JES
Journal of Engineering Sciences
Vol 13, Issue 02, Feb /2022
ISSN NO: 0377-9254

Learning to Classify Malicious Software using Malware Dataset

K.Padmasri[1], Dr. Sheo Kumar[2], Dr. Sheo kumar[3]
Student[1] Guide[2],Hod[3]
Department: Computer Science & Engineering
College Name: CMR Engineering College
Address: kandlakoya, Medchal Road, Hyderabad-501401

ABSTRACT

Malicious software is any application designed to cause damage or compromise the personal information of the user (also known as malware). Because of this, you should be on the lookout for any suspicious behavior from them. In any case, it's hardly worth the effort to speculate on a hacker's intentions. Instead, concentrate on questions to which you may be able to respond. It may be used in a variety of ways. This wide category includes everything meant to do damage What Can Malicious Software Do to A Computer? To begin with, the threat of malware is not restricted to the computer. Any gadget that can connect to the internet has the potential to become contaminated. If you get infected, you may have a variety of negative consequences.

Malware, for example, might enable a third party to take control of your computer or gadget. Installing applications, altering settings or passwords, or stealing intellectual property are all examples of this (among other things). The person in charge of the virus will have access to whatever you put on the computer.

INTRODUCTION

In many situations, malware is designed to benefit the attacker financially. Malware assaults have been used to lock individuals out of their computers in recent years. The attacker will first infect your computer using phishing or another social-engineering tactic in circumstances like these, which are known as "ransom ware assaults." They'll then encrypt the whole hard disc using the unauthorized access they've got. Normally, the hacker will issue a ransom demand, and the victim will be unable to recover access unless they comply with the hacker's demands.

Malware assaults (viruses, spam bots, it was found that at the beginning of 2007, there were over 350 incidents of mobile virus detections, suggesting a shift away from the widespread Internet and towards the more popular mobile networks. Malware spreads by transferring data over the internet, and here is why. Internet-connected devices and networks may now be infected with malware that was previously only accessible through a wired connection. Computer and network security are at jeopardy because of the fast growth and complexity of

Advantages of Proposed System

The suggested method has been tested on 16489 malicious and 8422 benign files. Using ensemble machine learning methods, we were able to identify malware with a 99.54 percent accuracy rate. Furthermore, it seeks to construct a high-accuracy behavior-based malware detection approach by repurposing runtime characteristics.

1.5 Software Requirements Specifications

1.4.1 We are developing our project in using a web-based application Using a browser interface, a user may access a remote server's copy of a Web application (Web app).

1.4.2 We are using technology of Python is a flexible programming language that may be used for a wide range of tasks. As a consequence, desktop and web applications may be developed using the programming language. Scientific and numerical applications may be built using Python as well, thanks to its flexibility. In terms of data analysis and visualisation, Python has a wide range of tools at its disposal.

1.4.3 We are using our web frame work by using High-level Python Web framework Django encourages speedy development and a logical, intuitive user interface. So you can focus on creating your app rather than re-inventing the wheel, it was designed by expert developers to alleviate many of the annoyances associated with Web development. It's free and open source. Astonishingly fast.

1.4.4 We are using **PostgreSQL** as our database, a free and open-source relational database management system (RDBMS), Postgre places a high value on adaptability and respect to technical standards. It is also known as Postgre. Its supports object oriented database.

1.4.5 Last we are using Microsoft Visual Code for IDE (Integrated development environment)

System Requirements

Hardware Requirements	
RAM	4 GB Minimum
Processor	i3 Minimum
Hard disk	500 GB
Software Requirements	
Technology	Python 3.6
Operating System	Windows Family

IDE	VS Code
Technology	Python, Django
Database Server	Postgre sql
Front Design Technology	HTML, CSS, JS

Literature Survey

[1] M. Alaeiyan, S. Parsa, M. Conti, Analysis and classification of context based malware behavior, Comp. Common. 136 (2019) 76–90 2019, doi: 10.1016/j.comcom.2019.01.003.

- An overview of the findings on malware behavior classification based on triggers is presented in this article, modeling, and the creation of behavioral signatures are all included in the elicitation process.
- With this new classification system, we can distinguish between evasive and provoked behaviors in trigger-based malware. Both of these acts are based on the creation of a certain environment.
- However, evasive behavior is motivated by self-defense, while triggered acts reveal virus-driven malfeasance.

[2] S.M. Bidoki, S. Jalili, A. Tajoddin, PbMMD: a novel policy based multi-process malware detection, Eng. Appl. Artif. Intell. 60 (August 2016) (2017) 57–70, doi: 10.1016/j.engappai.2016.12.008.

- To avoid detection, modern malware exploits a number of techniques. Behavior-based detection is the most prevalent method of detecting anomalies in the system effective method for detecting malware.
- This approach uses system call sequences to imitate malicious behavior. Multi-process malware is a new kind of virus that may elude detection depending on its behavior.
- Each of the several programmer that make up this virus performs a little part of the overall job, but none of them exhibit any dangerous behavior.

[4] A. Bushby and F. Cybersecurity, how deception may modify cyber security defences, Comp. Fraud Secur. Bull. 2019 (1) 12–14, doi:10.1016/S1361-3723(19)30008-9.

- Using deception technologies, it's becoming more popular as a practical post-breach active defense. in modern cyber security. However, like any new technology, there are certain misconceptions regarding it.

- There is a growing need for effective cyber defences that go beyond just identifying malicious actors in the middle of a sea of benign activity. In light of today's threats, an active defense system should be deployed to lure, detect and combat malware as well as other intruders travelling across the network.

System analysis

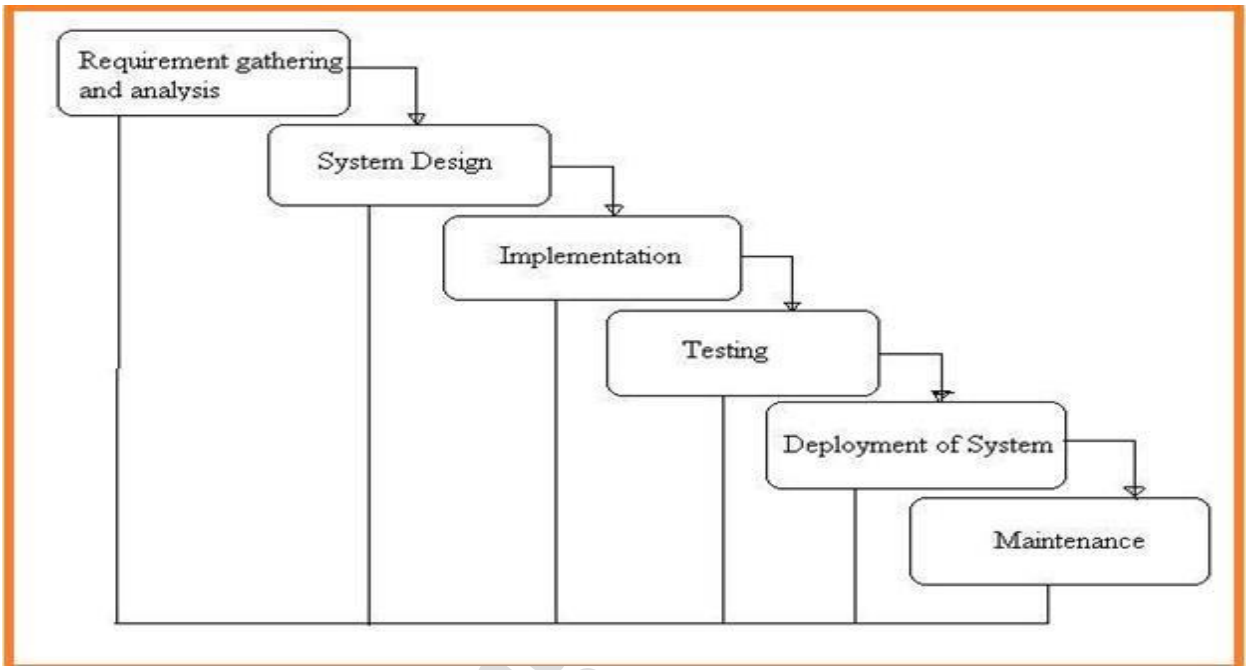


Fig: -1 Project SDLC

• Gathering and **analyzing** project requirements
• Designing an application system
• Putting it into **practice**
• Manual Testing of My Application
• Deployment of System Applications
• Keeping the Project Running

Gathering and Examining Needs

This is the first and most critical step of any endeavor, given that we are on a leave of absence from school. At this point in the project, we had already acquired a large number of IEEE Relegated papers, and so we decided to use the article "Individual Web Re-visitation by Setting and Substance Importance Input" as the basis for the project's requirements gathering

System Design

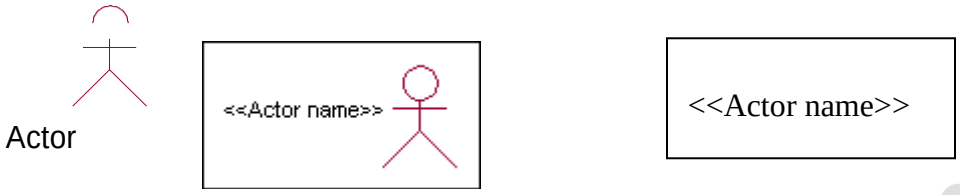
This document outlines all aspects of a system, from its basic needs to its operational environment to it's from the design of its files and databases to the architecture of its systems and subsystems to its user interfaces and other external interfaces.

Diagrams of World-Wide Use:

Identifying the actors:

Actor: An actor represents a user's position in the esteem of the framework When a character is used, the term "actor" is used to refer to them. may engage with them, but he or she has no say in the outcome.

Graphical representation:



To be an actor, one must be able to:

Interacts with the system in some way.

4.2 Data is entered into the system and information is retrieved from it.

4.3 Is not a part of the system and has no influence on the use cases

4.4 The following traits may be used to identify actors:

4.5 Who is the system's primary user?

4.6 Who is in charge of keeping the system up to date?

4.7 The system's utilization of external hardware.

4.8 Any computers and other devices that need to communicate with the system.

4.1 SEQUENCE DIAGRAMS

What happens initially, and then what happens after that, is shown as a time-based sequence in a sequence diagram. Class responsibilities and interfaces may be identified using sequence diagrams, which help show the roles of objects. In contrast to collaboration diagrams, sequence diagrams depict how items interact with one another over a longer period. Sequence diagrams show separate entities arranged vertically, temporally, and horizontally.

Object:

The state, behavior, and identity of an item are all intertwined. The structure and behaviour of similar things are defined by the shared class. A single instance of a class may be represented by a single item in a

diagram. An instance of a class is an object without a name.

The name of the item is highlighted, which gives it the appearance of a class icon: Object concurrency is determined by the concurrency of the class in which it is implemented.

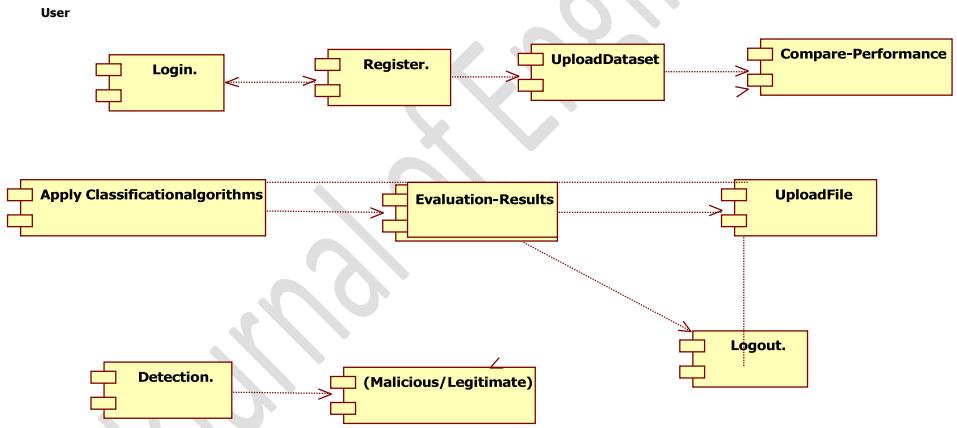
Message:

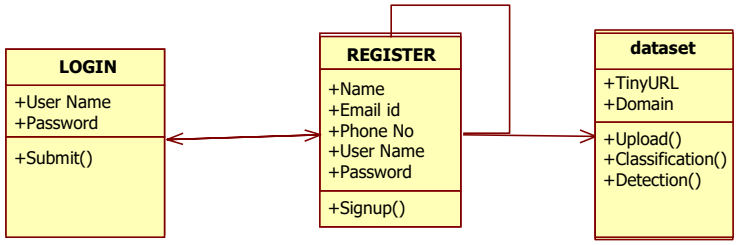
When a message is sent between two entities, it triggers a certain event. Sending and receiving a message is a two-way process that takes place between two points of control. A message's synchronization may be tweaked using the message specification. An instance of synchronisation occurs when a message is sent and then the sender waits for a response.

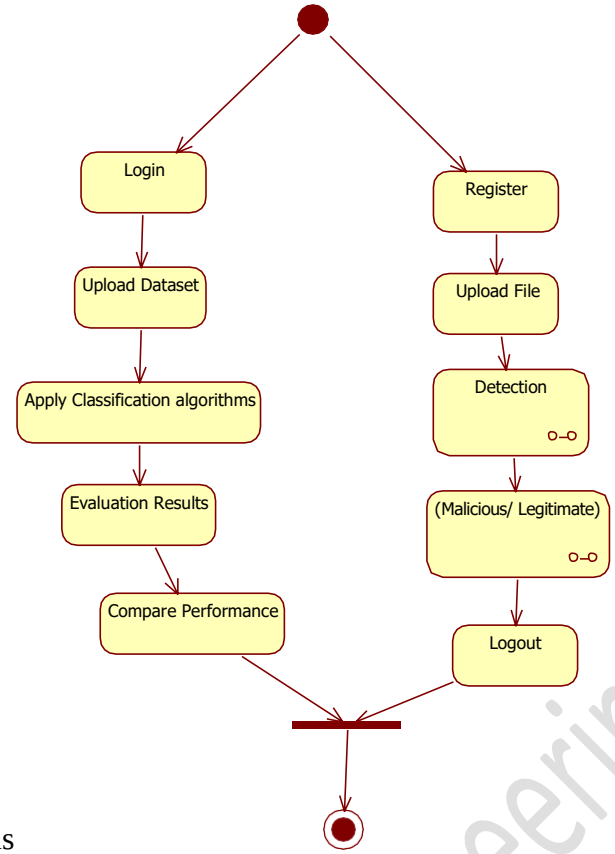
Link:

Objects, including class utilities, may only be connected if there is a relationship between their respective classes. It is possible for an object to send messages to another object if there is a connection between the instances of two classes. An object's link to a class is shown as a straight line in a cooperation diagram. If an item is related to itself, use the loop icon.

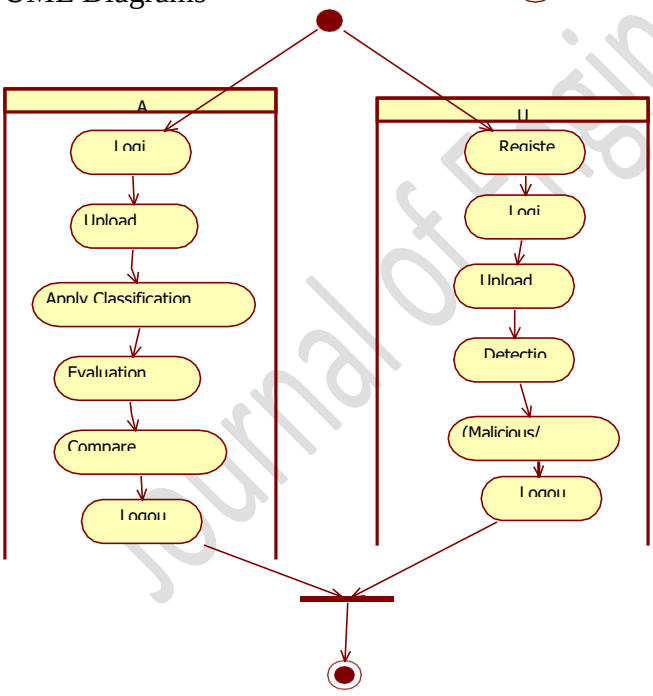
Sequence Diagram



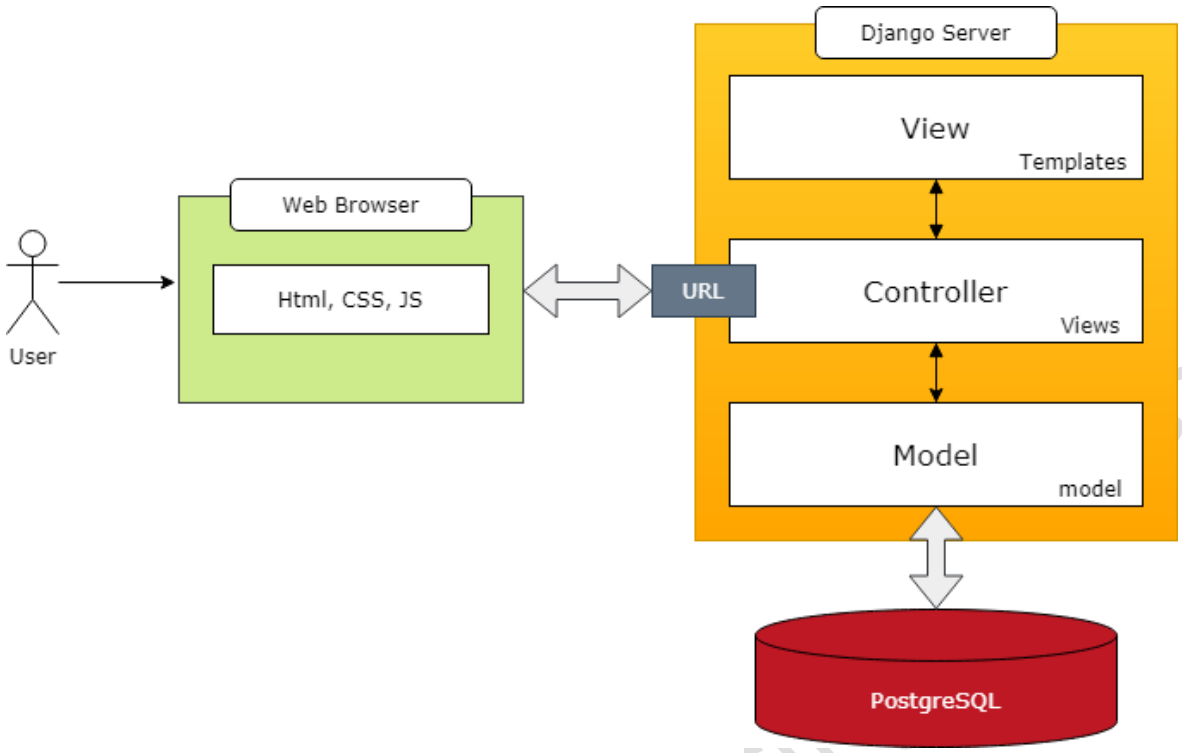




UML Diagrams



Journal of Engineering Sciences
Vol 13, Issue 02, Feb /2022
ISSN NO:0377-9254



Data Flow Diagrams

Diagrams showing how data moves through a system are known as data flow diagrams (DFDs).

A system's data flow diagrams provide a comprehensive picture. Notations such as Yourdon, Game, and Sarsen are used in the creation of the data flow diagrams. Each element of a DFD is given a descriptive name. Serial numbers are also used to identify the product. DFD'S is the result of a multi-stage process. Each lower-level graphic may be broken down into a more detailed DFD in the following level. "Context diagram" is a term used to describe the lop-level diagram here. To grasp the current system, one process bit is all that is required. The process shown in the context level diagram multiplies at the first level of the DFD.

It is the belief that information gained at one level of depth extends to a higher degree of detail at the next. The procedure is then done as many times as necessary to offer the analyst with as much information as they need to understand it.

For the modular architecture, Larry Constantine devised the DFD to communicate system requirements in an easy-to-read visual style.

By using a DFD, or "bubble chart," system designers may better communicate system requirements and identify key modifications that will be included into the system's programming. As a consequence, it serves as the starting point for the design at the most fundamental level possible. A DFD is made up of a series of data flows that link the bubbles in the system.

4.2 DFD Symbols

Distinct system components are shown by four distinct symbols in DFD diagram.

1. A square signifies the source (originator) or destination of system data (destination).
2. An arrow denotes data flow. It's the conduit via which information travels.
3. A process that converts incoming data flows into exiting data flows is represented by a circle or a bubble.
4. A data storage, data at rest, or a transitory data repository is an open rectangle.

4.3 Constructing a DFD

DFD's may be sketched using a few basic rules, such as:

Each process should have a name and a number assigned to it for easy referencing. The name of a process should accurately describe what it does. From top to bottom and left to right, the flow remains uninterrupted. In any case, data being sent back to the original source is rare. One way to indicate this is by drawing a long flow line back to the source. The source symbol may also be used as a destination symbol, if desired. Due of its frequency in the

Silent Feature of DFD's

1. Data flow diagram (DFD) displays data flow rather than control loops, thus decision-making aspects are not included in a DFD.
2. The DFD does not take into account the time invested in any action, regardless of how often it happens.
3. The sequence of events is not brought out on the DFD.

4.4 Data Flow

1) Symbols in a Data Flow may only flow in one direction at a time. It may indicate a read before an update in both directions between a process and a data store. Two different arrows are often used to depict the latter as they occur at discrete kinds.
2) The term "join" in DFD refers to the fact that data from two or more separate processes is transferred to a common destination.

3)A data flow cannot return to the same process from which it came. There must be at least one extra process to handle the data flow, create a new data flow, and return original data to the source process.

A When data is entered into a data store, it is updated automatically (delete or change).
Retrieving or utilizing data from a data repository is known as "data flow." All flows on a single arrow must travel together as one bundle for more than one data flow noun phrase to occur.

4.5 PostgreSQL

An RDBMS that ememphasizesdaptability and adherence to industry best practices is Relational database management system PostgreSQL is a popular open-source database management system (RDBMS). We can handle everything from a single PC to a data warehouse or a Web service with a large

number of concurrent users. it can handle a broad range of workloads. MacOS Server uses it by default and it is available on Linux, FreeBSD, OpenBSD and Windows. As an ACID database, PostgreSQL has all of the features you'd expect to see in an ACID database: updatable views, triggers and foreign keys, as well as stored procedures. The PostgreSQL Global Development Organization, a wide-ranging consortium of businesses and people, is responsible for the development of PostgreSQL.

4.5.1 Data types

Numerous native data kinds are at your disposal, including:

Numeric with arbitrary precision Boolean Characteristics (text, archer, char) Binary Date/Time (timestamp/time, date, interval, with/without time zone) Money

Enum

Bit strings are a kind of data that may be

Type of text search Composite

HStore is a PostgreSQL extension that allows you to use a key-value store.

Arrays are a sort of data structure (variable length and can be of any data type, including text and composite types) Total storage space of up to 1 GB

IPv4 and IPv6 addresses are geometric primitives.

CIDR blocks and MAC addresses (Classless Inter-Domain Routing) Support for XPath queries in XML

Identifier that is universally unique (UUID)

Since version 9.4, JavaScript Object Notation (JSON and JSONB, not to be confused with BSON) have been implemented in the JavaScript language (JSON). PostgreSQL's indexing infrastructures — GiST, GIN, and SP-GiST – allow users to create their own data types that are totally indexable. There are several examples of this, such as the PostGIS project for PostgreSQL's GIS data types. Another sort of data is called as "domain," which contains optional restrictions established by the domain's creator. In other words, any data that is entered into a column using the domain must comply to the domain's set of rules. Using range types, a collection of data may be represented and expressed in a variety of ways. Integer numbers 1 to 10 are an example of a discrete range, although they may also be continuous ranges (for example, all integer values 1 to 10). between the hours of 10:00 and 11:00 am, for example). The built-in range types include integers, big Examples of data include integers, decimal numbers, time stamps (with and without time zone), and dates. New kinds of ranges, such as IP address ranges based on the inet data type or float data type, may be created using custom range types. If you want to specify an inclusive or an exclusive range, you may use the characters [/] and (/). For example,

[4,9] indicates any values between 4 and 9, but excluding 9.) It is also possible to use range types with existing operators to verify whether an object is encircled or has the right of way.

Implementation

5.1 Introduction to Django

The following explanation is available at djangoproject.com if you use your browser or, depending on whatever decade you're reading this destined to be eternal literary masterpiece, with your cell phone, electronic notebook, shoe, or any other Internet-superseding technology.

As a high level Python Web framework, Allows you to convert URLs into code that responds to those requests. That is to say, you may define for each URL which code should be performed. You may direct the framework to "execute code that displays the profile for a user with that username for URLs that look like /users/joe/."

- Display, validation, and re-display of HTML forms are made easier by using this tool. A Web framework should make it easy to display HTML forms and handle the arduous coding of form display and redisplay, since they are the most popular methods for Web users to contribute data (with errors highlighted).
- Users' input is transformed into data structures that are easy to manipulate. It is possible, for example, that the framework will convert HTML form submissions into data types specific to the programming language itself.
- You may change the look and feel of your site without affecting the information, and vice versa, thanks to a template system that separates the two.
- Although it isn't necessary, it can readily communicate with storage layers such as databases.
- For example, if you were coding against HTTP, you would not be able to work efficiently at a higher abstraction level. But you can still go "down" a level of abstraction if you need to.
- As a result, stains like ".aspx" or “. php" URLs aren't left on your application, and you don't notice them.

It's never too late to learn something new, and because the Django Web framework for Python is free, there's no reason not to! Attending classes does not need paying for expensive courses or commuting from one part of town to another. Downloading and opening the PDF file is all you need to do to get started. Other courses in the Web programming subcategory include this one.

Thanks to others (like you?), who are willing to share their experience, you don't have to spend a lot of money to find out how well selected we are. Django, a Python web framework created by its creator, is now

open source. However, there are also a number of extra classes that may be found for free!

A computer PDF may save you both time and money when you're studying.

Django is a web framework for Python that is called Django. To get in touch with us if you have any problems, please use the form below.

It also has courses on HTML, CSS, Javascript, PHP, Asp, J2EE and a lot of other things about IT that you can read in PDF format.

Take a look at our website's programming guides for further information. You won't have a problem getting pleasure!

What sets Computer PDF out from the others is that it has the most current information and the best tutorials on your favorite topics, such as Django Web framework for Python!

Other tutorials on the Django Web framework for Python may be found at this link. You're going to see it! We promise to help you to the best of our abilities!

5.2 ALGORITHM USED

5.2.1 NEAREST NEIGHBOR ALGORITHM (I. K) (KNN):

As an algorithm for supervised learning, KNN is one of the slowest out there. Data training and testing on new instances are two separate processes. The K Nearest Neighbour algorithm's basic concept is that each data point, referred to as a "neighbour," is given a weight. For each of the K Nearest data points in the training dataset, three different kinds of distances must be measured using KNN. The majority of votes is now used to classify items. For each of the K Nearest data points in the training dataset, there are three kinds of distances that need to be measured. The majority of votes is now used to classify items. Each of the training datasets in KNN has three sorts of distances that need to be measured. There is a formula that may be used to determine the distance between Euclidian, Manhattan and Minkowski, with Euclidian being the most significant.

Defining the KNN algorithm begins with the following:

If you want to know how many closest neighbors you have, you need to know how many samples you need.

Create a super class for each sample class.

Calculate the Euclidian distance for each training sample.

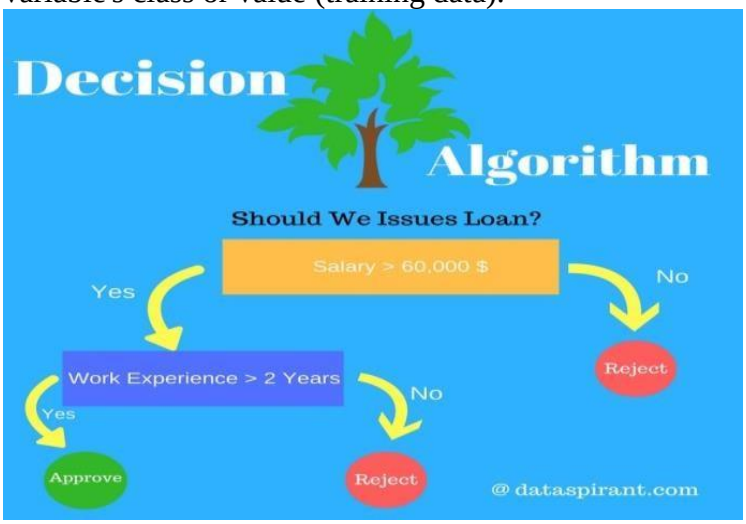
K- Nearest Neighbor

Decide on the sample's classification depending on which classes make up the vast majority in the immediate vicinity.

5.2.2 Decision-making Tree:

In the supervised learning algorithm class, Decision Trees are a subclass. Like other supervised learning algorithms, the decision tree technique may be used to tackle regression and classification problems. It is

possible to build a training model that employs the decision rules gained from prior research to predict a variable's class or value (training data).



Coding

6.1 Database connection

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.postgresql',
        'NAME': 'malware',
        'USER':'postgres',
        'PASSWORD':'sajid',
        'HOST':'localhost',
        'PORT':'5433',

    }
}
```

6.2 Database tables

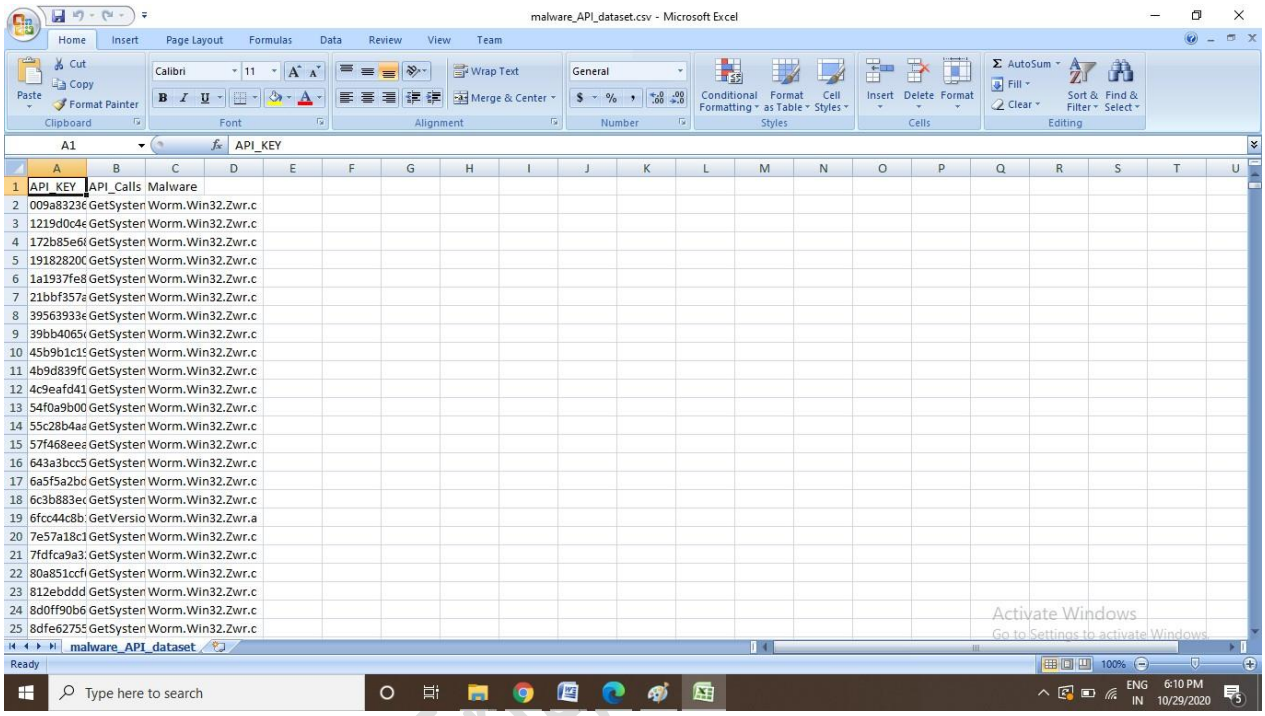
```
from django.db import models

# Create your models here.
class dataset(models.Model):
    API_KEY=models.CharField(max_length=10000);
    API_Calls=models.CharField(max_length=10000);
    Malware=models.CharField(max_length=1000);
class accuracysc(models.Model):
    algo=models.CharField(max_length=100);
    accuracyv=models.FloatField(max_length=1000)

class user(models.Model):
    email=models.CharField(max_length=100);
    pwd=models.CharField(max_length=100);
    name=models.CharField(max_length=100);
    phone=models.CharField(max_length=100);
    addr=models.CharField(max_length=100);
```

Datasets

malware_API_dataset



Dataset Attributes

API_KEY

API_Calls

Testing

Testing Methods

A) Software Testing

During the design and development phase, a software programmer is evaluated and confirmed to meet the technical specifications given. An further benefit is that it may be used to track down and fix any software flaws. It ensures the high quality of the programmer. Various types of software testing methods exist. These include manual testing, unit testing, black box testing, performance tests, stress tests, regression tests, and white box testing, among others. The following sections address some of the

most important forms of performance and load testing for an Android application.

Result of My Application on UC browser

Result of my Project in chrome

Journal of Engineering Sciences
Vol 13, Issue 02, Feb /2022
ISSN NO: 0377-9254

Result of my Project in Opera

ScreenShots of OutPut screens

Welcome screen

Home Screen

Journal of Engineering Sciences
Vol 13, Issue 02, Feb /2022
ISSN NO: 0377-9254

Admin Login

Admin

CONCLUSION

In this paper, a behavior-based method to malware detection is presented. In order to develop this method, we created a dynamic analysis environment and put malware samples through machine learning algorithms. There are a variety of ways in which software attacks may harm or destroy equipment, and they can even take over whole systems and steal or change data. When it comes to creating a dynamic analysis-based malware detection system, we're particularly interested in leveraging machine learning. Multiple machine learning approaches, such as K-Nearest Neighbors (KNN), Random Forest, AdaBoost Algorithm, Nave Bayees, and Decision Tree algorithms, were used to identify and compare the malware API calls dataset. Our Decision Tree algorithms have the greatest accuracy among them. Decision Trees for dynamic malware analysis prediction and data uploading were constructed by our team.

BIBLIOGRAPHY

[1] M. Alaeiyan, S. Parsa, M. Conti, Analysis and classification of context based malware behavior, Comp. Common. 136 (2019) 76–90 2019, doi: 10.1016/j.comcom.2019.01.003.

[2] S.M. Bidoki, S. Jalili, A. Tajoddin, PbMMD: a novel policy based multi-process malware detection, Eng. Appl. Artif. Intell. 60 (August 2016) (2017) 57–70, doi: 10.1016/j.engappai.2016.12.008.

[3] J. Brownlee, Master machine learning algorithms. discover how they work and implement them from scratch, Mach. Learn. Mast. Python 163 (2016) Retrieved from http://machinelearningmastery.com .

[4] A. Bushby, F. Cybersecurity, how deception can change cyber security defences, Comp. Fraud Secur. Bull. 2019 (1) (2019) 12–14, doi:10.1016/S1361-3723(19)30008-9.

[5] I.K. Cho, T.G. Kim, Y.J. Shim, M. Ryu, E.G. Im, (2016). Malware analysis and classification using sequence alignments, 8587(June). 10.1080/10798587.2015.1118916 [6] Y. Ding, X. Xia, S. Chen, Y. Li, a malware detection method based on family behavior graph, Comp. Secur. 73 (2018) 73–86, doi: 10.1016/j.cose.2017.10.007.

[7] M. Dredze, K. Crammer, F. Pereira, Confidence-weighted linear classification, in: International Conference on Machine Learning (ICML), 2008, pp. 264–271.

[8] S. Dua, X. Du, Data Mining and Machine Learning in Cybersecurity n.d., CRC Press Taylor and Francis, 2011.