MALWARE DETECTION USING MACHINE LEARNING AND PERFORMANCE EVALUATION

Ms. PATHAN JUVERIA NAJNEEN[1], Ms. KANDI NIKHITHA[2], Ms. ALURU SAI DURGA PRANATHI[3], MS. THADIBOINA POOJITHA[4], Ms. PAPPULA MADHAVI [5].

1.B. Tech, Vijaya Institute Of Technology For Women, Enikepadu, Vijayawada, Andhrapradesh, India, Juveria17786@Gmail.Com

2. B. Tech, Vijaya Institute Of Technology For Women, Enikepadu, Vijayawada, Andhrapradesh, India

3. B. Tech, Vijaya Institute Of Technology For Women, Enikepadu, Vijayawada, Andhrapradesh, India

4. B. Tech, Vijaya Institute Of Technology For Women, Enikepadu, Vijayawada, Andhrapradesh, India

5. Assistant Professor, Computer Science And Engineering, Vijaya Institute Of Technology For Women, Enikepadu, Vijayawada, Andhrapradesh, India,Pappulamadhavi06@Gmail.Com

ABSTRACT

Malware analysis is a crucial part of any cyber protection system. Several studies using machine learning techniques have been conducted in the previous ten years on both static and dynamic analysis. The goal of malware developers has shifted from merely fame to political espionage or monetary gain, therefore the virus is likewise evolving in terms of shape and techniques of infection. Targeted malware is one of the most recent varieties of malware, and little research has been done on it. The amount and sophistication of targeted malware, which is a subset of advanced persistent threat (APT), have increased recently.The harmful role played by targeted cyber attacks (through targeted malware) in undermining the online social and financial systems is on the rise. APTs are made to steal corporate or governmental secrets and/or hurt corporate or governmental interests. Targeted malware is challenging for antivirus, IDS, IPS, and bespoke malware detection solutions to identify. For the purpose of deploying APTs, attackers use attractive social engineering tactics with one or more zero day vulnerabilities. Coupled with these, the recent appearance of Crypto Vault and Ransomware poses major concerns to both individuals and organizations/nations. In this study, we contrast different machine-learning methods for examining malware, concentrating on static analysis.

1. INTRODUCTION

Malware is an acronym for "malicious software," which includes viruses, trojan horses, worms, and other types of harmful software. These programmes offer a wide range of capabilities, including the ability to steal, encrypt, or delete sensitive data, alter or hijack standard computer operations, and monitor computer activity. Display user consent.Computer viruses are one type of malware.It is typically a programme that is installed against the user's will and has the potential to harm a computer's operating system as well as its hardware (physical) components.Consequences brought on by the virus include file destruction and size changes.Erase everything on the disc, including any formatting.The file allocation table is destroyed, rendering it difficult to access data from the drive.A variety of safe but unsettling sound and graphic effects; a slowing down of the computer's operation until it crashes; and worms.Computer worms are malicious software applications that spread by using computer-to-computer communication. Worms and viruses have characteristics in that they both have the ability to reproduce, although worms do it on other computers rather than locally. I propagated to other systems using computer networks.Internet worms, instant messaging worms, and email worms are some examples of computer worms.Sharing files via a network.

Trojan horses

Trojan horses are "masked" programmes that attempt to open up security holes in the operating system so a user can access it. Trojans lack the ability to replicate themselves like computer viruses do.

Trojan horses can be categorised into the following groups:

Backdoors: enable the attacker to access the victim's computer remotely through the Internet; Password stealers: applications that capture passwords (read data from the keyboard and store them in files that can be read later by the attacker or can be sent directly to the e-mail account). logical bombs: when specific conditions are fulfilled, these Trojans are capable of doing actions that jeopardise system security;

Denial of Service tools are programmes that transmit specific data sequences to the target audience, which is typically a website, with the aim of stopping that audience's Internet services.

•The ransomware

Malware known as ransomware prevents the victim from using the computer and demands payment in exchange for access. The award and the official justification for the victim's need to pay are based on the virus kind. Some ransomware variants assert that the payment must be paid in order to avoid penalty from a government agency (often the FBI or a local agency), while others assert that this is the only way to unlock encrypted data. Consequences of ransomware include their ability to encrypt sensitive user data.

Has the ability to remove certain documents, multimedia items, and any other files that house crucial data. Moreover, they might attempt to remove crucial system or other programme components.

• Threats from ransomware can be used to steal authentication names, passwords, priceless personal documents, identity data, and other private information. They can also quickly stop an anti-virus, anti-spyware, or other software's activity by blocking its processes and turning off crucial system services. 7th Root Kit A root kit software application typically takes advantage of a flaw in the host system to get full access to a system, modify it, and then exploit its resources covertly.can change the Linux system's "pHs" utility, which shows active processes, so that it does not show the root kit process.

It has the ability to conceal specific files, generally their own, from antivirus software scans. Cyber dangers that fall under the category of spyware include software designed to infect PC systems and subsequently start illicit operations. these. The majority of the time, the functionality of these dangers depends on the goals of the people who created them. For example, some spyware dangers have the ability to gather sensitive data (such as login names, passwords, and other personally identifiable information) and send it to the people who created it using covert internet connections. These are used to follow individuals.and keep track of the websites they visit the most, as well as the actions they take there. Spyware can also result in a rise in spasms because this information is typically used for marketing and promotional purposes by various third parties.

What purposes does malware serve?

To steal confidential data. Personal information, like login credentials, passwords, financial information, and other similar data, is of interest to these programmes. They can also keep tabs on the user's online activities, record their web browsing patterns, and communicate all of this information to a distant server.

• Display offensive creative. Pop-up advertising can be displayed in great numbers via spyware. That behaviour is more frequently related to adware parasites.

Forcibly diverting visitors to dubious or dangerous websites. Moreover, certain spyware attacks have the ability to modify web browser settings, including the default search engine and home page.

Add a tonne of links to the victim's search results, directing them to the locations you want them to go (third party spyware sites, websites and other associated fields).

Make necessary adjustments to the system's settings. Performance problems may result from these changes, which may also degrade overall security.

Using backdoors to connect to a compromised machine. The majority of spyware threats have the ability to let hackers covert remote access to the system.

A decline in the system's overall performance that leads to instability.

2.RELEATEDWORK

The first versions of Malware were primitive; they infested various machines through floppy disks. With the evolution of Networking and the maturation of the Internet, malware authors have adapted their malicious codes to take full advantage of this new communication environment. Below is a brief overview of the evolution of malware over time.

1.1.1 The years 1971-1999

- 1971-Creeper: An experiment designed to test how a program can move between computers.

- 1974-Wabbit: A program that multiplies itself at an accelerated pace, until the speed of the system slows down, the performance is measured, the system is reduced and eventually collapses.

- 1982-Elk Cloner: Written by a 15-year-old child, Elk Cloner is one of the first viruses, and widespread, to multiply itself and display a short "poem" to the infected person: “It will get on all your disks; it will infiltrate your chips; Yes, it's a Cloner!”

- 1986-Brain Boot Sector Virus: Considered the first virus to infect MS-DOS computers.

- 1986 — PC-Write Trojan: Malware disguised as one of the oldest Trojans as a popular program called "PC-Writer." Once on a system, it deletes all files of a user.

- 1988 — Morris Worm: Infected a substantial percentage of computers connected to ARPANET, the predecessor of the Internet, which brought the network to its knees in 24 hours. This Worm marked a new beginning for malicious software.

- 1991 — Michelangelo Virus: The virus was designed to erase information from hard drives on March 6, the birthday of the famous Renaissance artist.

- 1999 - Melissa Virus: used Outlook addresses from infected machines and was sent to 50 people at once.

1.1.2 The years 2000-2010

- 2000 – ILOVEYOU Worm: the worm infected about 50 million computers. The damage caused major corporations and government agencies, including portions of the Pentagon and the British Parliament, to shut down their e-mail servers. Worms have spread globally and cost more than $ 5.5 billion in damage.

- 2003 – SQL Slammer Worm: One of the fastest spreading worms of all time, SQL Slammer infected nearly 75,000 computers in ten minutes. The worm has had a major effect worldwide, slowing down worldwide Internet traffic by denial of service.

- 2004 – Caber Virus: Although this virus has caused some damage, it is noteworthy because it is widely recognized as the first cell phone virus.

- 2005 – Koobface Virus: One of the first cases of malware to infect PCs and then spread to social networking sites. If rearranged, the letters in "Koobface" are old, and you get "Face book". The virus has also attacked other social networks such as MySpace and Twitter.

- 2008 – Conficker Worm: A combination of the words “configure” and “ficker”, this sophisticated worm has caused some of the 11 worst damage observed since Slammer appeared in 2003.

1.1.3 2010- present

- 2010 – Stuxnet Worm: Shortly after its release, security analysts openly speculated that the malware was designed to explicitly attack Iran's nuclear program and include the ability to affects hardware and software. The incredibly sophisticated worm is considered to be the work of a whole team of developers, making it one of the most intensive malware resources create**d to date.**

- 2011 — Zeus Trojan: Often detected for the first time in 2007, the author of the Trojan Zeus released the code to the public in 2011, giving a new life to malware. Sometimes called the Zbot, this Trojan has become one of the most successful pieces of botnet software in the world, impacting millions of machines.

- 2013 – Crypto locker: had a significant impact globally and helped fuel the ransom ware era. • 2014 – Back off: Malware designed to compromise Point-of-Sale (POS) systems to steal credit card data.

- 2016 – Cerberus: One of the most prolific crypto-malware threats. At one point, Microsoft found more company PCs infected with Cerberus than any other family of ransom ware.

- 2017 – WannaCry Ransom ware: Exploiting a vulnerability first discovered by the National Security Agent, WannaCry Ransom ware has knelt down a number. systems in Russia, China, the United Kingdom and the United States, blocking access to data and demanding redemption or loss of everything. The virus has affected at least 150 countries, including hospitals, banks, telecommunications companies, warehouses and many other industries.

3.LITERATURE SURVEY

Joseph Redmon's You Only Look Once: Unified, Real-Time Object Detection. Their earlier research focused on utilising a regression approach to find items. In this study, the YOLO algorithm was presented to obtain high accuracy and good forecasts [1]. Juan Du's Knowledge of Object Detection Using CNN Family and YOLO. In this study, they examined the efficacy of object detection families such CNN and R-CNN, and created the YOLO technique to improve efficiency. By Matthew B. Blaschko, "Learning to Localize Objects using Structured Output Regression." The topic of this essay is object localization. To get around the limitations of the sliding window method, they adopted the bounding box method in this case.

Trojans, spyware, and viruses—oh my! The road to coverage in the Internet's Oz is the yellow brick road

Author: Tort Trial & Insurance Practice Law Journal Roberta D.

Every business faces a cyber risk. The headlines support the truth that cyberattacks are becoming more frequent, sophisticated, and massive than ever before. They also transcend both geographical and industry borders. Regulations concerning data privacy and security are expanding as significant cyberthreats are making daily headlines. Addressing and reducing cyber risk is a primary priority for businesses all over the world as a result of the surge in data security breaches, denial of service attacks, and other attacks as well as data loss.. It is abundantly evident that network security cannot completely handle the problem of cyber risk on its own because neither a firewall nor an impenetrable security system exist. A company's overall strategy to handle, mitigate, and optimise protection against cyber risk can greatly benefit from the use of insurance. The Securities and Exchange Commission is aware of this fact. The SEC's Division of Corporate Finance has released recommendations on cyber security disclosures under the federal securities laws in response to "increasing frequent and serious cyber events." According to the guidance, businesses "should examine the appropriateness of their disclosure relating to cyber security risks and cyber events on a continuous basis" and that "acceptable disclosures may include" a "[d] description of applicable insurance coverage," among other things.•An enhanced Android malware detection method based on permission-based features and an evolving hybrid neuro-fuzzy classifier (EHNFC)

Alter Alta her Tasha is the author

Several kinds of attackers have been drawn in by the rising popularity of Android devices and consumers. By using code obfuscation techniques, malware developers produce new versions of malware from older ones. The exponential rise in the production of malware varieties may have been facilitated by obfuscated malware. Obfuscated malware is difficult to detect since behavior-based detectors cannot reliably identify it, and signature-based malware detectors can readily avoid it.As a result, an effective method for detecting obfuscated malware in smart phones running Android is required. There aren't many malware detection methods that can find obfuscated malware in the literature on Android malware classification. Nevertheless, these malware detection methods lacked the ability to enhance their effectiveness through rule-based learning and evolution. This research suggests an evolving hybrid neuro-fuzzy classifier (EHNFC) for Android malware classification using permission-based features, based on the idea of developing soft computing systems. The suggested EHNFC is not only capable of detecting malware that has been obscured using fuzzy rules, but it can also adapt its structure by learning new malware detection fuzzy rules to increase the accuracy of its detection when used to detect more malware apps.In order to do this, an adaptive approach for updating the radii and centres of clustered permission-based features was added to an evolving clustering method for adapting and evolving malware detection fuzzy rules. By increasing cluster convergence and producing rules that are better suited to the input data, this change to the developing clustering algorithm raises the proposed EHNFC's classification accuracy. The suggested EHNFC outperforms a number of cutting-edge obfuscated malware classification algorithms in terms of false negative rate (0.05) and false positive rate, according to testing results (0.05). The findings also show that the proposal, in terms of accuracy, identifies Android malware more effectively than existing neuro-fuzzy systems (namely, the adaptive neuro-fuzzy inference system and the dynamic evolving neuro-fuzzy system).Detecting malware variants

Author: Malware variant detection, Alzarooni, KMA (2012). UCL doctoral thesis (University College London). a green open access

Malware programmes, such as Trojan horses, worms, and viruses, are rampant everywhere. Research and data indicate that malware's effects are deteriorating. The main weapons in the fight against malware are malware detectors. For the purpose of identifying malicious software within a computer system, the majority of commercial anti-malware scanners keep a database of malware patterns and heuristic signatures. To create new stealth versions of their malicious programmes, malware authors use semantic-preserving code modification (obfuscation) techniques. Today's detection methods struggle to identify malware variants because they primarily focus on syntactic characteristics while ignoring the semantics of malicious executable programmes.To combat this new security danger, a strong malware detection technique is needed. In this thesis, we present a new methodology that, by examining the semantics of known harmful code, overcomes the limitation of current malware detection methods. The creation of a semantic signature, slicing analysis, and test data generation analysis are the three main analysis methodologies that make up the methodology. This method's main component is to approximate the semantics of malware code and to create signatures that may be used to recognise malware variants that may be disguised but are nonetheless semantically comparable. A programme test input plus semantic traces of known malware code make up a semantic signature. Finding a balance has been the main obstacle in creating our semantics-based method to malware variant detection.Finding a balance between increasing the detection rate (i.e. matching semantic traces) and performance, with or without accounting for the effects of obfuscation on malware variants, has proven to be the main issue in creating our semantics-based method to malware variant detection. In order to improve the creation of semantic signatures, we develop slicing analysis. We support our trace-slicing method with a theoretical finding that demonstrates the slicer's accuracy. Our malware detector's proof-of-concept implementation shows how the semantics-based analysis method could enhance existing detection technologies and make it harder for malware authors to create new viruses. Exploring programme semantics for the selection of an appropriate component of the semantic signature is another crucial aspect of this thesis, and for this, we present two new theoretical findings.Classifying unknown packing techniques for malware detection using entropy analysis Authors: Mahn Soo Choi, Hongzhe Li, Heejo Lee, and Munkhbayar Bat-Erdene

Around 80% of all currently active malware is packed, a number that has been steadily increasing. In this study, we suggest a system for categorising the regardless of whether they are malicious software or safe programmes, packing algorithms of provided unknown packaged executables. The entropy values of a certain executable are first scaled, and the entropy values of a specific region in memory are then represented symbolically. Symbolic aggregate approximation (SAX), which is known to be efficient for massive data conversions, is the foundation of our suggested approach. Second, we use supervised learning classification techniques to categorise the distribution of symbols,Our trials with a set of 324 packed benign programmes and 326 packed malicious programmes, each with

19 packing algorithms, show that our method can identify the packing algorithms of provided executables with a high accuracy of 95.35%, a recall of 95.83%, and a precision of 94.13%. On the basis of incremental aggregate analysis and SAX representations of the entropy values, we suggest four similarity metrics for identifying packing techniques. The fidelity similarity measurement, which is from 2 to 13 greater than the other three metrics, shows the best matching result among these four metrics, with an accuracy rate ranging from 95.0 to 99.9%.A framework for intrusion detection for mobile phones based on specifications Authors: Sencun Zhu, Zhi Xu, and Ashwin Chaugul

Malware is increasingly becoming more prevalent on mobile devices as a result of the mobile market's rapid rise. One characteristic of a lot of malware that is frequently discovered on mobile devices is that it always tries to access private system functions on the device in a covert and sneaky method. For instance, the malware might secretly communicate with the device's audio peripherals or send messages automatically without the user's knowledge or consent. We introduce SBIDF, a Specification Based Intrusion Detection Framework, which uses keypad or touch screen interrupts to distinguish between malware and human action, to detect the illegal malicious activities.In the proposed framework, we use an application independent specification to describe the typical behaviour pattern. This specification is written in Temporal Logic of Causal Knowledge (TLCK), and it is enforced to all third-party applications on the mobile phone during runtime by watching the inter-component communication pattern among crucial components. Our analysis of the behaviour of simulated real-world malware demonstrates our capability to identify all types of malware.

that tries to access private information without the user's consent. Also, the SBIDF has a very low overhead (20 secs), which makes deployment in the real world highly possible.

•In-execution dynamic malware analysis and detection by mining information in process control blocks of Linux OS

Author: FarrukhShahzad, M.Shahzad, MuddassarFarooq

Run-time behavior of processes - running on an end-host - is being actively used to dynamically detect malware. Most of these detection schemes build model of run-time behavior of a process on the basis of its data flow and/or sequence of system calls. These novel techniques have shown promising results but an efficient and effective technique must meet the following performance metrics: (1) high detection accuracy, (2) low false alarm rate, (3) small detection time, and (4) the technique should be resilient to run-time evasion attempts. To meet these challenges, a novel concept of genetic footprint is proposed, by mining the information in the kernel process control blocks (PCB) of a process that can be used to detect malicious processes at run time. The genetic footprint consists of selected parameters - maintained inside the PCB of a kernel for each running process - that define the semantics and behavior of an executing process. A systematic forensic study of the execution traces of benign and malware processes is performed to identify discriminatory parameters of a PCB (task_struct is PCB in case of Linux OS). As a result, 16 out of 118 task structure parameters are short listed using the time series analysis. A statistical analysis is done to corroborate the features of the genetic footprint and to select suitable machine learning classifiers to detect malware. The scheme has been evaluated on a dataset that consists of 105 benign processes and 114 recently collected malware processes for Linux. The results of experiments show that the presented schemeachieves a detection accuracy of 96% with 0% false alarm rate in less than 100ms of the start of a malicious activity. Last but not least, the presented technique utilizes partial knowledge that is available at a given time while the process is still executing; as a result, the kernel of OS can devise mitigation strategies. It is also shown that the presented technique is robust to well known run-time evasion attempts.. [8] A state-of-the-art survey of malware detection approaches using data mining techniques Author: Airless Sauri , Rahil Hosseini Data mining techniques have been concentrated for malware detection in the recent decade.= The battle between security analyzers and malware scholars is everlasting as innovation grows. The proposed methodologies are not adequate while evolutionary and complex nature of malware is changing quickly and therefore turn out to be harder to recognize A state-of-the-art survey of malware detection approaches using data mining techniques

Author: Alireza Souri1* and Rahil Hosseini

Data mining techniques have been concentrated for malware detection in the recent decade. The battle between security analyzers and malware scholars is everlasting as innovation grows. The proposed methodologies are not adequate while evolutionary and complex nature of malware is changing quickly and therefore turn out to be harder to recognize.

•Design and implementation of a malware detection system based on network behavior

Author: Xue, L., & Sun, G. (2015). Design and implementation of a malware Detection system based on network behavior. Security and Communication Networks, 8(3), 459-470.

This paper presents a malware detection method based on network behavior evidence chains. The proposed new method will detect the specific network behavior characteristics on three different stages as connection establishment, operating control, and connection maintenance.

4.IMPLEMENTATION STUDY

Methods for Malware Analysis:

The creation of efficient methods for identifying infected files requires malware analysis. This analysis entails looking at the goals and operations of a malware programme. In order to describe how malware functions and what impact it has on the system, three separate analysis methodologies are used, although their time and skill requirements are significantly different.

Static evaluation

Another name for it is code analysis. In other words, malicious software code is examined to learn more about how it operates. Tools for disassembly, decompilation, debugging, and source code analysis are used in this reverse engineering technique. As this technique has no execution time overhead, we shall use it exactly as is.

Dynamic evaluation

Also known as behavioural analysis. During execution in a secluded setting like a virtual machine, simulator, or emulator, infected files are examined. Following file execution, the system's behaviour and impacts are observed.

Hybrid research

This method is suggested as a way to get beyond static and dynamic analysis' constraints. In order to improve the comprehensive analysis of malware, it first analyses the specification of the signature for every malicious code and then combines it with the other behavioural parameters. Owing to this method, hybrid scanning is more advanced than static and dynamic scanning.

Signaturedetection

Using signature-based detection, a threat's individual identifier is established and made public so that the threat can be discovered. to be determined later. In the context of a virus scan, this may be a special code template that is attached to a file or something as straightforward as the hash of a malicious file. Known. The file may be flagged as malicious if that particular pattern or signature is found again. As malware has evolved, its creators have started to use novel methods like polymorphism to alter the pattern each time an object spreads from one system to another.

Detection of behaviour

In contrast to signature-based scanning, which reveals The heuristic scan [5] employs rules and/or algorithms to seek for commands that may suggest intent or evil if the signatures detected in the file match those of a known malware database. Certain heuristic scanning techniques can identify malware using this technique without the need for a signature. Because of this, the majority of antivirus products use signature and heuristic methods to detect any malware that may attempt to avoid detection.

Determine features

A variation on behavior-based detection known as feature detection aims to reduce the frequency of false alarms that are typically **associ**ated with it. Program characteristics that characterise the security behaviour of crucial programmes serve as the foundation for characteristic detection. Instead of ostensibly recognising specific attack patterns, this entails watching programme executions and spotting deviations from the specification and its behaviour. This method is similar to that for detecting anomalies, but it differs in that it is based on characteristics created by hand to record the system's behaviour rather than using machine learning methods**.**

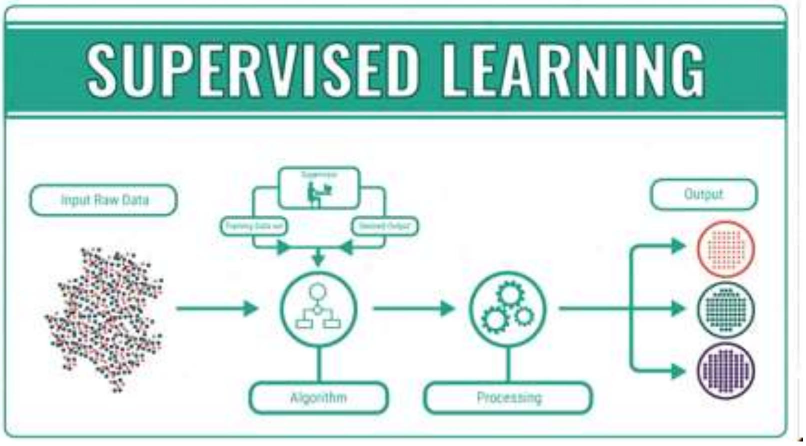
Machine learning

Machine learning is a class of methods that enables software applications to predict outcomes considerably more accurately without being specifically designed. Building algorithms that take input data and apply statistical analysis to forecast output data while output data is updated as much input data become valid is the fundamental tenet of machine learning. Machine learning procedures are comparable to data mining and predictive modelling procedures. For both, it is necessary to look for specific patterns by date and modify software operations accordingly. A lot of people are also aware with machine learning because of online purchasing and the customised marketing they see.

The two types of machine learning algorithms are supervised and unsupervised, respectively.

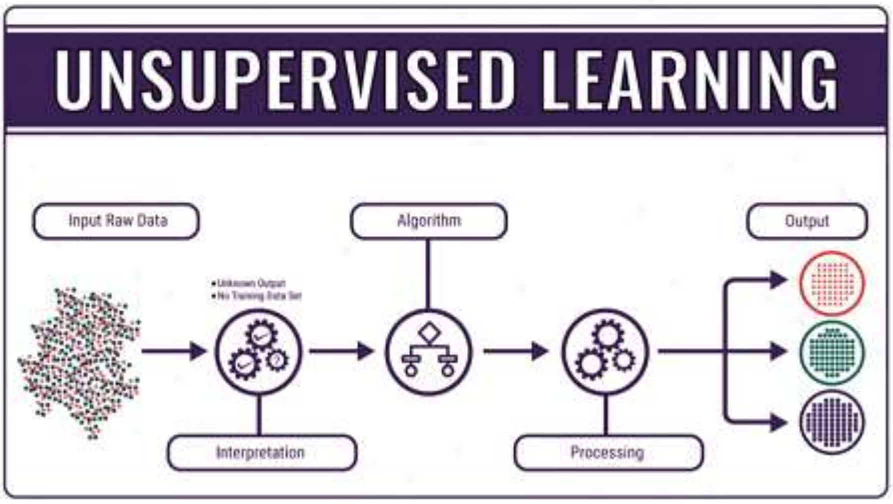
Supervised algorithms

They need a data researcher or data analyst with machine learning expertise to provide the required input and output data as well as provide feedback on the precision of the predictions, which is especially important during algorithm training. Data scientists choose which traits or factors the model should take into account while making predictions. The algorithm will apply what it has learned to new data after the training is over. Regression and classification issues are within the category of supervised learning challenges. When the output variable is a category, such as "red" or "blue" or "illness" and "no disease," a classification difficulty exists.Regression: When the output variable has a real value, such "dollars" or "weight," a regression problem exists. Recommendation and time series prediction are two typical sorts of challenges built on top of classification and regression, respectively. Popular supervised machine learning algorithms include the following: Regression issues using linear regression. Random forest for regression and classification.



algorithms without supervision

They don't require instruction using output data. Instead, they analyse the data and draw conclusions using a technique called deep learning. Compared to supervised algorithms, which are used for tasks like image recognition, speech-to-text conversion, and natural language creation, unsupervised and learning algorithms, commonly referred to as neural networks, are employed for more complicated tasks. In order for these neural networks to function, a large amount of data and training examples must first be combined. Next, tiny correlations between various variables are automatically found. Associates can use the algorithm to interpret new data after it has been trained. Due of their extensive training requirements, these algorithms are only practical in the information era. They are referred to as unsupervised learning because, in contrast to the supervised learning described above, there are no right answers and no teacher.



4.**PROPOSEDWORK AND ALGORITHM**

In order to successfully discriminate between malicious files and clean files while attempting to reduce the amount of false positives, we suggest a flexible framework in which one may make use of various machine learning methods.

The concepts underlying this framework were put through a scaling-up procedure that enables us to work with very big datasets of malware and clean files after being successfully tested on medium-sized datasets of malware and clean files.

Benefits include the ability to identify harmful files before they are executed, ease of use, quick identification, and detection of polymorphic malware.

5. METHODOLOGIES

Supervised Machine Learning algorithm can be broadly classified into Regression and Classification Algorithms.

Regression Algorithms: Regression finds correlations between dependent and independent variables. Regression algorithms therefore aid in the prediction of continuous variables such as real estate prices, economic trends, climatic patterns, oil and gas prices (a crucial job in today's world!), etc.

Linear Regression

- Decision Tree Regressor
- Random Forest Regressor

Linear Regression:

One of the simplest and most widely used Machine Learning methods is linear regression. It is a mathematical technique for performing predictive analysis. For continuous/real/numeric variables like sales, salary, age, and product price, among others, linear regression generates predictions. The linear regression algorithm, also known as linear regression, demonstrates a linear connection between a dependent (y) and one or more independent (y) variables. Given that linear regression demonstrates a linear connection, it can be used to determine how the dependent variable's value changes as a function of the independent variable's value. The connection between the variables is represented by a sloping straight line in the linear regression model.

Decision Tree Regression:

Decision tree regression can be used to perform non-linear regression in machine learning. The decision tree regression algorithm's primary job is to divide the information into more manageable chunks. The values of all data points that relate to the issue statement are plotted using the subsets of the dataset. This algorithm divides the data collection into decision and leaf nodes, producing a decision tree. When the data collection has not undergone enough change, ML experts favour this model. One should be aware that even a small shift in the data can have a significant impact on the decision tree's structure. Additionally, one should avoid over-pruning the decision tree regressors. since there won't be enough remaining end nodes to make the forecast. One should not overly prune the decision tree regressors in order to have numerous end nodes (regression output values). This develops a model with a tree-like structure that can forecast data in the future and generate useful ongoing output.

Random Forest Regression Algorithm:

Another popular method for non-linear regression in machine learning is random forest. A random forest employs multiple decision trees to predict the outcome as opposed to decision tree regression (single tree). With the help of this algorithm, a decision tree is constructed using k randomly chosen data points from the provided dataset. The worth of any new data point is then predicted using a number of decision trees. A random forest algorithm will forecast multiple output values because there are numerous decision trees. To determine the final result for a new data point, you must discover the average of all the predicted values. This occurs as a result of the numerous decision trees that must be mapped using this method.more processing capacity. Trees run parallel; it's a bagging method, not a boosting technique. i.e., there is no interaction between these trees as you construct trees.

Classification Algorithms:

An algorithm called classification discovers functions to categorise the dataset into groups based on different criteria. On the basis of what it learns from the training dataset, a computer programme divides the data into different groups.

- Logistic Regression
- Decision Tree Classifier
- Random Forest Classifier
- KNN - K Nearest Neighbour

Logistic Regression:

Under the category of supervised learning is logistic regression. Using a predetermined collection of independent variables, it is used to predict the categorical dependent variable. In a categorical dependent variable, the outcome is predicted by logistic regression. As a result, the result must be a discrete or categorical number. Rather than providing the precise values of 0 and 1, it provides the probabilistic values that fall between 0 and 1In logistic regression, we fit a "S" shaped logistic function, which forecasts two maximum values, rather than a regression line. (0 or 1). By using various kinds of data to categorise the observations, logistic regression can be used to quickly determine the most efficientfactors that were used for classification.

Classification Algorithm (Decision Tree):

A supervised learning method called a decision tree can be used to solve classification and regression problems, but it is typically favoured for doing so. It is a tree-structured classifier, where internal nodes stand in for a dataset's features, branches for the decision-making process, and each child node for the classification result. The Decision Node and Leaf Node are the two elements in a decision tree. While Leaf nodes are the results of decisions and do not have any additional branches, Decision nodes are used to make decisions and have numerous branches. It is a graphical representation for obtaining all feasible answers to a decision or issue based on predetermined conditions. Since it functions like a tree, it is known as a judgement tree. begins with the base node and grows on additional branches to create a structure resembling a tree. The CART algorithm, which means for Classification and Regression Tree algorithm, is used to construct a tree. It can be applied to classification and regression.

Random Forest Algorithm:

Popular machine learning algorithm Random Forest is a part of the guided learning methodology. In order to increase the dataset's predictive accuracy, a classifier called Random Forest uses multiple decision trees on different segments of the input data. It can be applied to ML issues involving both classification and regression. It is founded on the idea of ensemble learning, which is a method of combining various classifiers to address complex issues and enhance model performance. Even for the large dataset, it operates effectively and predicts the outcome with a high degree of accuracy. When a significant amount of the data is absent, accuracy can still be maintained.**K-Nearest Neighbour (KNN)**

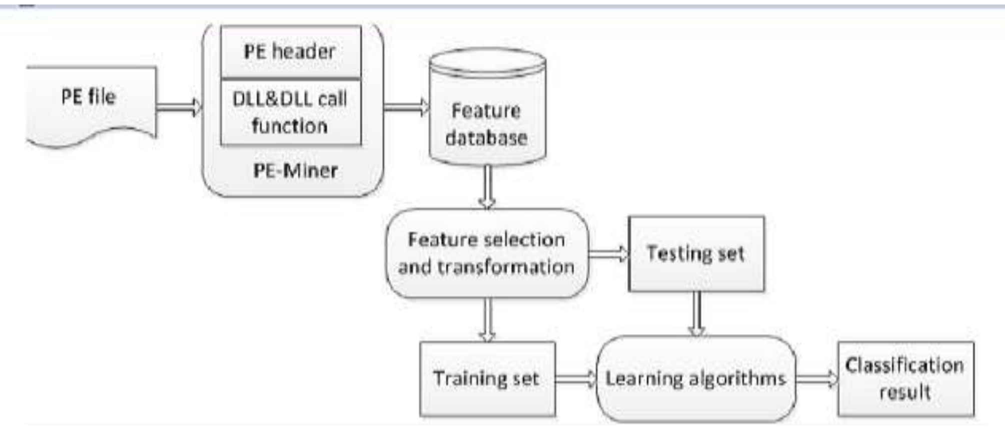
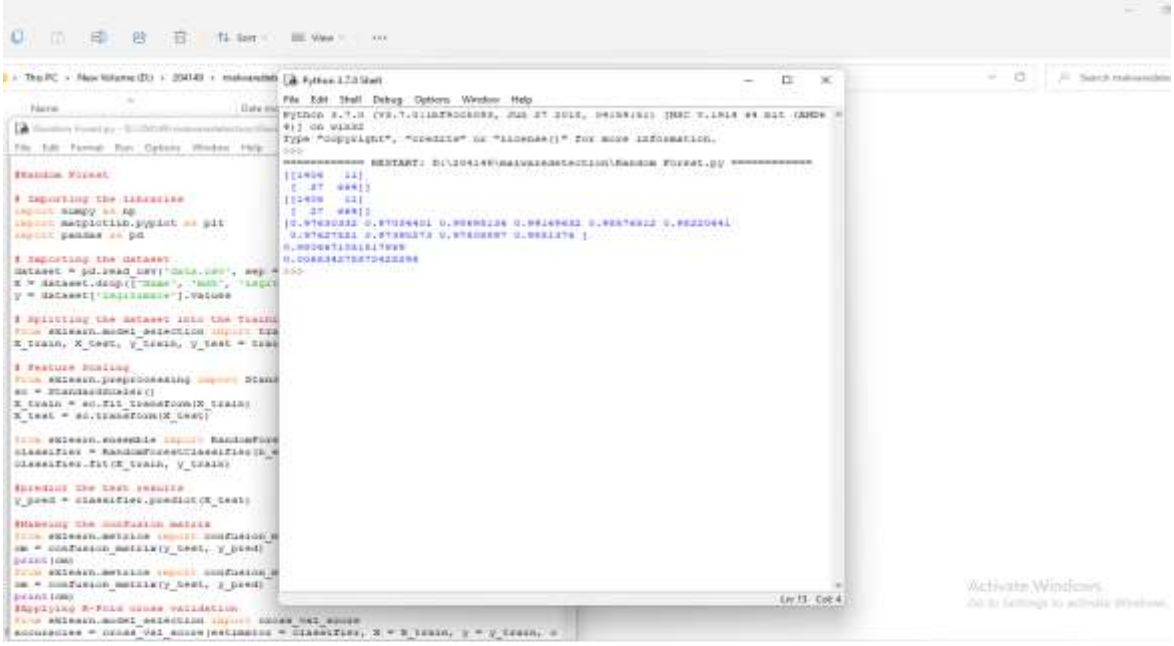


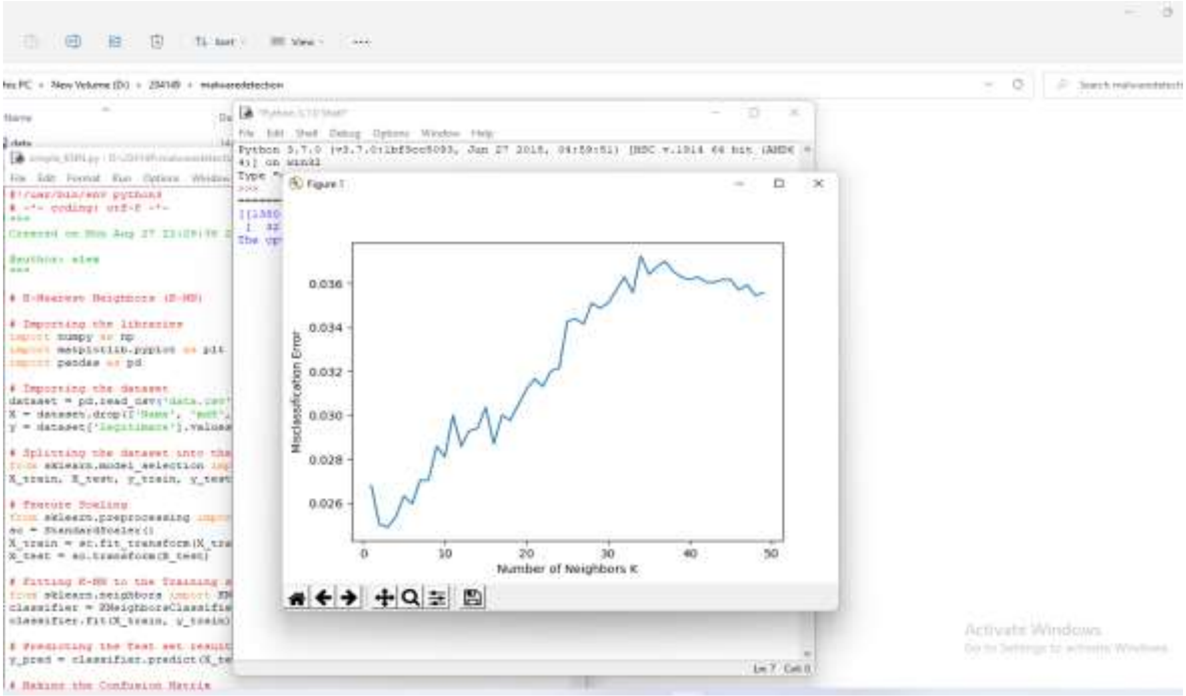
Fig.1:Propose Architecture

6. RESULTS



```
#Random Forest

# Importing the libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# Importing the dataset
dataset = pd.read_csv('data.csv', sep =
X = dataset.drop(['Name', 'md5', 'legit
y = dataset['legitimate'].values

# Splitting the dataset into the Traini
from sklearn.model_selection import tra
X_train, X_test, y_train, y_test = trai

# Feature Scaling
from sklearn.preprocessing import Stand
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)

from sklearn.ensemble import RandomFore
classifier = RandomForestClassifier(n_e
classifier.fit(X_train, y_train)

#predict the test results
y_pred = classifier.predict(X_test)

#Making the confusion matrix
from sklearn.metrics import confusion_m
cm = confusion_matrix(y_test, y_pred)
print(cm)
from sklearn.metrics import confusion_m
cm = confusion_matrix(y_test, y_pred)
print(cm)
#Applying K-Fold cross validation
from sklearn.model_selection import cross_val_score
accuracies = cross_val_score(estimator = classifier, X = X_train, y = y_train, c
```

```
Python 3.7.0 (v3.7.0:[illegible], Jun 27 2018, [illegible]) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
============ RESTART: D:\204149\malwaredetection\Random Forest.py ============
[[1406   11]
 [  27  [illegible]]]
[[1406   11]
 [  27  [illegible]]]
[[illegible]]
[illegible]
[illegible]
>>>
```



```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Mon Aug 27 11:29:?? 2

@author: alex
"""

# K-Nearest Neighbors (K-NN)

# Importing the libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# Importing the dataset
dataset = pd.read_csv('data.csv'
X = dataset.drop(['Name', 'md5',
y = dataset['legitimate'].values

# Splitting the dataset into the
from sklearn.model_selection imp
X_train, X_test, y_train, y_test

# Feature Scaling
from sklearn.preprocessing impor
sc = StandardScaler()
X_train = sc.fit_transform(X_tra
X_test = sc.transform(X_test)

# Fitting K-NN to the Training s
from sklearn.neighbors import KN
classifier = KNeighborsClassifie
classifier.fit(X_train, y_train)

# Predicting the Test set result
y_pred = classifier.predict(X_te

# Making the Confusion Matrix
```

```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD6
4)] on win32
Type "
>>>
=======
[[1380
 [  32
The ac
```

This PC > New Volume (D:) > 234149 > malwaredetection
Sort
View
Name
data
File Edit Format Run Options Window
File Edit Shell Debug Options Window Help
Figure 1
0.026
0.028
0.030
0.032
0.034
0.036
Misclassification Error
0
10
20
30
40
50
Number of Neighbors K
Activate Windows
Go to Settings to activate Windows.
Ln: 7 Col: 0

7.CONCLUSIONANDFUTUREWORK

•This paper's objective is to introduce a machine learning solution to the malware problem. We require automatic solutions to find infected files due to the sudden increase of malware.

•In the initial stage of the project, corrupt and clean executables were utilised to create the data set. A Python script was used to extract the data required for the data set's creation. The data collection must be prepared for machine learning algorithms to be trained after being created.

•Decision trees, Random Forest, Naive Bayes, GradientBoost, and ADABoost are the algorithms that were utilised and were compared. It had a Random Forest method with an accuracy of 99.406012% after using the best accuracy techniques.

FUTURE SCOPE

- This can be made more accurate with adding more data set
- More algorithms with better performance can add on to accuracy
- It can hosted on web for real time analysis of exe files on the cloud

8. REFERENCES

- Malware Types and Classifications, Bert Rankin, 28.03.2018, published in LastLine, last accessed 12.09.2018.

- A Brief History of Malware - Its Evolution and Impact, Bert Rankin, 05.04.2018, published in LastLine, last accessed 12.09.2018.

- Detecting malware through static and dynamic techniques, Jeremy Scott, 14.09.2017, published in NTT Security, last accessed 12.09.2018.

- Hybrid Analysis and Control of Malware, Kevin A. Roundy and Barton P. Miller, International Workshop on Recent Advances in Intrusion Detection, pp. 317-338, 2010, Springer.

- Advanced Malware Detection - Signatures Vs. Behavior Analysis John Cloonan Director of Products, Lastline, 11.04.2017, published in Infosecurity Magazine, last accessed 12.09.2018.

- What is Machine Learning? Daniel Faggella, 12.08.2017, published in techemergence, last accessed 12.09.2018.

- Data mining, Margaret Rouse, Search SQL Server last accessed 12.09.2018, the article can be found here. • https://searchsqlserver.techtarget.com/definition/data-• • mining• • [8]

- Supervised and Unsupervised Machine Learning Algorithms, Jason Brownlee, 16.03.2016, published in Machine Learning Algorithms, last accessed 12.09.2018.

- Decision trees, scikit-learn.org last accessed 12.09.2018.

- RandomForestClassifier, scikit-learn.org last accessed 12.09.2018.

- GradientBoostingClassifier, scikit-learn.org last accessed 12.09.2018.

- Malware Researcher's Handbook, Resources Infosecinstitute, last accessed 129.2018.