Asian Journal of Information Technology 15 (19): 3846-3851, 2016
ISSN: 1682-3915

Stream Life Estimation Based on the End Point Elimination Technique to Restrict Client Side Script for Secure Computing

[1]P. Malathi and [2]P. Vivekanandan
[1]Dhanalakshmi Srinivasan College of Engineering and Technology, Chennai, India
[2]Department of Chemical Engineering, A.C.Tech, Anna University, Chennai, India

Abstract: The growing internet access rate makes the malicious users to spread malware, spyware, viruses into the genuine users storage medium and steal many user specific personal information. Also, the malware spoils the working of user devices and reduces the performance and usage of the user computers. The user information steals by the malware are used to perform various malformed activities by them. Sometimes, the malware which intrudes into the user devices transfer many information about the customer information which can be used to perform various network threats. To solve this issues, there are many antivirus programs and tools are designed which monitor the activities and presence of viruses in the user devices. Apart from this, there are few scripts which are running at the time of user visit which transfers much user personal information without knowing the users. This kind of programs or scripts will not be monitored by any virus programs or security tools. By considering these client side scripts, a stream estimation based end point elimination technique is proposed and it identifies a set of endpoints or connections established at any point of time and eliminates un_trusted connections to secure the internet access. The proposed approach has produced efficient results in client script restriction and has reduced the time complexity also.

Key words: End point elimination, client scripts, internet security, stream life estimation, malware

INTRODUCTION

The usage of internet has growing in rapid manner and the people run behind internet for everything to be done in their day to day life. Whatever the job has to be done, they believe on internet and they spend their most of the time in surfing internet. While surfing they access many banking web sites or commercial web sites where they submit many user specific personal security information. For example, to purchase a cosmetic through online, they submit the credit card number and the security password details in the web form to complete the shopping.

Now a days, there are many advertisements being generated while viewing the web page and each has different purpose. Generally, the web user does not care about the advertisements being displayed and they simply ignore the advertisements. If there is any malware presents in the machine then the malware can monitor the web pages being visited and trace the keyboard reading or the web page content. The captured user data could be transferred to another malware controller located somewhere in the world. The captured information can be used to perform variety of threats towards not only financial stability but also against the information stability of the country also.

There are cases where the user generates intensive endpoint connections to download some files from the web server. Once the connection has been made, the user may close the parent process but still the child may be alive or the malicious thread may be alive even after the parent thread stop which may be running and sending information to the remote servers. In order to provide data security, the connections have to be monitored and also the end points have to be closed properly to provide higher internet security.

The lifetime of the connection has to be estimated to perform security enforcement in internet communications. The lifetime of the stream connections can be estimated according to the priority and the frequency of the connection has to be used to estimate the lifetime of the stream connection. By estimating the lifetime of the stream, the end points can be controlled to provide higher order internet security by restricting the client side scripts. Also, not all the client scripts use the same Secure Socket Layer connection to the server where the web page is available and they also use different connections which work alive even after the user has closed the parent process. In this cases, the previous approaches miss the unidentified client connections which run under the authorized parent process.

Literature review: There are many approaches have been discussed in literature, a few of them related to problem identification is discussed. Identity-Based secure distributed data storage schemes propose two

Corresponding Author: P. Malathi, Dhanalakshmi Srinivasan College of Engineering and Technology, Chennai, India

Identity-Based Secure Distributed Data Storage (IBSDDS) schemes. The proposed schemes can capture the following properties: The file owner can decide the access permission independently without the help of the Private Key Generator (PKG). For one query, a receiver can only access one file, instead of all files of the owner Our schemes are secure against the collusion attacks, namely even if the receiver can compromise the proxy servers, he cannot obtain the owner's secret key. Although, the first scheme is only secure against the Chosen Plaintext Attacks (CPA), the second scheme is secure against the Chosen Cipher text Attacks (CCA). To the best of our knowledge, it is the first IBSDDS schemes where an access permissions are made by the owner for an exact file and collusion attacks can be protected in the standard model.

Modeling the pairwise key predistribution scheme in the Presence of Unreliable Links (Yagan and Makowski, 2013), present conditions on how to scale the model parameters so that the network has no secure node that is isolated and securely connected, both with high probability, when the number of sensor nodes become large. The results are given in the form of zero-one laws and exhibit significant differences with corresponding results in the full-visibility case.

NICE: Network Intrusion Detection and Countermeasure Selection in virtual network systems (Chun *et al*., 2013), repose a multi-phase distributed vulnerability detection, measurement and countermeasure selection mechanism called NICE, which is built on attack graph based analytical models and reconfigurable virtual network-based countermeasures. The proposed framework leverages Open Flow network programming APIs to build a monitor and control plane over distributed programmable virtual switches in order to significantly improve attack detection and mitigate attack consequences. The system and security evaluations demonstrate the efficiency and effectiveness of the proposed solution.

Privacy Preserving Data Sharing With Anonymous ID Assignment (Dunning and Kresman, 2013), designed an algorithm for anonymous sharing of private data among parties. This technique is used iteratively to assign these nodes ID numbers ranging from 1-N. This assignment is anonymous in that the identities received are unknown to the other members of the group. Resistance to collusion among other members is verified in an information theoretic sense when private communication channels are used. This assignment of serial numbers allows more complex data to be shared and has applications to other problems in privacy preserving data mining, collision avoidance in communications and distributed database access. The required computations are distributed without using a trusted central authority.

SORT: A Self-Organizing Trust Model for Peer-to-Peer Systems (Singh and Kayalvizhi, 2013) presents distributed algorithms that enable a peer to reason about trustworthiness of other peers based on past interactions and recommendations. Peers create their own trust network in their proximity by using local information available and do not try to learn global trust information. Two contexts of trust, service and recommendation contexts are defined to measure trustworthiness in providing services and giving recommendations. Interactions and recommendations are evaluated based on importance, recentness and peer satisfaction parameters. Additionally, recommender's trustworthiness and confidence about a recommendation are considered while evaluating recommendations.

WARNINGBIRD: A Near Real-time Detection System for Suspicious URLs in Twitter Stream (Lee and Kim, 2013), propose WARNINGBIRD, a suspicious URL detection system for Twitter. The proposed system investigates correlations of URL redirect chains extracted from several tweets. Because attackers have limited resources and usually reuse them, their URL redirect chains frequently share the same URLs. Some methods are proposed to discover correlated URL redirect chains using the frequently shared URLs and to determine their suspiciousness. Numerous tweets are collected from the Twitter public timeline and built a statistical classifier using them.

Two tales of privacy in online social networks (Gurses and Diaz, 2013) first provide an introduction to the surveillance and social privacy perspectives emphasizing the narratives that inform them, as well as their assumptions, goals and methods. We then juxtapose the differences between these two approaches in order to understand their complementarily and to identify potential integration challenges as well as research questions that so far have been left unanswered.

Twitsper: Tweeting privately (Singh *et al*., 2013) introduce Twitsper, to support fine-grained control over who sees a user's messages. Twitsper provides privacy controls to the users of Twitter today without relying on Twitter to make changes. This is because it is a wrapper around Twitter that enables private group communication while preserving Twitter's commercial interests. It preserves privacy both from the Twitsper server as well as from undesired Twitsper users.

Protecting user against phishing using Antiphishing (Kirda and Kruegel 2005), presents a novel browser extension, AntiPhish, that aims to prevent users against spoofed web site-based phishing attack. To this end,

AntiPhish tracks the sensitive information of a user and generates warnings whenever the user attempts to give away this information to a web site that is considered unfaithed. The most effective solution to phishing is training users not to blindly follow links to web sites where they have to enter sensitive information such as passwords. However, expecting that all users will understand the phishing threat and surf accordingly is unrealistic. The users are is tricked to visit the phishing web site. Therefore, it is important for researchers and industry to provide solutions for the phishing threat. Most proposed phishing solutions are based on the crawling of websites to identify "clones" and the maintenance of black lists of phishing websites. Such solutions, however, require the antiphishing organizations to be much faster than the attackers.

Classification of Phishing Email Using Random Forest Machine Learning Technique (Akinyelu and Adewumi, 2014), investigates and reports the use of random forest machine learning algorithm in classification of phishing attacks with the major objective of developing an improved phishing email classifier with better prediction accuracy and fewer numbers of features. From a dataset consisting of 2000 phishing and ham emails, a set of prominent phishing email features (identified from literature) were extracted and used by the machine learning algorithm with a resulting classification accuracy of 99.7% and low False Negative (FN) and False Positive (FP) rates.

Automatically determining phishing campaigns using the USCAP methodology (Layton *et al*., 2010), presented that looks at the differences that occur between phishing websites from an authorship analysis perspective and is able to determine different phishing campaigns undertaken by phishing groups. The methodology is named USCAP, for Unsupervised SCAP which builds on the SCAP methodology from supervised authorship and extends it for unsupervised learning problems. The phishing website source code is examined to generate a model that gives the size and scope of each of the recognized phishing campaigns. The USCAP methodology introduces the first time that phishing websites have been clustered by campaign in an automatic and reliable way, compared to previous methods which relied on costly expert analysis of phishing websites. Evaluation of these clusters indicates that each cluster is strongly consistent with a high stability and reliability when analyzed using new information about the attacks, such as the dates that the attack occurred on. The clusters found are indicative of different phishing campaigns, presenting a step towards an automated phishing authorship analysis methodology.

Textual and visual content-based anti-phishing: a Bayesian approach (Zhang *et al*., 2011) presented a novel framework using a Bayesian approach for content-based phishing web page detection. The proposed model takes into account textual and visual contents to measure the similarity between the protected web page and suspicious web pages. A text classifier, an image classifier and an algorithm fusing the results from classifiers are introduced. An outstanding feature of this paper is the exploration of a Bayesian model to estimate the matching threshold. This is required in the classifier for determining the class of the web page and identifying whether the web page is phishing or not. In the text classifier, the naive Bayes rule is used to calculate the probability that a web page is phishing. In the image classifier, the earth mover's distance is employed to measure the visual similarity and the proposed Bayesian model is designed to determine the threshold. In the data fusion algorithm, the Bayes theory is used to synthesize the classification results from textual and visual content. The effectiveness of the proposed approach was examined in a large-scale dataset collected from real phishing cases. Experimental results demonstrated that the text classifier and the image classifier we designed deliver promising results, the fusion algorithm outperforms either of the individual classifiers and the proposed model can be adapted to different phishing cases.

A multi-tier phishing detection and filtering approach (Islam and Abawajy, 2013), propose a new approach called multi-tier classification model for phishing email filtering. An innovative method is proposed for extracting the features of phishing email based on weighting of message content and message header and select the features according to priority ranking. Also, the impact of rescheduling the classifier algorithms in a multi-tier classification process is examined to find out the optimum scheduling. A detailed empirical performance and analysis of the proposed algorithm is present. The results of the experiments show that the proposed algorithm reduces the false positive problems substantially with lower complexity.

All the above discussed methods reviewed only explicit behaviors of social network or external behavior of web pages or users on the web. The problem is approached in different way so that the back end process could be monitored to provide internet security to the users.

MATERIALS AND METHODS

Proposed approach: The proposed client script restriction approach which performs the script monitoring based on various parameters like the parent process and their lifetime estimation, amount of data to be allowed. Based

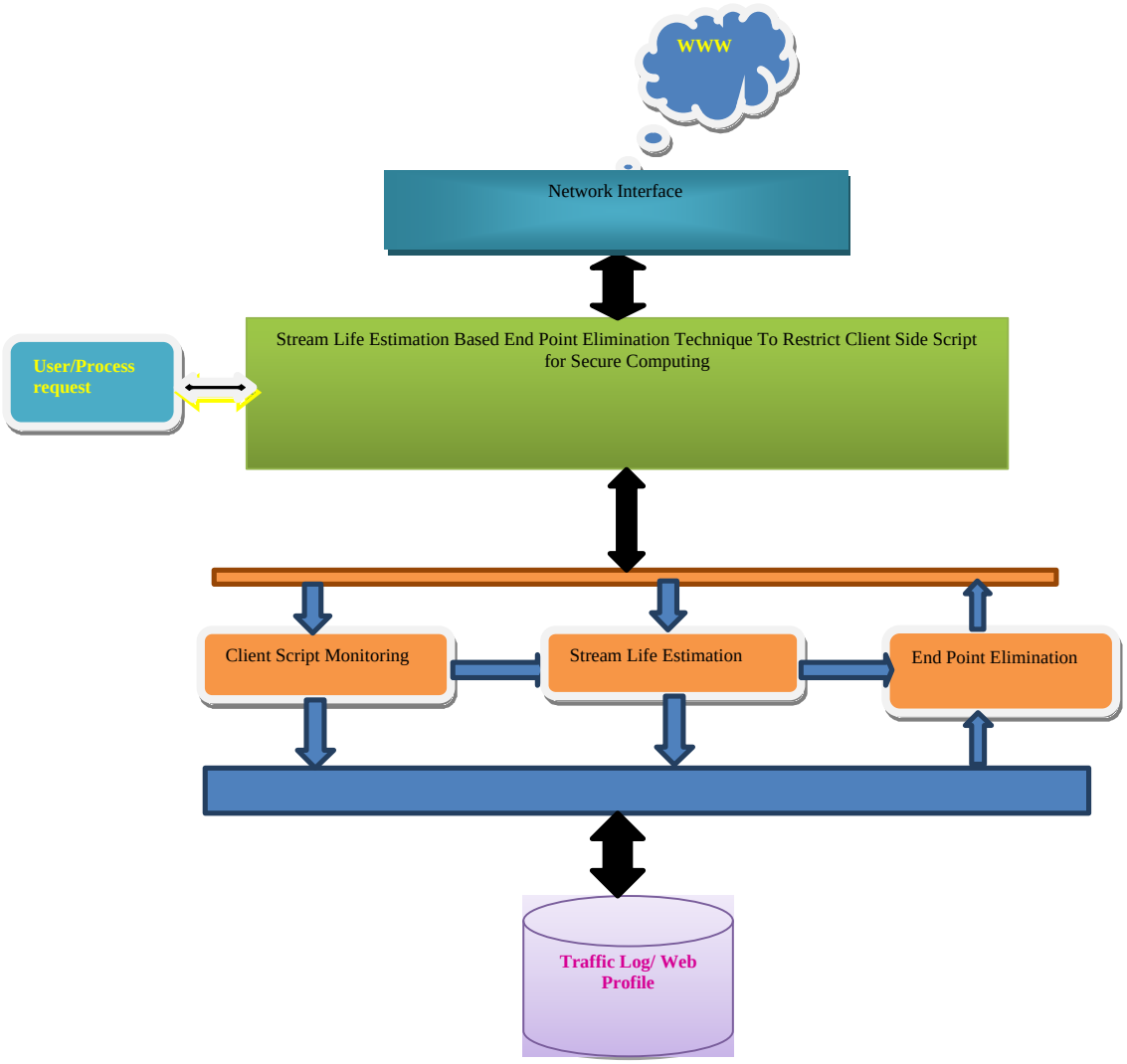

Fig. 1: Proposed system architecture

on all these parameters, the endpoints are monitored and closed on malformed transformation of data from the source computers. The proposed method has different functional models, namely, Stream Life Estimation, Endpoint Elimination, Client script Monitoring. Each of them is discussed in detail. Figure 1, shows the architecture of the proposed system and its functional components of the system.

Client script monitoring: The proposed method collects set of all scripts and parent processes running in the system. For each parent process, there will be a set of url and domain address available. The addresses and the mac addresses are retrieved from the available connections and the list of client scripts enabled are retrieved. For each client script running, the parent processes are identified and the time they start and the amount of data being sent. Based on all these the network access rate is computed and it specifies the amount of data transfer involved. The lifetime of the client script is computed by performing stream approximation. Based on all these measures, a set of client scripts are being selected for elimination at all the time.

Algorithm:

Input: Process set Ps
Output: Client Script suspected SCS.
Step1: Initialize script set SS.
Step2: for each parent P_i from Ps
Identify all scripts

$$S=\int_{i=1}^{Size(Ps)}\sum CS\in(Pi)$$

End

Step3: for each script Si from SS
Compute running time RT = Si(C)-Si(S).
Identify the parent status PS.
If Ps==Running then
Compute network access rate NAR.
NAR = $\int_{i=1}^{N} \sum \text{Size(Data)} \in \text{Pi}$

If NAR>ATH then //access threshold
Add to suspected script SCS = SCS = ∑sj(SCS)+Si
End
Else
Add to suspected script SCS = SCS = ∑sj(SCS)+Si
End.
End
Step4: stop.

Stream life estimation: The stream line has unique or limited lifetime which is a script and the scripts can be run upto certain time from the start time. For each client script running, we estimate the life time by identifying the parent process running time and the parent process end time and the client script total running time and the network access rate. Based on all these measures, we compute each client scripts trustworthy by the measure Client Script Lifetime Factor value to conclude the trustworthy of the client script.

Algorithm:
Input: Suspected client script SC, Access History Ah.
Output: client script lifetime factor
Step1: compute total running time of script RT = $\int$ Sc(St)-Sc(Et)
Step2: compute parent process end time Pet.
Step3: compute client script start time St.
Step4: compute deviation Dev = Pet-st/Rt
Step5: Compute client script network access rate NAR =

$$\int_{i=1}^{size(Ah)} \sum \text{Size(Data)} \in \text{Ah(i)}$$

Step6: Trustworthy Tw = NAR*Dev
Step7: If Tw> TH then //Th-trust threshold
Close Client script

End
Step8: stop.

Endpoint elimination: The client scripts are communicating with the remote servers through a particular end point which will be generated at the start of the script or at the start of the machine. Those endpoints are identified using the client script monitoring process and stream life estimation technique. Based on the result of these two approaches the client scripts are closed and the end point associated with the client script is closed to secure the internet communication.

Algorithm:
Step1: Initialize client script monitoring.
Step2: get Suspected client scripts SCS.
Step3: Compute Stream life estimation.
Step4: Identify malicious scripts.
Step5: Close end points and generate access histories.
Step6: stop.

RESULTS AND DISCUSSION

The proposed end point elimination technique has been implemented and tested for its efficiency. The method has identified the malicious client side scripts and restricts them in efficient manner. The proposed method monitors the client scripts for its running time and data transfer performed and their parent process status and the time running after the parent being closed. Based on all these details we compute various measures like to compute the trustworthy of the client process or client script to identify the trustworthy.

Figure 2 shows the time complexity achieved by different methods for 1000 URL's and their log base. It shows that the proposed End Point Elimination method has produced less time complexity values.

Figure 3 shows that the space complexity achieved by different approaches and the less memory taken to process the same set of logs by the proposed approach. Also, it shows that it has produced less space complexity than other methods.

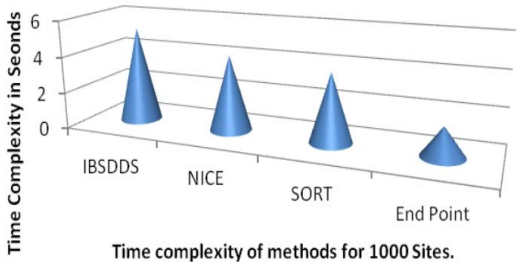

Fig. 2: Shows the time complexity of different methods for 1000 sites

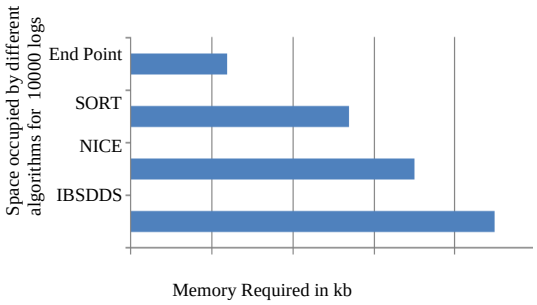

Fig. 3: Space complexity of different approaches

CONCLUSION

A new extended security scheme called End point elimination technique is proposed based on stream life estimation, which monitors outgoing traffic and extracts various features of the packet. The extracted features are used to compute the lifetime of the stream and the client script which specifies the trustworthy of the client script. The network access rate of each suspected client script is computed to identify the trustworthy of the client script. The analysis is performed even for the trusted web sites and scripts by computing the memory access traces performed by the scripts of authorized web sites. The proposed method has produced higher efficient security and better results. Also, the proposed method has produced less time and space complexity values.

REFERENCES

Akinyelu, A.A. and A.O. Adewumi, 2014. Classification of phishing email using random forest machine learning technique. J. Appl. Math., Vol. 2014, Can, A.B. and B. Bhargava, 2013. Sort: A self-organizing trust model for peer-to-peer systems. IEEE. Trans. Dependable Secure Comput., 10: 14-27.

Chung, C.J., P. Khatkar, T. Xing, J. Lee and D. Huang, 2013. NICE: Network intrusion detection and countermeasure selection in virtual network systems. IEEE. Trans. Dependable Secure Comput., 10: 198-211.

Dunning, L.A. and R. Kresman, 2013. Privacy preserving data sharing with anonymous ID assignment. Inf. Forensics Secur. IEEE. Trans., 8: 402-413.

Gurses, S. and C. Diaz, 2013. Two tales of privacy in online social networks. IEEE. Secur. Privacy, 11: 29-37.

Islam, R. and J. Abawajy, 2013. A multi-tier phishing detection and filtering approach. J. Network Comput. Appl., 36: 324-335.

Kirda, E. and C. Kruegel, 2005. Protecting users against phishing attacks with antiphish. Proceedings of the 29th Annual International Conference on Computer Software and Applications (COMPSAC'05), July 26-28, 2005, IEEE, Vienna, Austria, ISBN: 0-7695-2413-3, pp: 517-524.

Layton, R., P. Watters and R. Dazeley, 2010. Automatically determining phishing campaigns using the uscap methodology. Proceedings of the Conference on ECrime Researchers Summit (eCrime), October 18-20, 2010, IEEE, Ballarat, Australia, ISBN: 978-1-4244-7760-9, pp: 1-8.

Lee, S. and J. Kim, 2013. Warningbird: A near real-time detection system for suspicious urls in twitter stream. Dependable Secure Comput. IEEE. Trans., 10: 183-195.

Singh, I., M. Butkiewicz, H.V. Madhyastha, S.V. Krishnamurthy and S. Addepalli, 2013. Twitsper: Tweeting privately. IEEE. Secur. Privacy, 11: 46-50.

Singh, N.H. and A. Kayalvizhi, 2013. Combining cryptographic primitives to prevent jamming attacks in wireless networks. Proceedings of the 2013 International Conference on Information Communication and Embedded Systems (ICICES), February 21-22, 2013, IEEE, India, ISBN: 978-1-4673-5786-9, pp: 251-255.

Yagan, O. and A.M. Makowski, 2013. Modeling the pairwise key predistribution scheme in the presence of unreliable links. IEEE. Trans. Inf. Theor., 59: 1740-1760.

Zhang, H., G. Liu, T.W.S. Chow and W. Liu, 2011. Textual and visual content-based anti-phishing: A bayesian approach. IEEE Trans. Neural Networks, 22: 1532-1546.