An Efficient Ransomware Detection System

Anjalee Menen, R. Gowtham

Abstract--- Cyber security protects the system from unauthorized access and destruction of data. The intention is to provide security to the system by blocking attackers. Malware or malicious software is any kind of program which is developed with the aim of doing harm to victim's data. Viruses, worms, Trojan horses, Ransomware, and spyware are different types of malware. When malicious software enters into the system, it will encrypt the user data, deletes or modifies the data. This type of software also used to steal the user data. Ransomware is one of the types of malware which was developed with the intention of getting money from the victims. When Ransomware starts executing in our system, it will start encrypting, deleting and modifying files. The user will get decryption key only after paying the claimed money. Many have found some solutions for detecting some specific Ransomware. The existing technique includes Static based technique which uses signature analysis which can only detect known Ransomware since it compares the extracted code snippet of the target executable with the database of known malware samples. The existing technique is based on the known input and known output and can only detect known Ransomware samples. In this paper we have proposed an efficient Ransomware detection system based on the analysis of behavior with the help of machine learning technique. In the proposed technique, we analyzed the possible behavior of Ransomware based on the changes to user's files, addition of registry key, stopping the active processes. Based on this behavior, the decision is made using Machine learning technique.

Keywords--- Ransomware, Cybersecurity, Malware, Machine Learning

I. INTRODUCTION

Ransomware is one of the dangerous malware that affects the user's data by encrypting, modifying or deleting it or block access to the files [1]. The main intention of the attackers is to get money from the victim. Some Ransomware locks the desktop screen and some other Ransomware uses the cryptographic technique to encrypt the victim's files, and make them inaccessible and demands a payment to decrypt the encrypted file. When Ransomware starts activating in the system, after making changes to the files, a ransom note which gives the directions to recover the files will appear on the screen. Ransomware will enter into systems when users visit malicious websites or download attachment from the mail.

When screen locking Ransomware enters into the system, an image or notification is displayed on the victim's system's screen, which blocks the victims from accessing their system. This will show the directions to victims on how users can pay for the ransom. Ransomware targets the files such as DOC, .XLS, .JPG, .ZIP, .PDF, SQL files etc. which are commonly used by the user [2].

A lot of researchers are working to find an efficient method for detecting Ransomware. Many have found some solutions for detecting some specific ransomware. Ransomware has created data losses and critical IT industry disruption. More than hundred countries have been affected globally. Such behavior of ransomware has resulted in various researchers in many organizations and universities [14][17][18].

The proposed system will first do behavior analysis. Behavior analysis module consists of three submodules. They are tracking file changes, the addition of registry key, stopping the active processes. After analyzing those behaviors, Pattern extraction and feature vector generation module is used to extract features then in vulnerability scanning module, SVM is used as classification scheme. After that decision module takes the decision based on the vulnerability scanning module. Finally, if Ransomware is detected, the user gets an alert message.

II. RELATED WORK

Deepak et al. [3] proposed Signature-based technique which was used to detect malware. This technique can identify only the known malware. A database consists of signature of known malware will be created and the code string patterns of the target executable are extracted and compare with the database. If the extracted code pattern matches the code pattern in the database, then the executable is malicious. This method is applicable only for detecting known Ransomware.

Chris Moore [4] proposed Ransomware detection method based on honey pot computers. Honeypot computers are fake computers which act as decoy computers. This is used to discover malicious access. Since traditional antivirus software cannot detect new forms of malware, it is necessary to detect the new forms of Ransomware when ransomware begins to execute. It was assumed that honeypot computers will be affected first. When any malicious behavior is detected, an email is sent to the network administrator and on further detection; the corresponding system will be disconnected from the system. This method couldn't guarantee that the ransomware will affect the honeypot computers first.

Krzysztof Cabaj et.al. [5] Proposed Software-defined networking for the reduction of ransomware. Two SDN applications were developed. The SDN1 application will forward all the DNS traffic which are sending to the victim's computer is sent to the controller. DNS message will be inspected and compared with the values in the database. The list of known proxy servers are there in this database. If the list in the database matches with the domain name, then that process will be discarded.

Manuscript received February 01, 2019

Anjalee Menen, Dept of Computer Science and Engineering, Amrita School of Engineering, Coimbatore, Tamil Nadu, Amrita Vishwa Vidyapeetham, India. (e-mail: cb.en.p2cse16003@cb.students.amrita.edu)

Dr.R. Gowtham, Dept of Computer Science and Engineering, Amrita School of Engineering, Coimbatore, Tamil Nadu, Amrita Vishwa Vidyapeetham, India. (e-mail:r_gowtham@cb.amrita.edu

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

SDN2 that was developed after the SDN1. SDN2 will improve the performance of the SDN1. The drawback of SDN1 is that since for each of the messages is checked with the lists in the database, the DNS traffic will be delayed. Because of this, SDN2 forwards the DNS message to the recipient and forward the copy of the DNS message to its controller. This technique is only applicable for detecting known ransomware.

Butler et al. [6] developed CryptoDrop, which is a ransomware detection system that alerts a user in prior to the suspicious file activity. In this they used a set of indicators based on the behavior such as similarity measurement, entropy measurement; CryptoDrop will alert the user and halts the process. By using the set of indicators, the system was set for detecting ransomware. CryptoDrop halted ransomware from executing. This method couldn't find whether user or the suspicious process is making changes to the file system.

Kharraz et.al [7] proposed that Protecting Master File Table and Inspecting the file system will also help in detecting Ransomware. For monitoring file system activity they captured all the I/O requests. For capturing I/Or requests, they used minifilter driver, This technique is not efficient because it cannot be done at user level.

Nikolai Hampton et al. [16] analyzed behavior of ransomware based on the Windows platform call tracer approach. This method tracked the Windows API calls of processes in the system.

III. PROPOSED WORK

The proposal has six main modules. They are Behavior analysis module which includes Tracking file changes, Addition of registry key, stopping the active processes. Based on behavior analysis pattern is extracted and feature vector is generated in the pattern extraction and feature vector generation module. After that Support Vector Machine [15] is used for classification purpose which is done in vulnerability scanning module. After that decision is made in the decision making module based on the outcome of the vulnerability scanning module. If decision making module predicts that the application running in the system is Ransomware, then an alert is sent to the user. Figure 1 shows the Ransomware detection system for the proposal.

3.1 Behaviour Analysis Module

Behavior analysis module analyses the behavior shown by the ransomware and also detects the presence of Ransomware based on the abnormalities arise in the system behavior. For efficient behaviour analysis we need an application which notifies the changes happening in the system. For this we planned to deploy following dynamic monitoring techniques.

Tracking File Changes

Whenever Ransomware starts executing in our system, it will start making changes to our file system. It will rename, delete, create and encrypt the files. So it is necessary to monitor the file systems for changes. Our program will monitor deletion, changes in the content of files, the creation of files. This will helps to track suspicious process happening in the system. When ransomware starts working in our system, it will encrypt files, replace the content of data with some other contents, rename the files and folders, and create new files.

Addition of Registry Key

To complete the ransomware execution process, it should continue executing during the system reboots. To ensure the completion of attack, ransomware will create a registry entry in one of the startup directory in the registry so that it can continue its process after reboot.

The malware creates the registry keys in:
Windows\CurrentVersion\Run\"value" in HKCU registry
Windows\CurrentVersion\Run\"value" in HKLM registry

So it is necessary to monitor registry key entry in the HKCU\..\CurrentVersion\Run\registry.

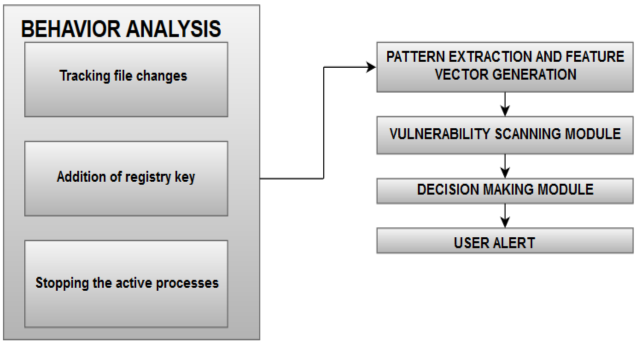

BEHAVIOR ANALYSIS
Tracking file changes
Addition of registry key
Stopping the active processes
PATTERN EXTRACTION AND FEATURE VECTOR GENERATION
VULNERABILITY SCANNING MODULE
DECISION MAKING MODULE
USER ALERT

Fig. 1: Ransomware Detection System

Stopping the active processes

For making changes to the files, Ransomware stops active processes. This submodule monitors the active processes and notifies when the process stops.

3.2 Pattern extraction and feature vector generation

In this module, we use Supervised-machine learning. This method is the training of a decision algorithm to recognize certain behavioral traits of running processes that optimally discriminates between ransomware and benign applications. The training of the decision algorithm requires the training dataset containing examples of known (i.e. labeled) ransomware and known benign applications, the capturing of behavioral traits in a quantifiable manner, and the classification scheme which defines the training and prediction algorithm. In this stage, pattern is extracted and the feature vector is generated. We have vectors like file change per second, deletions per second, renaming per second, creation per second, number of active processes terminated per second.

3.3 Vulnerability scanning module

Support Vector Machine is used as a classification method. In the training data and real time data we are monitoring the vectors such as file change per second, deletions per second, renaming per second, creation per second, number of active processes terminated per second and addition of registry key value. And during execution we find the file change per second, deletions per second, renaming per second, creation per second, number of active processes terminated per second rates and compare it with the training set using nonlinear classification. Depending on the proximity of the new data points, data is separated into two different planes.

3.4 Decision making module

Decision making module makes decision based on the outcome of the vulnerability scanning module.

By analyzing the points, our program identifies whether the application running in our system is Ransomware or not. If the points are falling under the training data set, then the application is considered as the Ransomware otherwise it is considered as legitimate application.

3.5 User alert

If the decision making module predicts that the application is Ransomware, an alert message us immediately send to the user. After getting alert message, user will be able to take necessary actions. Disconnecting other computers connected to the affected computers will save computers from the data loss.

IV. IMPLEMENTATION DETAILS

Ransomware Behavior Analysis was carried out in the Oracle Virtual Box. We Prepared data sets for the experimental purposes. Downloaded 700 active Ransomware samples and 1000 legitimate samples from various sources on the internet (8) (9) (10) (11) (12) (13). The description of the Ransomware samples and legitimate samples used for the experimental analysis are shown in Table 1 and Table 2.After that, we executed ransomware and analyzed its behavior. Based on the behavior observed, we developed a program to monitor the analyzed behavior using C#. For developing SVM, we installed SVM package with NuGet. After that, we prepared data and trained the data set and used to make predictions.

Table 1: Legitimate and Ransomware sample source

SOURCE	NUMBER OF SAMPLES	LINKS
File Forum	512	https://fileforum.betanews.com/
Major Geeks	205	http://www.majorgeeks.com/
Softpedia	283	http://www.softpedia.com
Ytisf/theZoo	96	http://bit.ly/2KI3CqQ
TPSC Forums	444	https://forum.thepcsecuritychannel.com/c/malware/malware-samples
RISS-Ransomware Dataset	160	http://rissgroup.org/ransomware-dataset/

V. RESULTS AND ANALYSIS

In our experiments, sensitivity, miss rate and accuracy are considered for evaluating the performance of our proposed system.

The sensitivity or True Positive Rate(TPR) calculates the rate of ransomware samples which are correctly detected as ransomware with respect to sum of correctly classified ransomware samples and incorrectly classified ransomware samples.

$$TPR = \frac{M_{r\to r}}{(M_{r\to r}+M_{r\to l})}$$

The False Negative Rate (FNR) indicates the rate of ransomware samples incorrectly classified as not ransomware.

$$FNR = 1\text{-}TPR$$

The fall-out or False Positive Rate (FPR) indicates the ratio between the numberoflegitimate samples that are judged as Ransomware and the sum of the legitimate samples correctly classified as legitimate samples and the number of legitimate samples classified as Ransomware samples.

$$FPR = \frac{M_{l\to r}}{M_{l\to l}+M_{l\to r}}$$

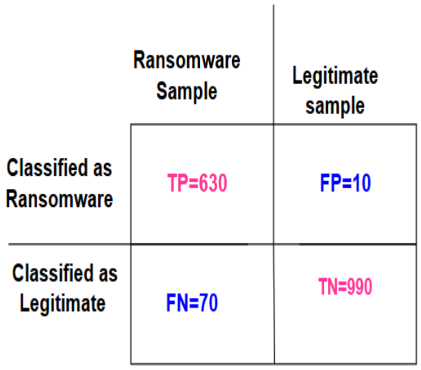
The True Negative Rate (TNR) measures the rate of legitimate samples classified as non ransomware.

$$TNR = 1\text{-}FPR$$

Accuracy indicates how often the detection of ransomware is correct.

$$ACC = \frac{M_{l\to l}+M_{r\to r}}{M_{l\to l}+M_{l\to r}+M_{r\to r}+M_{r\to l}}$$

The notations $M_{l\to l}$, $M_{r\to r}$ represent correctly classified samples and the notations $M_{l\to r}$, $M_{r\to l}$ represent wrongly classified samples by our proposed system.



	Ransomware Sample	Legitimate sample
Classified as Ransomware	TP=630	FP=10
Classified as Legitimate	FN=70	TN=990

Fig 2: Experimental results

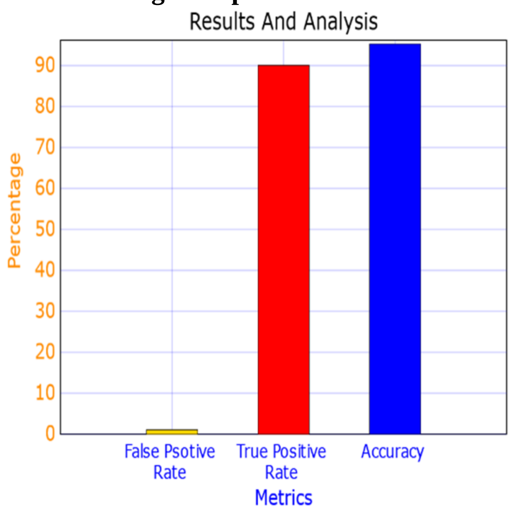


Fig 3: Experimental inference

The experiment results and experimental inference of our proposed system are shown in Fig 2 and Fig 3. From the experiments we could find that the fall out (FPR) of our proposed system is one percent and sensitivity (TPR) of the proposal is 90 percent, and the accuracy of the proposal is 95.2 percent as shown in Fig. 3. Thus our result analysis shows that our proposal could detectransomware more accurately with fewer false alarms.

VI. CONCLUSION

As a part of the proposed system, we have developed application that detects ransomware based on the behavior shown by the ransomware. This proposed system tracked file changes, monitored addition of registry keys, monitored system process as part of behavior analysis then used machine learning technique for decision making. The experimental analysis shows that the proposed system could effectively detect the ransomware based on the behavior with the help of machine learning technique.

REFERENCES

1. Ali, Azad. "Ransomware: a research and a personal case study of dealing with this nasty malware." *Issues in Informing Science and Information Technology* 14 (2017): 087-099.
2. Ransomware Damage Report, Cybersecurity Ventures, 2017. . [Visited on April 2018]
3. cybersecurityventures.com-ransomware-damage-report-2017-5-billion
4. Venugopal, Deepak, and Guoning Hu. "Efficient signature based malware detection on mobile devices." *Mobile Information Systems* 4.1 (2008): 33-49.
5. Moore, Chris. "Detecting ransomware with honeypot techniques." *Cybersecurity and Cyberforensics Conference (CCC), 2016*. IEEE, 2016.
6. Cabaj, Krzysztof, and WojciechMazurczyk. "Using software-defined networking for ransomware mitigation: the case of cryptowall." *IEEE Network* 30.6 (2016): 14-20.
7. Scaife, N., Carter, H., Traynor, P., & Butler, K. R. (2016, June). Cryptolock (and drop it)- stopping Ransomware-attacks on user-data. In *Distributed Computing Systems- (ICDCS), 2016- IEEE -36th International Conference on* (pp. 303-312)- IEEE.
8. Kharraz, Amin, et al. "Cutting the gordian knot: A look under the hood of ransomware attacks." *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer, Cham, 2015.
9. TPSC Forums. [Visited on August 2017] https://forum.thepcsecuritychannel.com/c/malware/malware-samples
10. ytisf/theZoo. [Visited on August 2017] http://bit.ly/2KI3CqQ
11. RISS-Ransomware Dataset. [Visited on October 2017] http://rissgroup.org/ransomware-dataset/
12. File forum. [Visited on August 2017] https://fileforum.betanews.com/
13. Major Geeks. [Visited on January 2018] http://www.majorgeeks.com/
14. Softpedia. [Visited on August 2017] http://www.softpedia.com/
15. Ramesh, Gowtham, Jithendranath Gupta, and P. G. Gamya. "Identification-of- phishing- webpages- and -its –target- domains -by -analyzing the –feign-relationship." *Journal of Information- Security and Applications* 35- (2017)-75-84.
16. Support vector machines for machine learning, [Visited on January 2018], machine learning mastery.com-support-vector-machines-for-machine -learning
17. Hampton, Nikolai, ZubairBaig, and SheraliZeadally. "Ransomware behavioural analysis on windows platforms." *Journal of Information Security and Applications* 40 (2018): 44-51.
18. Aravind, V., and M. Sethumadhavan. "A-framework-for-analysing-the-security of-chrome-extensions." Advanced Computing, Networking-and-Informatics-Vol-2. Springer, Cham, 2014. 267-272.
19. Dr. M. Sethumadhavan and H. VaNath, Gangadharan, Kb, and, “Reconciliation engine and metric for network vulnerability assessment”, in ACM- International Conference- Proceeding-Series, Kerala, 2012, pp. 9-21.