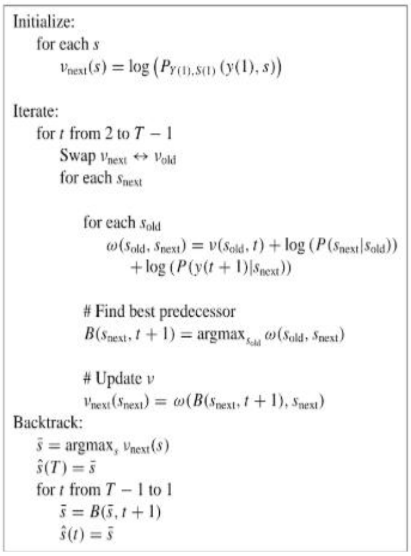
```
function FORWARD-BACKWARD(observations of len $T$, output vocabulary $V$, hidden state set $Q$) returns $HMM=(A,B)$

 initialize $A$ and $B$
 iterate until convergence
   E-step
     $\gamma_t(j) = \frac{\alpha_t(j)\beta_t(j)}{\alpha_T(q_F)}\ \forall\, t$ and $j$
     $\xi_t(i,j) = \frac{\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)}{\alpha_T(q_F)}\ \forall\, t, i,$ and $j$
   M-step
     $\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1}\xi_t(i,j)}{\sum_{t=1}^{T-1}\sum_{k=1}^{N}\xi_t(i,k)}$
     $\hat{b}_j(v_k) = \frac{\sum_{t=1 s.t.\, O_t=v_k}^{T}\gamma_t(j)}{\sum_{t=1}^{T}\gamma_t(j)}$
 return $A, B$
```

Fig 5: pseudocode of Baum Welch algorithm

This algorithm estimate HMM parameter for each URL. Initial probabilities are made in two sets A and B. Modification and estimations are done on these two sets.

3.3 Viterbi algorithm:

This algorithm finds the best path for each input in the testing phase.[11] It checks whether linked URLs are taking in the proper path. It also predicts what are the hidden sequence and observed sequence in the testing set. [12]



```
Initialize:
    for each $s$
        $v_{\text{next}}(s) = \log\left(P_{Y(1),S(1)}(y(1), s)\right)$

Iterate:
    for $t$ from 2 to $T-1$
        Swap $v_{\text{next}} \leftrightarrow v_{\text{old}}$
        for each $s_{\text{next}}$

            for each $s_{\text{old}}$
                $\omega(s_{\text{old}}, s_{\text{next}}) = v(s_{\text{old}}, t) + \log\left(P(s_{\text{next}}|s_{\text{old}})\right)$
                    $+ \log\left(P(y(t+1)|s_{\text{next}})\right)$

            # Find best predecessor
            $B(s_{\text{next}}, t+1) = \text{argmax}_{s_{\text{old}}}\, \omega(s_{\text{old}}, s_{\text{next}})$

            # Update $v$
            $v_{\text{next}}(s_{\text{next}}) = \omega(B(s_{\text{next}}, t+1), s_{\text{next}})$
Backtrack:
    $\bar{s} = \text{argmax}_s\, v_{\text{next}}(s)$
    $\hat{s}(T) = \bar{s}$
    for $t$ from $T-1$ to 1
        $\bar{s} = B(\bar{s}, t+1)$
        $\hat{s}(t) = \bar{s}$
```

Fig 6: pseudocode of Viterbi algorithm

3.4 Pearson's Correlation Coefficient:

It helps you find out the association between two quantities. It gives you the measure of the strength of association between two variables. The value of Pearson's Correlation Coefficient can be between -1 to +1. 1 means that they are benign Urls and -1 means that they are Malicious Urls.[13]

4. Conclusion

Malware codes are designed to gain access to the victim's or to lead an organization to huge financial losses. These are the major security threat for a user or to an organization. These malware codes can be injected in the normal websites by the attackers to gain access of the victim's system or to gather victim information such as passwords, PIN, fingerprints, iris, pictures, and video etc. The attacker breaches into the victim system through the internet using malign sites without victim acknowledge. Web crawlers, detection, blacklisting, the lightweight algorithm doesn't work well to identify all types of malign sites. This problem has been solved by using a machine learning approach such as HMM. The issues in HMM are solved so HMM works effectively to identify all types hidden Malign URLs in benign URLs. Large computations are done in less time but it is expensive to process. It works well to build an application in offline if we build online detection process we need to use Spark for fast feature extraction, Weka tool to detect malicious sites.

References

[1] Malicious URL Detection using Machine Learning: A Survey Doyen Sahoo, Chenghao Liu, and Steven C.H. Hoi

[2] https://books.google.co.in/books?id=YydaDwAAQBAJ&pg=PA19&lpg=PA19&dq=malicious+internet+web+site+collection+system&source=bl&ots=IUrKvIk9nj&sig=j8B26i4HxnZZdX3JenqFeqEc-u4&hl=en&sa=X&ved=2ahUKEwjB94Le0sTcAhWVTX0KHRc_CaMQ6AEwAHoECAgQAQ

[3] Honeypot Frameworks and Their Applications: A New Framework ,By Chee Keong NG, Lei Pan, Yang Xiang

[4] https://arxiv.org/pdf/1701.07179.pdf

[5] https://link.springer.com/search?query=malicious+web+page+detection+using+machine+learning+

[6] https://www.youtube.com/watch?v=1R00TP8iNrU

[7] https://www.youtube.com/watch?v=n3iANHusfcY

[8] Annachhatre, C., et al (2015) Hidden Markov Models for Malware Classification. Journal in Computer Virology and Hacking Techniques, 11, 59-73. http://dx.doi.org/10.1007/s11416-014-0215-x

[9] Bazrafshan, Z., Hashemi, H., Fard, S.M.H. and Hamzeh, A. (2013) A Survey on Heuristic Malware Detection Techniques. The 5th Conference on Information and Knowledge Technology (IKT 2013), Shiraz, 28-30 May 2013, 113-120. http://dx.doi.org/10.1109/ikt.2013.6620049

[10] Ichise H, Jin Y, Iida K. Detection Method of DNS-based Botnet Communication Using Obtained NS Record History[C]// Computer Software and Applications Conference. IEEE, 2015:676-677.

[11] http://cecas.clemson.edu/~ahoover/ece854/refs/Gonze-ViterbiAlgorithm.pdf

[12] https://www.quora.com/What-is-an-intuitive-explanation-of-the-Viterbi-algorithm

[13] http://www.shokhirev.com/nikolai/abc/alg/hmm/hmm.html