Pune Institute of Computer Technology
Dhankawadi, Pune

A SEMINAR REPORT
ON

MALWARE DETECTION AND CLASSIFICATION USING MACHINE LEARNING AND CUCKOO SANDBOX

SUBMITTED BY

Shubhankar Gaikwad
Roll No. 31265
Class TE-2

Under the guidance of
Prof. S. N. Girme

DEPARTMENT OF COMPUTER ENGINEERING
Academic Year 2019-20

DEPARTMENT OF COMPUTER ENGINEERING

Pune Institute of Computer Technology
Dhankawadi, Pune-43

CERTIFICATE

This is to certify that the Seminar report entitled

“MALWARE DETECTION AND CLASSIFICATION USING MACHINE LEARNING AND CUCKOO SANDBOX”

Submitted by
Shubhankar Gaikwad Roll No. 31265

has satisfactorily completed a seminar report under the guidance of Prof. S. N. Girme towards the partial fulfillment of third year Computer Engineering Semester II, Academic Year 2019-20 of Savitribai Phule Pune University.

Prof. S. N. Girme
Internal Guide

Prof. M.S.Takalikar
Head
Department of Computer Engineering

Place:
Date:

ACKNOWLEDGEMENT

I sincerely thank our Seminar Coordinator Prof. B.D.Zope and Head of Department Prof. M.S.Takalikar for their support.

I also sincerely convey my gratitude to my guide Prof. S. N. Girme, Department of Computer Engineering for her constant support, providing all the help, motivation and encouragement from beginning till end to make this seminar a grand success.

I would also like to thank Mr. Kamalkanta Sethi and Mr. Rahul Kumar(IIT-BBS) who introduced me to the topic during the ACM-India Winter School of Cybersecurity 2019 at NISER, Bhubaneshwar.

Contents

List of Tables

List of Figures

Abstract

In this era of technological disruptions and big data where algorithms have began taking decisions, and data has become the new currency and power, there is a need for stricter surveillance regimes to protect all individuals from cyber attacks and threats.

Monitoring the immense amount of data to resist malicious actors is becoming tougher day by day. As intruders are becoming smarter with time, we need intelligent systems to tackle these modern day threats to information systems. The aim is to study various network attacks and discuss different types of intrusion detection systems (IDS). Various machine learning models can be used to detect malicious attacks based on the signature of the malware. It is necessary to provide an intelligent system to select suitable algorithm for malware detection in different scenarios.

Cuckoo Sandbox is a tool used for dynamic analysis of malicious files. Machine learning algorithms are used to extract features from report for further classifications to improve the efficiency of an IDS.

Keywords

Malware Detection, Malware Classification, Cyber Security, Machine Learning, Cuckoo Sandbox, Cyber Attacks

1 INTRODUCTION

Cybersecurity has been an integral part of the information systems since the beginning of information revolution. The battle between code makers and code breakers is now at a critical stage. Today the amount of data in the world is manifold. Given the importance and power of data in our lives, it has now become need of the hour to protect our systems from malicious actors and softwares.

Security breaches due to malwares can lead to service denials, damage to resources and many business losses. Illegal access to the network and system resources tend to prove fatal for the users. Huge amount of data loses can occur if the systems aren't secure. If the system is vulnerable, the hosts are at a huge risk- take the example of WannaCry ransomware attack of 2017. To prevent such anomalies, we need firewalls and advanced security tools like intrusion detection systems(IDS) to detect malwares and suspicious activities. These IDSs can be host-based- which examine the operating sysytem log files, and network based- which monitor the data over network packets or pcap files. Furthermore, malware detection can be done by analysing signatures and deviations from normal behaviour; to identify threats and analyse attacks effectively so that we can prevent future attacks.

Even though as we make advancements in the security field, the code breakers aren't shriveling, they are becoming more malignant. Many of the available security tools fail to identify a novel attack. Adaptive softwares which do dynamic analysis of network changes are needed.

Cuckoo Sandbox is one such tool which helps in analysing malware automatically. It is a free and open source software which provides detailed analysis of malicious activities. It creates a virtual network where the cuckoo host runs the malicious files on the guest machine and returns the behavioral report of the activities taking place on the guest machine. Log file reports and screenshots of guest machine can be viewed. The detailed report can then be used by machine learning algorithms to classify the malwares. All in all Cuckoo helps in testing malicious codes and view its reactions by running the files on the honeypot like guest machines.

Machine learning algorithms provide an automated method to classify malwares. From analysis reports provided by Cuckoo sandbox, suitable feature selection and extraction can be done. A part of data can be trained on various machine learning classifiers and remaining can be used to test the efficiency of the algorithms. NSL-KDD dataset is used for this purpose.

With fast changing nature of attacks, the available datasets need to be updated and made viable for future needs. This diversity of attacks make it difficult for the machine learning algorithms to reduce false positive rates. Real time and modern day attacks require a more active dataset.

2 MOTIVATION

The importance of secure and robust codes in this information revolutionized world is a constant source of motivation. The time when quantum computing would be a reality and threaten the cryptographic tools is another thought initiator. In this world where we depend so much on data, it's security should be two steps ahead of malicious actors. Analysing malware signatures and automating the process of classification using machine learning algorithms seems like a working solution.

Signature and anomaly based detection systems are useful in detecting static and known attacks. But to detect novel attacks, we need tools that can analyse malicious activities quickly. Cuckoo Sandbox is one such tool for testing malware activities.

With the emergence of newer attacks, the efficiency of algorithms trained on earlier datasets becomes unreliable. This thus makes it important to use honeypots and intelligent intrusion detection systems.

Thus, this report is made to understand the working of Cuckoo Sandbox tool and various machine learning algorithms used to analyse malware detection datasets.

3 LITERATURE SURVEY

3.1 Deep Learning Approach for Intelligent Intrusion Detection System

R. Vinayakumar, Mamoun Alazab, K. P. Soman, Prabaharan Poornachandran, Ameer Al-Nemrat, Sitalakshmi Venkatraman

The paper analyses and studies various machine learning algorithms for the publicly available datasets and compares the intrusion detection results with Deep Learning approach. The research focuses on host based and network based intrusion detection. A scalable framework architecture used to analyze big data. Binary and multiclass classifications of malwares are done using suitable feature selection of the DNNs.

Testing the algorithms on numerous datasets like KDDCup99, NSL-KDD, UNSW-NB-15, Kyoto dataset and WSN-DS is useful to determine the efficiency in varied situations. It is found that minimal feature selection of the multi-class DNN worked more efficiently than traditional machine learning algorithms.

The following table shows efficiency of various machine learning algorithms on available datasets:

Table 1: Results Table Sample

S.No	Data set	Algorithm	Attack	Accuracy
1	KDDCup 99	DNN 1 Layer	DoS	0.953
2	KDDCup 99	KNN	DoS	0.617
3	NSL-KDD	DNN 5 Layers	U2R	0.903
4	NSL-KDD	Decision Tree	U2R	0.882
5	UNSW-NB15	DNN 4 Layers	Worms	0.988
6	UNSW-NB15	Random Forest	Worms	0.988
7	WSN-DS	DNN 3 Layers	Flooding	0.987
8	WSN-DS	SVM-rbf	Flooding	0.956

3.2 A Novel Machine Learning Based Malware Detection and Classification Framework

Kamalakanta Sethi, Rahul Kumar, Lingaraj Sethi, Padmalochan Bera, Prashanta Kumar Patra

Changing nature of malwares makes it difficult to analyse and detect novel malicious data. So a new testing dataset is used by testing malicious files from VirusTotal and VirusShare sites using Cuckoo SandBox.

Cuckoo Sandbox is used to extract log files from malicious activities. Data set is created and various features are extracted using chi2, random forest classifiers. Cuckoo provides additional features in the reports which help in improving the efficiency of algorithms to suit new malwares.

A reduce in false positive rates is achieved and it is found that Decision Trees provide better results on analysis of the data.

Table 2: Malware Classification Results

S.No	Algorithm	Accuracy	Precision
1	K-Nearest Neighbors	0.965	0.97
2	Decision Tree	0.991	0.997
3	Support Vector Machine	0.867	0.88
4	Random Forest	0.882	0.90

4 PROBLEM DEFINITION AND SCOPE

4.1 Problem Definition

To set up a Cuckoo Sandbox environment and analyse behavior of malicious programs. To find the accuracy of various machine learning algorithms on the NSL-KDD dataset.

4.2 Scope

Cuckoo Sandbox creates a virtual network for testing malicious codes. Malwares are launched from the host machine on the virtual Windows XP guest machine. Cuckoo generates report logs and takes back the guest machine to a stable state. Cuckoo provides logs with additional features which may improve the efficiency of various machine learning algorithms.

The data from analysis can be used to check efficiency of various machine learning algorithms. In absence of a big dataset, algorithms are run on the known NSL-KDD datasets and results are found.

5 SYSTEM DESIGN

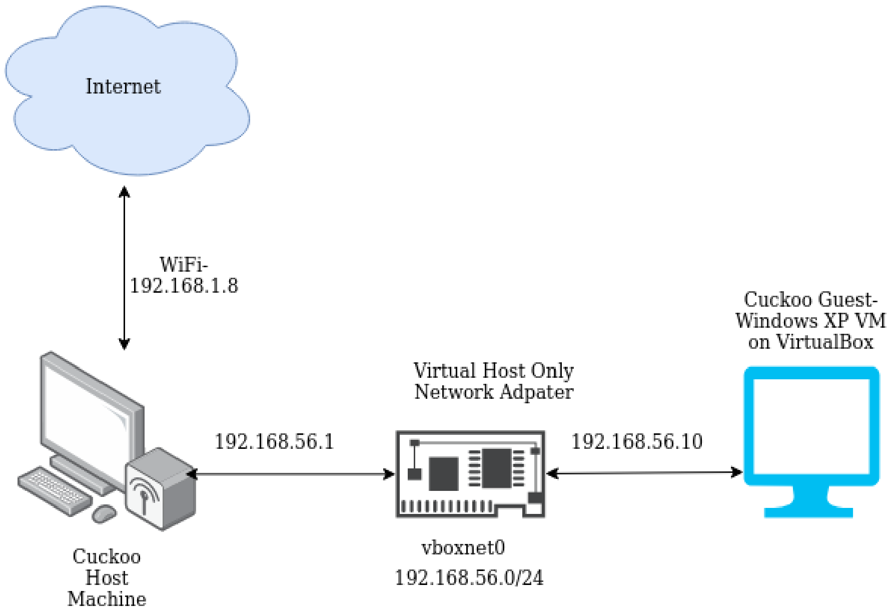


Figure 1: Cuckoo Sandbox Architecture

5.1 Architecture

Cuckoo Sandbox runs on the Linux host machine. The machine is connected to the internet. A virtual network is created and a Windows XP host machine is connected to network through a virtual host only adapter setting.

5.2 Working of Cuckoo Sandbox

- Start the Cuckoo host machine.
- Start the web interface of Cuckoo Sandbox
- Insert malicious file to run on guest
- Wait for the analysis report
- Check the report and screenshots
- Close the Cuckoo host

6 MACHINE LEARNING ALGORITHMS

6.1 K-Nearest Neighbors

It is a supervised discriminative discrete learning model. It is pretty simple and classifies a point by comparing to the majority of nearest k training points. Can be used to detect abnormal behaviors over wireless sensor networks.

6.2 Decision Trees

A supervised model of classification where leaf nodes contain labels and intermediate nodes are conditions. Repetitive checking is done against conditions to reach down to a label. Useful for signature based malware detection.

6.3 Support Vector Machines

Used for two classification problems. In this case normal behavior and attack. It converts multiple labeled data into a higher dimensional space. It provides margin for outliers.

6.4 Random Forests

It is similar to bagging. It creates large collection of de-correlated trees and then averages them out.

6.5 Neural Networks

Inspired by the biological neural networks. The input layer is activated and data is passed to hidden layers with hyper parameters. Multiclass output layer can be achieved.

7 METHODOLOGY

7.1 Workflow

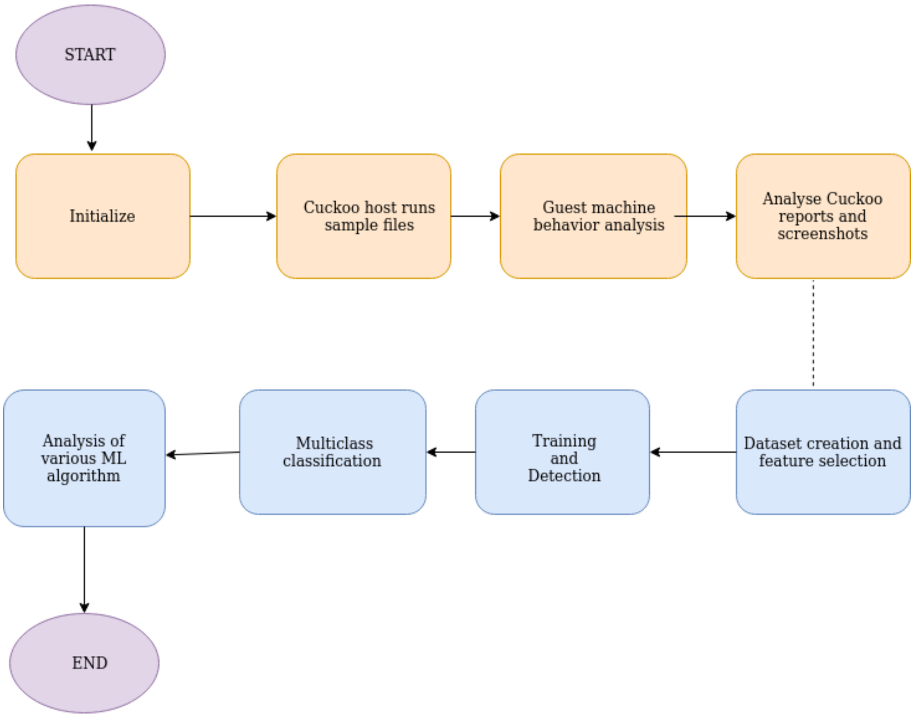

START
Initialize
Cuckoo host runs sample files
Guest machine behavior analysis
Analyse Cuckoo reports and screenshots
Dataset creation and feature selection
Training and Detection
Multiclass classification
Analysis of various ML algorithm
END

Figure 2: Workflow

7.2 Configuring the network

- Set up the Cuckoo installation on a different user other than main user on the Ubuntu machine.
- Create a virtual machine with Windows XP installation.
- Configure the network to create a virtual box network 192.168.56.0/24
- Setup the host ip as 192.168.56.1
- Setup the network configurations on guest virtual machine with ip 192.168.56.10
- Check if machines can ping.
- Now install suitable softwares on VM and install the python agent file from host to guest.
- Take the snapshot of stable VM
- Now try installing malicious files from host to guest through command line or Cuckoo web server.
- Wait for analysis report
- Check the report and screenshots to analyze the malicious activity.

7.3 Mathematical model

S = {q_0,q_f, A, B, g_i, | ϕ }

q_0: start state.
q_f: final state.

Let A be the input set consisting of:- A = D_i
where D is analysis data from cuckoo logs or available datasets like pcaps from NSL-KDD dataset. These are data vectors used from the datasets.

Let B be the output set consisting of:-
B = P,F where F $\in F_i$ is class defined as a malware class like U2R, DoS etc. P is the Normal behavior

Functions

g_i - Let 'k' be the function to detect the malware such that:-

k : input data vectors from training and testing datasets → B

i indicates various algorithms- KNN, RF, NN etc

Statistical measures

True Positive(TP): Actually normal behavior predicted as normal
True Negative(TN): Anomalous behavior predicted as a malware
False Positive(FP): Malware detected as a normal behavior
False Negative(FN): Normal behavior detected as malware

$$TruePositiveRate = \frac{TP}{TP + TN} \tag{1}$$

$$FalsePositiveRate = \frac{FP}{FP + TN} \tag{2}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

8 Results

8.1 Data

Table 3: Data Table

S.No	Data set	size
1	NSL-KDD Train	14.8 MB
2	NSL-KDD Test	2.7 MB

8.2 Implementation Results

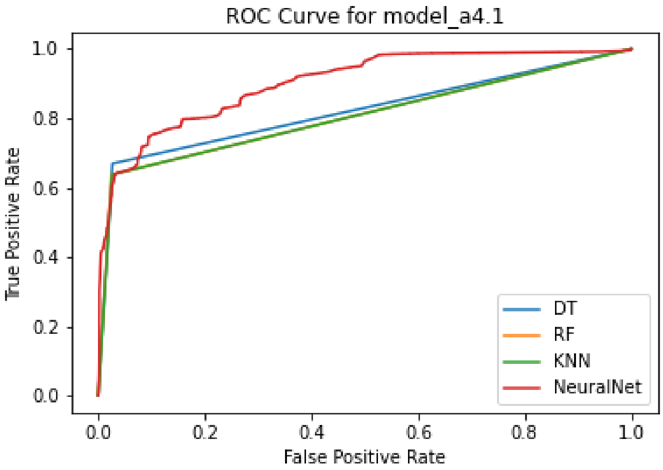
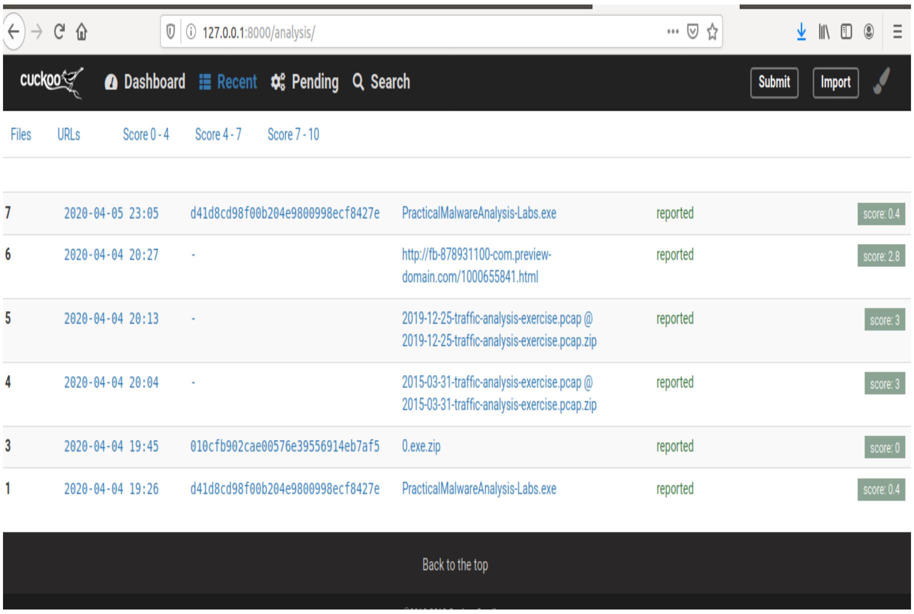
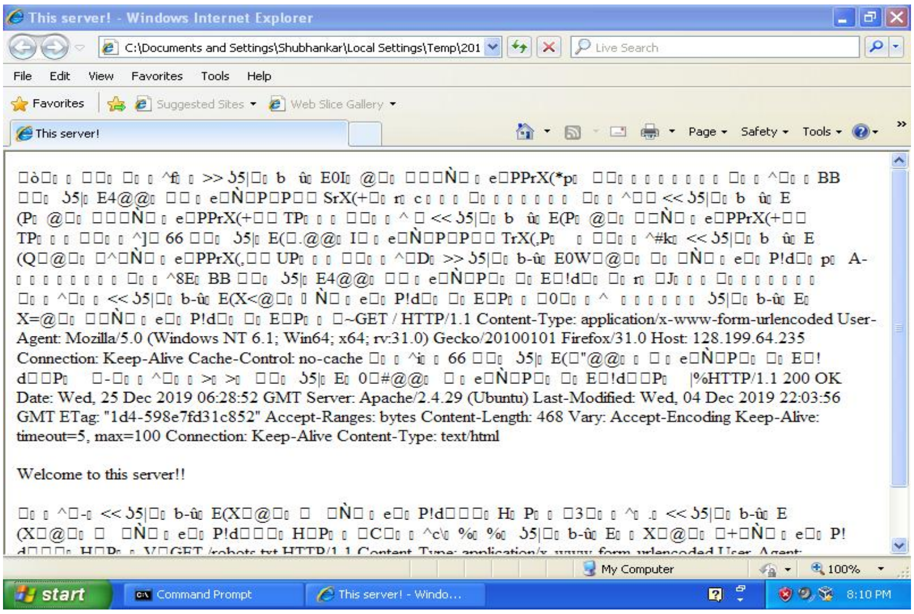


Figure 3: ROC curve for all models



7	2020-04-05 23:05	d41d8cd98f00b204e9800998ecf8427e	PracticalMalwareAnalysis-Labs.exe	reported	score: 0.4
6	2020-04-04 20:27	-	http://fb-878931100-com.preview-domain.com/1000655841.html	reported	score: 2.8
5	2020-04-04 20:13	-	2019-12-25-traffic-analysis-exercise.pcap @ 2019-12-25-traffic-analysis-exercise.pcap.zip	reported	score: 3
4	2020-04-04 20:04	-	2015-03-31-traffic-analysis-exercise.pcap @ 2015-03-31-traffic-analysis-exercise.pcap.zip	reported	score: 3
3	2020-04-04 19:45	010cfb902cae00576e39556914eb7af5	0.exe.zip	reported	score: 0
1	2020-04-04 19:26	d41d8cd98f00b204e9800998ecf8427e	PracticalMalwareAnalysis-Labs.exe	reported	score: 0.4

Figure 4: List of malwares analysed

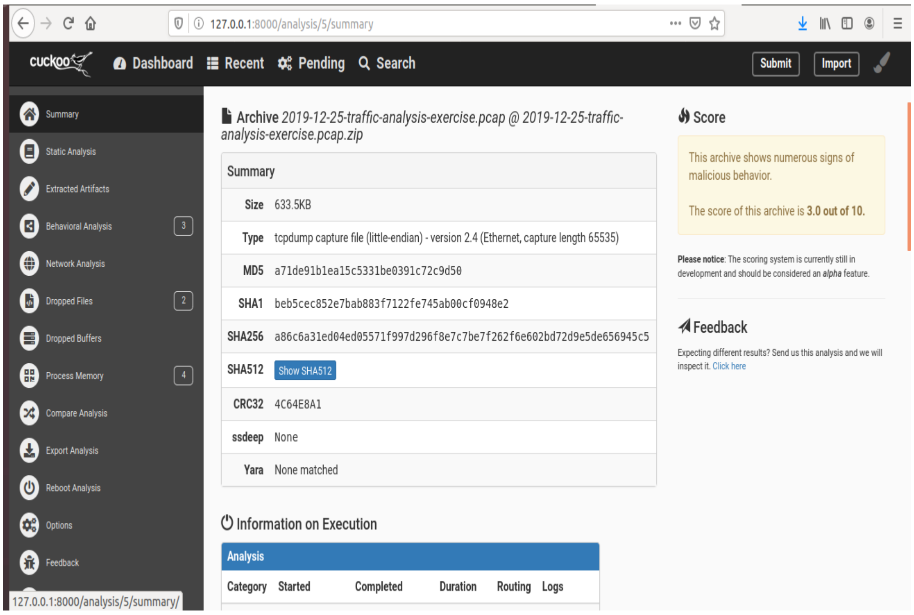


```
□ò□▯ ▯ □□▯ □▯ ▯ ^fî ▯ >> Ꝺ5|□▯ b ûₗ E0I▯ @□▯ □□□Ǹ□ ▯ e□PPrX(*p▯ □□▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ □▯ ▯ ^□▯ ▯ BB
□□▯ Ꝺ5|▯ E4@@▯ □□ ▯ e□Ǹ□P□P□□ SrX(+□▯ rₗ c ▯ ▯ ▯ □▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ □▯ ▯ ^□□ << Ꝺ5|□▯ b ûₗ E
(P▯ @□▯ □□□Ǹ□ ▯ e□PPrX(+□□ TP▯ ▯ ▯ □□▯ ▯ ^ □ << Ꝺ5|□▯ b ûₗ E(P▯ @□▯ □□Ǹ□ ▯ e□PPrX(+□□
TP▯ ▯ ▯ □□▯ ▯ ^]□ 66 □□▯ Ꝺ5|▯ E(□.@@▯ I□ ▯ e□Ǹ□P□P□□ TrX(,P▯ ▯ □□▯ ▯ ^#k▯ << Ꝺ5|□▯ b ûₗ E
(Q□@□▯ □^□Ǹ□ ▯ e□PPrX(,□□ UP▯ ▯ ▯ □□▯ ▯ ^□D▯ >> Ꝺ5|□▯ b-ûₗ E0W□@□▯ □▯ □Ǹ□ ▯ e□▯ P!d□▯ p▯ A-
▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯ □▯ ▯ ^8E▯ BB □□▯ Ꝺ5|▯ E4@@▯ □□ ▯ e□Ǹ□P□▯ □▯ E□!d□▯ □▯ rₗ □J▯ ▯ ▯ □▯ ▯ ▯ ▯ ▯ ▯ ▯ ▯
□▯ ▯ ^□▯ ▯ << Ꝺ5|□▯ b-ûₗ E(X<@□▯ ▯ Ǹ□ ▯ e□▯ P!d□▯ □▯ E□P▯ ▯ □0□▯ ▯ ^ ▯ ▯ ▯ ▯ ▯ ▯ ▯ Ꝺ5|□▯ b-ûₗ E▯
X=@□▯ □□Ǹ□ ▯ e□▯ P!d□▯ □▯ E□P▯ ▯ □~GET / HTTP/1.1 Content-Type: application/x-www-form-urlencoded User-
Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:31.0) Gecko/20100101 Firefox/31.0 Host: 128.199.64.235
Connection: Keep-Alive Cache-Control: no-cache □▯ ▯ ^i▯ ▯ 66 □□▯ Ꝺ5|▯ E(□"@@▯ ▯ □ ▯ e□Ǹ□P□▯ □▯ E□!
d□□P▯ □-□▯ ▯ ^□▯ ▯ >▯ >▯ □□▯ Ꝺ5|▯ E▯ 0□#@@▯ □ ▯ e□Ǹ□P□▯ □▯ E□!d□□P▯ |%HTTP/1.1 200 OK
Date: Wed, 25 Dec 2019 06:28:52 GMT Server: Apache/2.4.29 (Ubuntu) Last-Modified: Wed, 04 Dec 2019 22:03:56
GMT ETag: "1d4-598e7fd31c852" Accept-Ranges: bytes Content-Length: 468 Vary: Accept-Encoding Keep-Alive:
timeout=5, max=100 Connection: Keep-Alive Content-Type: text/html

Welcome to this server!!

□▯ ▯ ^□-▯ << Ꝺ5|□▯ b-ûₗ E(X□@□▯ □ □Ǹ□ ▯ e□▯ P!d□□□▯ H▯ P▯ ▯ □3□▯ ▯ ^▯ .▯ << Ꝺ5|□▯ b-ûₗ E
(X□@□▯ □ □Ǹ□ ▯ e□▯ P!d□□□▯ H□P▯ ▯ □C□▯ ▯ ^c\▯ %▯ %▯ Ꝺ5|□▯ b-ûₗ E▯ ▯ X□@□▯ □+□Ǹ□ ▯ e□▯ P!
d□□□▯ H□P▯ ▯ V□GET /robots.txt HTTP/1.1 Content-Type: application/x-www-form-urlencoded User-Agent:
```

Figure 5: Malicious behavior at Cuckoo guest VM



Archive *2019-12-25-traffic-analysis-exercise.pcap @ 2019-12-25-traffic-analysis-exercise.pcap.zip*

Summary	
Size	633.5KB
Type	tcpdump capture file (little-endian) - version 2.4 (Ethernet, capture length 65535)
MD5	a71de91b1ea15c5331be0391c72c9d50
SHA1	beb5cec852e7bab883f7122fe745ab00cf0948e2
SHA256	a86c6a31ed04ed05571f997d296f8e7c7be7f262f6e602bd72d9e5de656945c5
SHA512	Show SHA512
CRC32	4C64E8A1
ssdeep	None
Yara	None matched

Score

This archive shows numerous signs of malicious behavior.

The score of this archive is **3.0 out of 10.**

Please notice: The scoring system is currently still in development and should be considered an ***alpha*** feature.

Feedback

Expecting different results? Send us this analysis and we will inspect it. Click here

Information on Execution

Analysis					
Category	Started	Completed	Duration	Routing	Logs

Figure 6: Analysis report of Cuckoo Sandbox

9 CONCLUSION

Malware analysis and classification is of importance for securing information systems and networks. Cuckoo Sandbox tool helps in analysing the malwares effectively and quickly and provides vectors with new labels and features. These vector datasets are useful in classifying malwares with different machine learning algorithms to give better efficiencies.

References

[1] R. Vinayakumar, Mamoun Alazab, K. P. Soman, Prabaharan Poornachandran, Ameer Al-Nemrat, Sitalakshmi Venkatraman, “Deep Learning Approach for Intelligent Intrusion Detection System,” *IEEE Access, vol. 7, pp. 41525-41550, 2019.*

[2] Kamalakanta Sethi, Rahul Kumar, Lingaraj Sethi, Padmalochan Bera, Prashanta Kumar Patra,“A Novel Machine Learning Based Malware Detection and Classification Framework,” *International Conference on Cyber Security and Protection of Digital Services, Oxford, United Kingdom, pp. 1-4, 2019.*

[3] Cuckoo Sandbox-Automated Malware Analysis, ”Cuckoo Sandbox book” Available at: https://cuckoo.readthedocs.io/en/latest/ [Online]

[4] Trevor Hastie, Robert Tibshirani, Jerome H. Friedman, ”Elements of Statistical Learning- Data Mining, Inference, and Prediction” *Springer Series in Statistics* Available at: https://link.springer.com/book/10.1007/978-0-387-84858-7 [Online]

REVIEW AND VISIT LOG

PLAGIARISM REPORT