iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 **Issue:** VI **Month of publication:** June 2020

DOI: http://doi.org/10.22214/ijraset.2020.6015

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue VI June 2020- Available at www.ijraset.com

Malware Detection using Machine Learning and Cloud Computing

Dr. P. Indirapriyadarsini[1], Mohammed Uzair Mohiuddin[2], Mohammed Taqueeuddin[3], Ch Srikanth Reddy[4], T Koushik[5]
[1]*Associate Professor,* [2, 3, 4, 5]*Student, Department of Information Technology, Vardhaman College of Engineering, Hyderabad, Telangana, India.*

Abstract: ***Malware refers to malicious software program perpetrators dispatch to infect individual computers or an entire organization's network. It exploits target system vulnerabilities, inclusive of a worm in valid software program (e.G., a browser or internet software plugin) that can be hijacked. In the past few years, the malware industry has grown very hastily that, the organization of people make investments heavily in technology to avoid conventional protection, forcing the anti-malware groups/groups to build extra strong software's to discover and terminate those attacks. The major part of protecting a pc gadget from a malware attack is to identify whether or not a given piece of record/software program is a malware. Here, our model is mostly focused on byte files. While modeling we will go along with Random modeling first to get the worst log-loss and then go along with some other modeling like KNN, Logistic Regression, etc. And then examine the log-loss of every algorithm and then determine whether it's an excellent model. Finally, we deploy the machine learning model on the cloud (AWS) with the user-interface.***

I. INTRODUCTION

Malware is that the shortest term used for malicious software which may be a harmful malicious piece of code. The intention of malware is to harm the system or to steal the information from the system by exploiting vulnerabilities in the existing security infrastructure. Malwares are rapidly increasing with the passage of your time and that we can categorize malware in to different categories consistent with their behaviors. The malware are often a script, executable binary or the other piece of code, which have malicious intention. The most aims of malware are to realize access of system, disrupt system services, denial of service, and steal tip and destruction of resources. Sometime malware isn't defective software but some legitimate software can have malware inside it. Legitimate software usually acts as wrapper for malware. Downloading legitimate software from any website may download malicious software on itself. Mostly malwares are found in cracked software and pirated software [1].

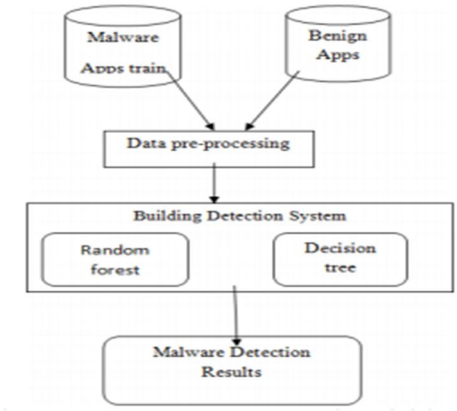
Malwares aren't only executable codes but sometimes they act as downloader for malware e.g. PDF and PHP link which gains control of system and download more malicious software to execute on system. Some software's gain control of system and do some legitimate work so we cannot classify them malicious [1].

Virus Total reports that 47.80% of malware is executable. So, the purpose of this article is to examine the binaries that can be executed. There are many sorts of malwares, which may be classified into Virus, computer virus, Adware, Worm and Backdoor. A number of malwares can't be classified into one category, because malwares have multiple characteristics which organize them in multiple categories and sometime, we called them generalize malware. Malwares are analyzed on basis of static also as dynamic features. Quite 2300 features are extracted from dynamic analysis and 92 features are extracted statically from computer file using PEFILE. Different dynamic features combinations are used for analysis. Four sorts of dynamic features are used for malware analyses which are Registry, DLLs, APIs and summary information. Machine learning is applied on these dynamic feature's combinations [1].

II. METHODOLOGY

The goal of the Malware Detection System Identification Method is to achieve immense malware disclosure and adaptability during the assessment of the total minute of the permissions [2]. Therein progress, our proposed system model scheme draws outs the list of permissions from the appliance folders or containers instead of concentrating on all the available permissions Significant permission process Detection for malware detection program explicitly aims to increase the share of malware discovery permissions. As an outcome, it excludes the demand to gauge permissions which has less impact for the malware discovery efficiency [2]. This is contained in of two main mechanisms: (i) Data pre- processing (ii) Building detection system a) Random Forest b) Decision tree. After Building detection system, Significant Permission Identification method for malware detection system report back to the malware detection results [2].

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue VI June 2020- Available at www.ijraset.com



Data pre-processing could be a knowledge processing system to involve remodeling data into a clear arrangement [2]. Real- world so as is typically unfinished, conflicting, along with/otherwise missing in bound behavior's otherwise trend, with is nearly certainly leaving to possess some errors. so, as pre-processing could be an evidence method of breakdown such trouble. Decision trees are classifier models inside which both hub of the tree speak to an examination on the traditional for the info set, with its children represent the result [2]. The youngster hubs speak to the top program of the actualities point. it's an overseen classifier show which use data by perceived mark to appearance the choice of tree with then the design is applied on the check data [2].

III. EXPERIMENTS AND RESULT

In this section, first, we present the model using different machine learning algorithms and choose the best one with accurate result. Second, we deploy our model on the cloud. The cloud Platform, we had chosen is AWS.

A. Feature Selection

The dataset is taken from Microsoft. It consists of 54 attributes. We need to perform feature selection in such a way that the most relevant features are selected in order to determine the legitimate from the malicious records.
There is an efficient way of selecting features, that is to identify the most interesting features and reduce the dimensionality of the data set. In order to achieve relevant features, we use the Tree-Based Feature Selection [3].

```
extratrees = ek.ExtraTreesClassifier().fit(X,y)
model = SelectFromModel(extratrees, prefit=True)
X_new = model.transform(X)
nbfeatures = X_new.shape[1]
print(nbfeatures)
14
```

The Tree-Based algorithm selected 14 important features among the 54.

B. Training the Model

The algorithms used to test the model are Decision Tree, Random Forest, Gradient Boosting, AdaBoost and Naive Bayes [4].
After testing the model with the above algorithms, we got to choose the best one with accurate results.

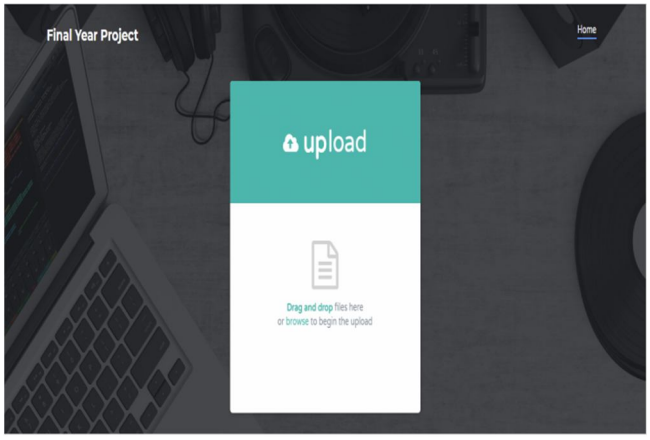
```
Now testing algorithms
GNB : 69.478450 %
DecisionTree : 98.960522 %
RandomForest : 99.351684 %
AdaBoost : 98.558493 %
GradientBoosting : 98.761318 %

Winner algorithm is RandomForest with a 99.351684 % success
```

Now that we got the efficient algorithm, we need to train the model. Once training is done, our model is ready to test any piece of software/file and predict whether it is legitimate or malicious [4].

C. User Interface

For the easy access of the model, we have come up with a web interface in order to check and predict whether the software/file is legitimate or malicious. The user needs to upload the file/software to determine the legitimacy of the file.

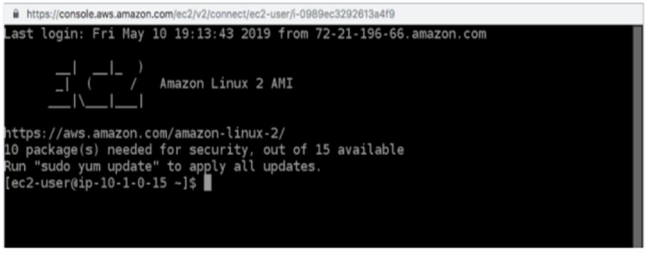


Once the file/software is uploaded, the user needs to click on the Start Detection button. Then the output is produced accordingly based on the file/software (i.e. legitimate or malicious).

D. Deployment of the Model

In order to deploy the model, we need to use the three important services of AWS. The services used are AWS Elastic Compute Cloud, AWS Elastic Beanstalk, AWS Elastic Container Registry.

Amazon Elastic Cloud (Amazon EC2) is a web service used to supply our model with computing capabilities. It is designed to make scaling of web computing easier for us [5].



AWS Elastic Beanstalk is service which is used to deploy and scale services and web applications. It reduces the complexity to manage our web Interface [5].

Amazon Elastic Container Registry (ECR) is one of the most unique service that allows us to easily store, manage and dockerize & deploy our model in terms of docker images. It is combined with AWS ECS, simplifying our development model to production workflow. The cloud provider manages the operation of server management [5].

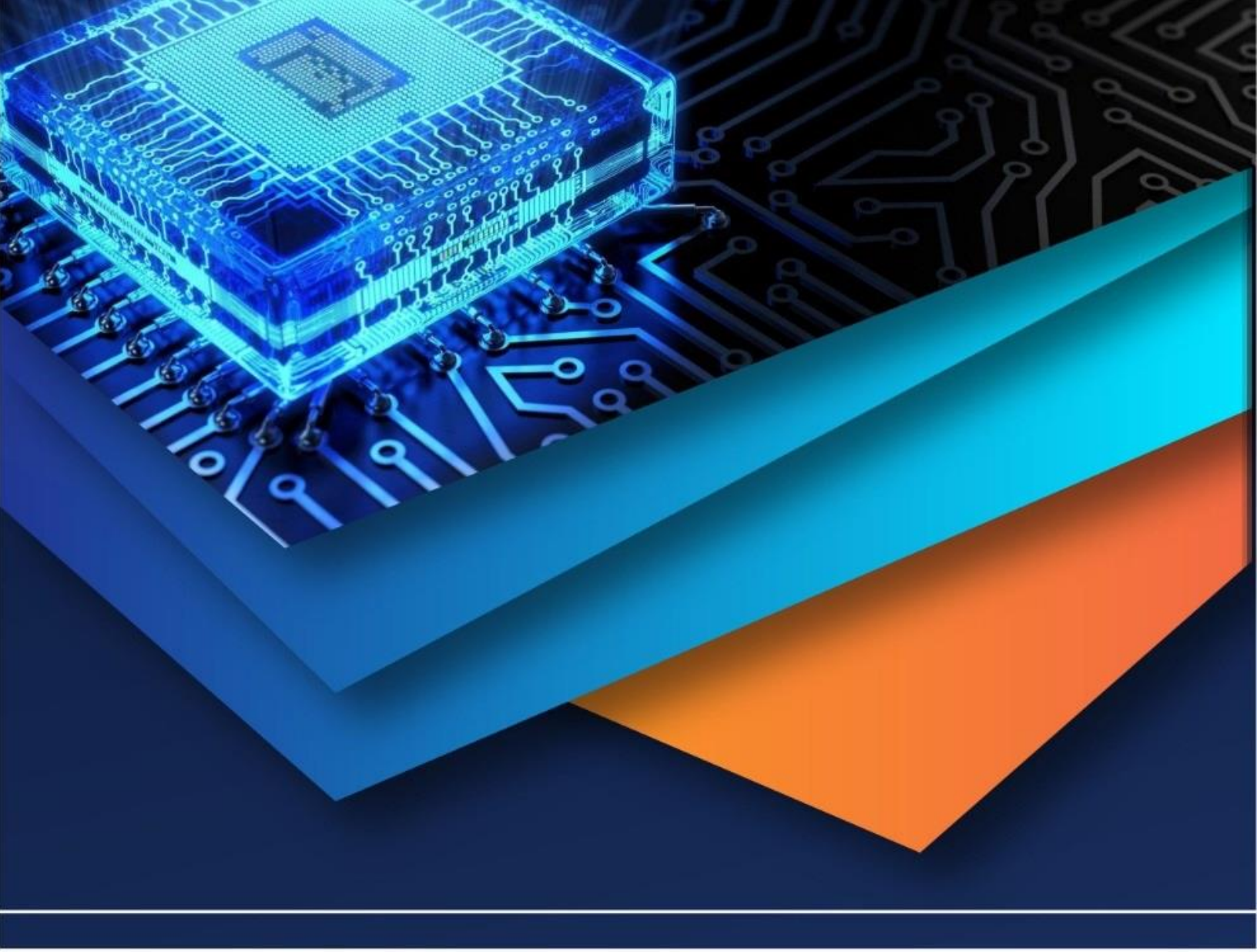
By using these services, we had deployed the model on the AWS. Now our model is accessed from anywhere on the internet. We made the model user-friendly to the user.

IV. CONCLUSION

In our project, we have come up with the unique solution by working with machine learning and cloud computing simultaneously in order to determine the legitimacy of the file/software. The standard detection methods may become obsolete for the files/software that are more flexible and dominating with the tough constraint of having a false positive rate. In our view, malware detection through machine learning cannot replace the standard detection methods that are used by various antivirus vendors, but it will be an added advantage for them.

REFERENCES

[1] M. Ijaz, M. H. Durad, and M. Ismail. Static and dynamic malware analysis using machine learning. In 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST), pages 687–691, Jan 2019.
[2] S.Lakshman Raju. “Identification Method for Malware Detection System." IOSR Journal of Engineering (IOSRJEN), vol. 09, no. 11, 2019, pp. 01-05.
[3] Liaw, A. and Wiener, Classification and regression by random Forest, 2002, R news, 2(3), pp. 18-22.
[4] C. D. Morales-Molina, D. Santamaria-Guerrero, G. Sanchez-Perez, H. Perez-Meana and A. Hernandez-Suarez, "Methodology for Malware Classification using a Random Forest Classifier," 2018 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC), Ixtapa, Mexico, 2018, pp. 1-6.
[5] A. Bedi, N. Pandey and S. K. Khatri, "Analysis of Detection and Prevention of Malware in Cloud Computing Environment," 2019 Amity International Conference on Artificial Intelligence (AICAI), Dubai, United Arab Emirates, 2019, pp. 918-921, doi: 10.1109/AICAI.2019.8701418.
[6] https://docs.aws.amazon.com/



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY