International Journal of Modern Agriculture, Volume 10, No.01, 2021
ISSN: 2305-7246

Malware Detection using Deep Learning

Dr. Anusha K,
AssociateProfessor, School of Computing, Science and Engineering(SCOPE),Vellore Institute of Technology,Chennai,India.

LakshayGrover,
Vellore Institute Of Technology, Chennai Campus,Chennai,India

Dr. S.Nachiyappan,
AssociateProfessor,School of Computing Science and Engineerint(SCOPE)
Vellore Institute Of Technology, Chennai.

Shivani Deshmukh,
Vellore Institute Of Technology, Chennai,Chennai,India

YASASVI Josyula,
Round Rock High School,201 Deep Wood Dr, Round Rock, TX 78681.

Abstract
Malware is a malicious software that someone can dispatch to infect individual computers or an entire organization's network. Once exploit it targets a particular system and can destroy them completely,one example of such malware is a bug in a software. A malware once affected can cause a lot of problems and also daily operations of any type of system.Malware is not a single entity but it comes in various forms and each has it's own way of entering and corrupting the system for example Virus, Worms, Trojans, Ransomware are various types of malware having different families. Windows has been prone to malware attack from a long period of time. The aim of research is to identify the existence of malware in a system.

Keywords: Malware, Worm, Trojans, Ransomware, Virus

1. INTRODUCTION

According to a latest report by Symantec it shows that the rate of creation of malware was 41% with around 400 different malwares. Internet security threat report says that there is an 36%increase in the attack with each day4,500 new attacks to the victim's data. In 2015 26% increase in new malware samples have been reported. Due to increase in number of such attacks day by day it has become necessary to obtain an efficient method which could stop malware from entering into user's system and helps in preventing user's data. Malware is not something that has evolved today or some time before but it has a big history. In 1968 something like malware came into picture. Basid and Ajmal 2 siblings in Pakistan created malware. They supplanted boot segment of floppy circle with a duplicate of virus. The actual boot part is moved to another area and consequently everytime you embed a floppy, it would taint the drive. The aim of this malware was to tackle a few issues as opposed to offering damage to a framework. The first malware began with the goal of hurting a framework was Omega infection which influenced the boot area and harm one's framework. Another malware in type of infection came into picture was Walker, when the infection enter your framework it shows a man strolling in your framework after at regular intervals and no client input is acknowledged till infection is taken out from the framework. Another infection like Walker was Ambulance infection in which an emergency vehicle was running starting with one side of screen then onto the next. Infection such as

International Journal of Modern Agriculture, Volume 10, No.01, 2021
ISSN: 2305-7246

Virus, took a major structure when we saw the new and most hazardous Casino infection coming into picture.Casino virus copies FAT into it's own memory and gives message to user that if he wants to use his PC he has to play a game and he cannot restart his PC because then he would never able to get his FAT. To win the user needs to get minimum 3 Euros from 5 chances else he will lost his DOS else if he wins then PC could be used normally. Then came into picture something known as mutation engine which is not a virus itself but it adds functionality to viruses that make it difficult to be detected by antivirus. The first virus released which was intended for Microsoft Windows was WinVir. It looks for any .exe file files in the current working directory. It cuts the middle of the file, moves it to end and place its own code in middle and removes itself from the file that it was executed from and attempts to restore it to its original condition.Monkey was one another virus that was infecting master boot record of hard drives and floppies, and millions of malwares came into picture later on. After malware began threating a number of the systems all over the world, it was ordered into various sorts based on how malware enters and assaults a specific devide or system. One such type of malware is Adware that deliver ads automatically to everyone. Here and there we see pop-ups coming in type of promotions, the adware can include spyware in such advertisements for which if a client opens that, it releases some malicious code into the user's system and perform some malicious activity into the system. Another type of such harmful malware are known as Bots. Bots are used to spread out some kind of malicious activities and they can also be utilized for spreading of DDoS attacks. The bug is another stream produces because of an undesired result. Bugs are generally an after effect of issues in source code or an undesired result. When spread it can totally freeze a system. Another kind of malware is Ransomware that holds a system under its control and ask for some kind of amount in order to release the system, it holds the framework under its influence by encoding the framework documents and requests some sum from to make the framework free and recapture their admittance to the PC. A rootkit is another kind of malware that once capture a system by the aggressor, it can get to the framework distantly from some other area, and subsequently it can get to, alter, take data effectively, once introduced it is exceptionally hard to get kept from Rootkit in light of the entire control moves to the assailant of one's framework and it turns out to be hard to dispose of it. Another such type of harmful virus is Trojan Horse. It would appear that an ordinary document on the web and deceives client of download the record yet contains malware in genuine for which client doesn't know about, once downloaded.

III. BACKGROUND AND ARCHITECTURE

The computer system is structed in a way where the task and resources are carried in sequence order. By altering this sequence, the chain of task can be destroyed. Like human beings a computer system can also get affected by the entity and these entities are virus, trojan, malware, Ransomware.
These malicious programs can self-replicates itself by copying it into another program and its process get initiated by an executable code or documents. A successful breach of these virus can cause serious issues for the user as it infects the resources and modify or delete the main functions. These viruses operate in two ways the first way is to lands on new operating system and then begin to replicate. The second way is to play ideal until a executable malicious code get executed. Virus has many different forms and it affects different systems in different ways. Boot Sector is one of the most common type of virus which infects the master boot record and Mostly it spreads through
removable media such as pen drive. Direct action virus also spreads through removable media. Resident Virus is a type of virus which is difficult to identify and removed from the system. Multipartite Virusspreads through multiple ways. It infects executable files and the boot system. It is difficult to identify Polymorphic virus with a traditional antivirus program. This is because the polymorphic viruses changes its signature pattern whenever it replicates. Overwrite virus is another such type of Virus which infects and delete files. The only possible mechanism to remove the virus is to permanently delete the infected files and thereby the user lost all his content contained in the file. Such type of virus is spread through emails and hence it is difficult to identify them.These viruses and malware usually execute in bits form so that the structured program can be altered and the resources,

International Journal of Modern Agriculture, Volume 10, No.01, 2021
ISSN: 2305-7246

task can be carried out by the encoded virus entity. In this way a system program can altered, destroyed and the sensitive data can transfer for illegal usage. One of the best example code of such virus which harmed millions of system is fork bomb virus. The code of fork bomb virus is as follows.

```
#include <unistd.h>

int main(void)
{
for(;;)
fork();
return0;
}
```

Different architecture describes how the different types of malware functions to harm a system. The architecture is an indication of how a malware could enter a system and starting from entering to destruction, the whole process is described for example architecture of different types of malwares are as above

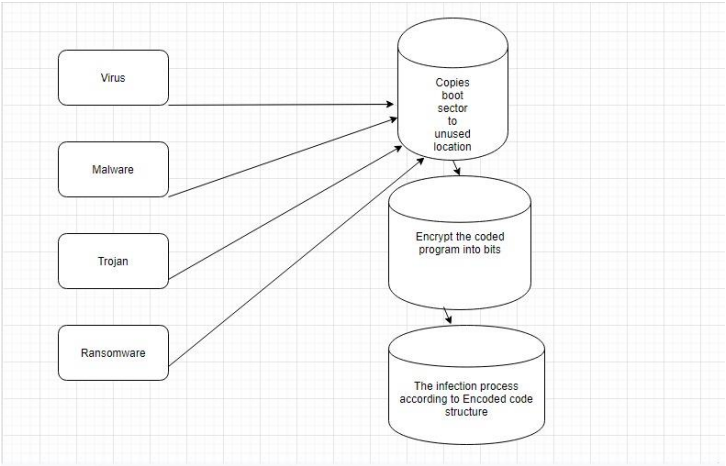

Figure 1

The following architecture(Figure 1) shows how different malware combine and enter into the system, different malware and ransomware copies the exact location which is the boot location to any other location which is unused and after that encrypt the coded program into small bits, and finally the coded program structure is executed and it affects the whole system. A virus enters mostly through a file for example a file in windows system is called as a Windows executable file. The system build helps to identify that weather the files in the system are begin or Malware. Deep learning ensemble models are build to identify the files as malware or begin. A small architecture of the system is shown above. In the following architecture(Figure 1) it is observed that different types of malwares such as Virus, Malware, Trojan, Ransomware copies boot sector in unused locations and the code is then encrypted in bits format. It further infects the PC with respect to the encoded code structures. Figure 2 is an architecture of system, in which a windows executable file is converted into malware or begin based on it's nature. The deep learning ensemble categorizes a .exe file as malware or begin.

International Journal of Modern Agriculture, Volume 10, No.01, 2021
ISSN: 2305-7246

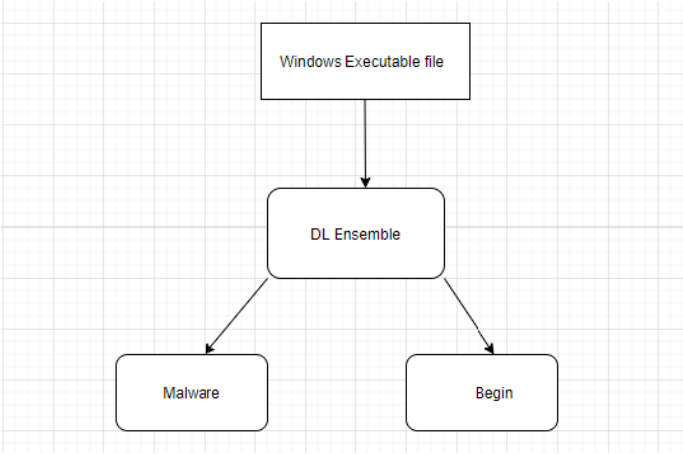

Figure 2

11. RELATED WORK

SHAILA SHARMEEN1, SHAMSUL HUDA 1, (Member, IEEE),
JEMAL H. ABAWAJY1, (Senior Member, IEEE), WALAA NAGY ISMAIL AND MOHAMMAD MEHEDI HASSAN 2, (Member, IEEE) Proposed Malware Detection on Industrial Mobile IOT Devices.
Published on 13 March 2018:
This paper investigates the endeavours in regards to malware dangers went for the gadgets conveyed portable IoT devices and organizes them with related recognition systems. In this execution, about static, dynamic, and mixture exam
based on informational index, highlight extraction was applied. Procedures, include choice strategies, recognition techniques, and the precision accomplished by these strategies. In this manner, it recognizes suspicious Programming interface calls, framework calls, and the authorizations that are removed and chosen as highlights to identify malware, and thus creating applications for modern IoT systems. After the investigation it was discovered that machine learning approaches are normally used to arrange malware and.

KeXu*, Yingjiu Li, and Robert H. Deng Proposed-
ICCDetector: ICC-Based Malware Detection on
Android on 03 February 2016
Most of malwares, most of the time are not identified because not all malware detectors are not able to detect malwares on the boundaries of an application. To address this test, ICC Detector was proposed. ICCDetector yields a location display subsequent to preparing with an arrangement of generous applications and an arrangement of malwares, what's more, utilizes the prepared model for malware identification. The execution of ICCDetector is assessed with 5,264 malwares, also, 12,026 benevolent applications. After physically examining false positives, it was found that 43 new malwares from the kindhearted dataset, and lessen the number of false positives to seven. All the more essentially, ICCDetector finds 1,708 more "progressed malwares" than the benchmark, while misses 220 "clear malwares" which can be effortlessly identified by the benchmark. For the identified malwares, ICCDetector further characterizes them into five recently characterized malware classes. It likewise gives fundamental investigation of ICC examples of considerate applications and malwares.

Mayank Jaiswal, Yasir Malik, FehmiJaafar Proposed
Android Gaming Malware Detection Using System
Call Analysis on 07 May 2018

International Journal of Modern Agriculture, Volume 10, No.01, 2021
ISSN: 2305-7246

Android working frameworks have turned into a prime focus for assailants as a large portion of the market is as of now ruled by Android clients. The circumstance deteriorates when clients unwittingly download or side load cloning applications,
particularly gaming applications that resemble kind-hearted recreations. In this paper, we present, a dynamic Android gaming malware identification framework dependent on framework call examination to order malignant and genuine diversions. We played out the dynamic framework call examination on ordinary and noxious gaming applications while applications are in execution state. Our investigation uncovers the and contrasts among considerate and malware amusement framework calls and demonstrates how powerfully dissecting the conduct of vindictive movement through framework calls amid run time makes it less demanding and is more compelling to recognize noxious applications. Trial investigation and results demonstrates the proficiency and adequacy of our approach.

Vaibhav Rastogi, Yan Chen, and Xuxian SHAMSUL Jiang Proposed," Catch Me If You Can: Evaluating Android Anti-Malware Against Transformation Attacks" on 11 November 2013.

This paper analysis the cutting-edge business versatile enemy of malware items for Android and test how safe they are against different normal confusion systems. Such an assessment is vital for not just estimating the accessible guard against versatile malware dangers, yet in addition proposing powerful, cutting edge arrangements. It created Droid Chameleon, a precise system with different change procedures, furthermore, utilized it for the examination. The results on 10 noticeable business antagonistic to malware applications for Android are alarming: none of these contraptions is protected against essential malware change techniques. Besides, a prevailing piece of them can be unimportantly squashed by applying slight change over known malware with little effort for malware makers. Finally, considering the outcomes, possible answers for improving the present state of malware distinguishing proof was proposed on Pacifically they, assessed ten enemy of malware items on Android for their versatility against malware changes. The discoveries utilizing changes of six malware tests demonstrate that all the hostile to malware items assessed are defenceless to normal avoidance. At long last, they investigated conceivable manners by which the present circumstance may be enhanced and cutting-edge arrangements might be created.

III. Proposed Work

After conducting the survey and understanding how different researchers have proposed the same method using different techniques of methodologies. It helped me in understanding that how the researchers, research on this particular report starting from where a malware evolves to how it enters into a system and how to detect it. To detect malware, it is necessary to know from where it comes from and how it enters. For my implementation I have used Deep learning ensemble which give me different outputs, on my data set 10 PE files were collected as 5 for malware,5 for begin. The and malware samples were collected from ipvoid.com and begin samples were collected from my own pc. The samples collected were pre-processed using DL ensemble models. In the first model, the PE record bundle was utilized to discover the passage purpose of the code. The recurrence of opcodes was discovered to locate the main 3 most regular opcodes. The chief model was set up on a mathematical vector port of words taken from the crude bytes of PE documents. Words were gotten by deciphering every bite of a record as an utf-8 encoded character, paying little brain to its arranged encoding, and joining characters to frame words, delimiting words with whitespace characters, for instance, spaces and tabs. The NLP (Natural Language Processing) strategy of highlight hashing was utilized to numericize these words.
The RF model uses the Capstone disassembly engine's Python package to disassemble instructions in the main block of PE files' machine code, using the file package to find the address of the entry point for this code. The entire dataset is disassembled, one file at a time, and the total frequency of each RF

International Journal of Modern Agriculture, Volume 10, No.01, 2021
ISSN: 2305-7246

counted to find the top-3 most frequent codes. Then the dataset is disassembled again, and the relative frequency of these top-3 codes within each sample is saved as a vector. This representation of the sample PE files is visualized in figures 1 and 2. The strings model's preprocessing decodes the raw bytes of sample files as UTF-8 characters, regardless of their intended encoding then tokenizes the resulting string into a vector of words using Keras 'text_to_word_sequence'. This splits the string into words when encountering whitespace and removes punctuation and special characters, then the 'hashing_trick' function is used to perform feature hashing. This hashing results in a sequence vector of integers with each unique value representing a unique word within the vocabulary of the entire corpus of word sequences generated from the dataset. Typically for a DLNN to train on samples, their representation must have a fixed size so these integer sequences are all truncated or zero-padded to a length of 10 not too long so that most samples are densely represented. Lastly these 10 value long integer sequences are reshaped into a 100x100 matrix, normalized between 0 and 1, multiplied by 255, and rounded to the nearest integer. This creates a 100x100 pixel 8-bit greyscale image which serves as a discretized representation of the first 5,000 words of a PE file's bytes decoded as UTF-8 strings. This discretization of input data helps classifiers to learn faster and more 25 accurately, as well as reducing the storage requirements of saving representations of these files to a hard-drive, requiring only a byte per word. This image representation is chosen to take advantage of the effectiveness of CNNs in classifying long sequences of data, as discussed in the literature review, and so the architecture of the strings model is that of a CNN. A full description of the architectures of both the strings and other models can be found in the previous subsection of this current section under the heading 'Design, Building, Programming, and Testing.

The ensuing model was prepared on an undeniably humbler number of highlights which is a probability presence of codes inside the get together code of document tabs. The recurrence of all of these fundamental 3 codes was assessed and normalized into the probability of its appearance by isolating it by an all-out number of codes. The other model takes its information bunches of relative frequencies of the principle 3 codes for given PE records, and feeds them through a movement of 5 totally related/thick covered layers, with the first of these layers being included some fake neurons, and each resulting thick layer having an enormous bit of the neurons of the past layer.
The strings model, takes a group of 100x100 pixel 8-piece greyscale pictures, each containing an induced depiction of the underlying 10 words tokenised from UTF-8 deciphering of the crude bytes of test PE documents.

1V. CONCLUSION

The conclusion of the above research and reading my literature survey papers is learning about the evolution of malware, different types of malware and how can DL Ensemble model could classify a windows executable file as malware and begin. This system helps in security of individual devices or devices of an organization. It achieved a high degree of accuracy of 90% on around 30 unseen malware sets.

V. REFERENCES

[1] Ki-Hyeon Kim, Mi-Jung Choi*," Android Malware DetectionusingMultivariate Time-Series Technique"

[2] Mayank Jaiswal, Yasir Malik, FehmiJaafar, "Android Gaming Malware Detection Using System Call Analysis"

[3] XIONG Ping1, WANG Xiaofeng2,5, NIU Wenjia3, ZHU Tianqing4, LI Gang, **"**Android Malware Detection with Contrasting Permission Patterns"

International Journal of Modern Agriculture, Volume 10, No.01, 2021
ISSN: 2305-7246

[4] SHAILA SHARMEEN1, HUDA 1, (Member, IEEE),
JEMAL H. ABAWAJY1, (Senior Member, IEEE), WALAA NAG ISMAIL2,AND MOHAMMAD MEHEDI HASSAN 2, (Member, IEEE),” Malware Threats and Detection for Industrial Mobile-IoT Networks”

[5]Shahrear Iqbal; Mohammad Zulkernine,”SpyDroid: A Framework for Employing Multiple Real-Time Malware Detectors on Android”

[6] Om Prakash, Samantray;Satya Narayan Tripathy, Susanta Kumar Das,”A study to Understand Malware Behavior through Malware Analysis”

[7] Fernando C. Colon Osorio, HongyuanQiu, Anthony Arrott, “Segmented-Sandboxing- A novel approach to malware polymorphism detection”

[8] N. Moses Babu, G. Murali, “Malware detection for multi cloud servers using intermediate monitoring server”

[9] M.Yeo,Y.Koo,Y.Yoon,T.Hwang,J.Ryu,J.Song,C. Park, “Flow-based malware detection using convolutional neural network”

[10] Byeong Kil Lee, Jordan Pattee, “Implications for Hardware Acceleration of Malware Detection”