DATA-DRIVEN DECISION SUPPORT FOR OPTIMIZING CYBER FORENSIC INVESTIGATIONS

A major project report submitted in partial fulfilment of the requirements for the award of degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

HEMANTH REDDY GALIGUTTA	**221810302021**
DUVVURI LAXMI PRANATHI	**221810302018**
CHINTAKINDI PRUTHVIRAJ	**221810302013**
SOMISETTY SAI CHARAN	**221810302056**

Under the guidance of

MR. RAJ MOHAMMED

Assistant Professor

Department of Computer Science and Engineering

GITAM School of Technology

GITAM Deemed to be University

Hyderabad Campus -502329

October - 2021

GANDHI INSTITUTE OF TECHNOLOGY AND MANAGEMENT (GITAM)

(Declared as Deemed-to-be-University u/s 3 of UGC Act 1956)

HYDERABAD CAMPUS

DECLARATION

We hereby declare that the mini project report entitled **"DATA-DRIVEN DECISION SUPPORT FOR OPTIMIZING CYBER FORENSIC INVESTIGATIONS"** is an original work done in the Department of Computer Science and Engineering, GITAM School of Technology, GITAM (Deemed to be University) submitted in partial fulfilment of the requirements for the award of the degree of "Bachelor of Technology" in Computer Science and Engineering. The work had not been submitted to any other college or university for the award of any degree or diploma.

Registration No(s)	Name(s)
221810302021	**HEMANTH REDDY GALIGUTTA**
221810302018	**DUVVURI LAXMI PRANATHI**
221810302013	**CHINTAKINDI PRUTHVIRAJ**
221810302056	**SOMISETTY SAI CHARAN**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GITAM SCHOOL OF TECHNOLOGY

GITAM

(DEEMED TO BE UNIVERSITY)

HYDERABAD CAMPUS

CERTIFICATE

This is to certify that the project report entitled **"DATA-DRIVEN DECISION SUPPORT FOR OPTIMIZING CYBER FORENSIC INVESTIGATIONS"** is a bonafide record of work carried out by **HEMANTH REDDY GALIGUTTA (221810302021), DUVVURI LAXMI PRANATHI (221810302018), CHINTAKINDI PRUTHVIRAJ (221810302013), SOMISETTY SAI CHARAN (221810302056)** submitted in partial fulfillment of the requirement for the award of the degree of Bachelors of Technology in Computer Science and Engineering.

Project Guide	**Head of the Department**
Mr. RAJ MOHAMMED	**S. PHANI KUMAR**
Assistant Professor of CSE	**Professor**

ACKNOWLEDGEMENT

Our project would not have been successful without the help of several people. we would like to thank the personalities who were part of our project in numerous ways, those who gave us outstanding support from the birth of the project.

We are extremely thankful to our honorable Pro-Vice Chancellor, **Prof. N.Siva Prasad** for providing necessary infrastructure and resources for the accomplishment of our project.

We are highly indebted to **Prof. N. Seetharamaiah**, Principal, School of Technology, for his support during the tenure of the project.

We are very much obliged to our beloved **Prof.S.Phani Kumar**, Head of the Department of Computer Science & Engineering for providing the opportunity to undertake this project and encouragement in completion of this project.

We hereby wish to express our deep sense of gratitude to **Mr. Raj Mohammed, Assistant Professor**, Department of Computer Science and Engineering, School of Technology for the esteemed guidance, moral support and invaluable advice provided by her for the success of the project.

We are also thankful to all the staff members of Computer Science and Engineering department who have cooperated in making our project a success. We would like to thank all our parents and friends who extended their help, encouragement and moral support either directly or indirectly in our project work.

Table of Contents

CHAPTER 1 ..... **14**
1.INTRODUCTION 14
1.1 Importance of computer forensics 14
1.2 Types of computer forensics 15
1.3 TECHNOLOGIES USED 15
1.3.1 HTML 15
1.3.2 CSS 16
1.3.3 JSP 16
1.3.4 MYSQL 17
1.4 PROJECT GOALS 17
1.5 MOTIVATION 18
1.6 ADVANTAGES 19
1.7 LIMITATIONS 19
1.8 Objectives 19
1.9 OUTCOMES 20
1.10 APPLICATIONS 21
CHAPTER 2 ..... **22**
2.LITERATURE SURVEY 22
CHAPTER 3 ..... **27**
3.PROBLEM ANALYSIS 27
3.1 PROBLEM DEFINATION 27
3.2 PROBLEM IDENTIFICATION 27
3.3 EXISTING SYSTEM 28
3.4 PROPOSED SYSTEM 28
3.5 REQUIREMENTS ENGINEERING 28
3.5.1 HARDWARE REQUIREMENTS 28
3.5.2 SOFTWARE REQUIREMENTS 29
3.6 FEATURES OF JAVA 29
3.6.1 THE JAVA FRAMEWORK 29
3.6.2 OBJECTIVES OF JAVA 30
3.6.3 OBJECT ORIENTED 30
3.7 COMPONENTS OF JAVA FRAMEWORK 31
3.7.1 JAVASERVER PAGES 31
3.7.2 SERVLETS 32

CHAPTER 4 ..... **33**
4.DESIGN 33
4.1 GENERAL 33
4.2 USE CASE 33
4.3 State Diagram 34
4.4 Class Diagram 35
4.5 Sequence Diagram 35
4.6 E-R Diagram 36
4.7 System Architecture 37
4.8 ARCHITECTURE Diagram 38
4.9 PROBLEM STATEMENT 39
CHAPTER 5 ..... **40**
5.IMPLEMENTATION 40
5.1 User Interface Design 40
5.1.1 Employee login 40
5.1.2 Add Report 40
5.1.3 View Report 41
5.1.4 Admin Login 42
5.1.5 Admin View 43
5.1.6 Admin Response 43
5.2 ALGORITHM 44
5.2.1 Working of Naïve Bayes' Classifier 44
5.2.2 Advantages of Naïve Bayes Classifier 45
5.2.3 Disadvantages of Naïve Bayes Classifier 45
5.2.4 Applications of Naïve Bayes Classifier 45
5.3 ALGORITHM IMPLIMENTATION 45
CHAPTER 6 ..... **46**
6.TESTING AND TRAINING 46
6.1 DEVELOPING METHODOLOGIES 46
6.2 TYPES OF TESTS 46
6.2.1 UNIT TESTING 46
6.2.2 FUNCTIONAL TEST 47
6.2.3 SYSTEM TEST 47
6.2.4 PERFORMANCE TEST 47
6.2.5 INTEGRATION TESTING 47
6.2.6 ACCEPTANCE TESTING 48
6.2.7 BUILD THE TEST PLAN 48

6.3 Performed testcases:.....48
6.3.1 EMPLOYEE DATA: 48
6.3.2 EMPLOYEE DATA VIEW 49
CHAPTER 7.....50
7.RESULT ANALYSIS.....50
7.1 ANALYSIS.....50
7.1.1 LIKELIHOOD VIEW 53
7.1.2 ALGORITHM SOLUTION 56
CHAPTER 8.....60
8.CONCLUSION & FUTURE SCOPE.....60
REFERENCES 61
REFERENCE LINKS 62

ABSTRACT

Cyber security threat is offensive action that targets computer networks, personal or professional devices by using various methods to alter, steal or destroy data. Cyber forensic explains how a policy became violated and who was responsible for it. Developing policies and procedures that establish the parameters for operation and function is an important phase of creating a computer forensics unit. An effective way to begin this task is to develop a mission statement that incorporates the core functions of the unit, whether those functions include high-technology crime investigations, evidence collection, or forensic analysis. However, Cyber forensic contains steps to investigate or collect the data., The analysis phase will lead to making the decision. It is defined as the processes and tools used in investigations and gathering evidence. Some of the instructions will be provided as default, such as category-wise. By analyzing the investigation report, the process will be optimized to reduce the investigation process. Staff will give the report daily basis; an expert will check and analyze the report to provide the instructions to the staff.

Departments should create policies and procedures for the establishment and/or operation of a computer forensics unit. Cyber forensic will help to identify the attack and give the reports. From the report, the analysis process provides the decision-making report. This may happen on a daily or weekly, or monthly basis based on the investigation reports. It will provide the instructions or any other kind of format to the staff to minimize their investigation process.

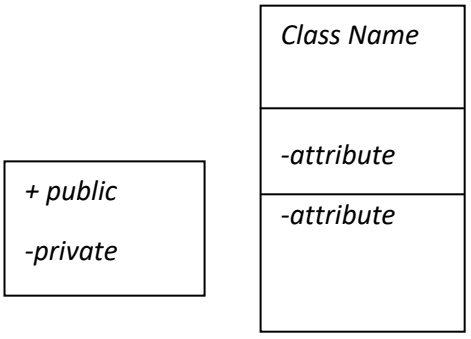
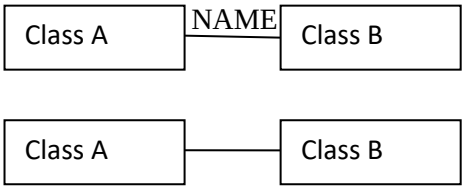
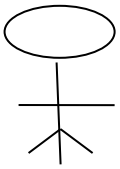
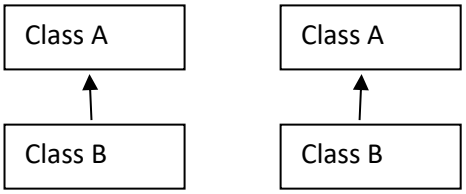
List of Figures

FIGURE NO.	NAME	PAGE NO.
Fig-4.1	Use case Diagram	33
Fig-4.2	State Diagram	34
Fig-4.3	Class Diagram	35
Fig-4.4	Sequence Diagram	36
Fig-4.5	E-R Diagram	36
Fig-4.6	System Architecture	37
Fig-4.7	Architecture Diagram	38
Fig-5.1	Employee login	40
Fig-5.2	Add report	41
Fig-5.3	View report	42
Fig-5.4	Admin login	42
Fig-5.5	Admin view	43
Fig-5.6	Admin response	44
Fig-6.1	Employee Data	48
Fig-6.2	Employee Data view	49
Fig-7.1	Likelihood view	55
Fig-7.2	Algorithm solution	59

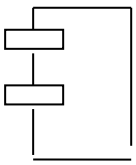
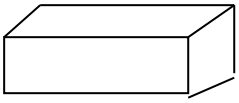
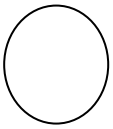
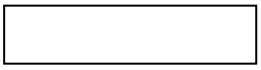
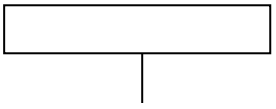
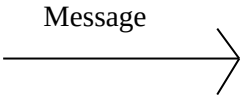
LIST OF ABBREVATION

S.NO	ABBREVATION	EXPANSION
1.	DB	Database
2.	JVM	Java Virtual Machine
3.	JSP	Java Server Page
4.	CB	Collective Behavior
5.	SD	Social Dimension
6.	JRE	Java Runtime Environment
7.	SSD	Sparse Social Dimension
8.	LGP	Line Graph Partition

LIST OF SYSMBOLS

S.NO	NOTATION NAME	NOTATION	DESCRIPTION
1.	Class		Represents a collection of similar entities grouped together.
2.	Association		Associations represents static relationships between classes. Roles represents the way the two classes see each other.
3.	Actor		It aggregates several classes into a single classes.
5.	Aggregation		Interaction between the system and external environment

6.	Relation (uses)	uses	Used for additional process communication.
7.	Relation (extends)	extends	Extends relationship is used when one use case is similar to another use case but does a bit more.
8.	Communication		Communication between various use cases.
9.	State	State	State of the process.
10.	Initial State		Initial state of the object
11.	Final state		Final state of the object
12.	Control flow		Represents various control flow between the states.
13.	Decision box		Represents decision making process from a constraint
14	Usecase	Uses case	Interact ion between the system and external environment.

15.	Component		Represents physical modules which is a collection of components.
16.	Node		Represents physical modules which are a collection of components.
17.	Data Process/State		A circle in DFD represents a state or process which has been triggered due to some event or acion.
18.	External entity		Represents external entities such as keyboard,sensors,etc.
19.	Transition		Represents communication that occurs between processes.
20.	Object Lifeline		Represents the vertical dimensions that the object communications.
21.	Message		Represents the message exchanged.

CHAPTER 1

1.INTRODUCTION

Computer forensics is the application of investigation and analysis techniques to gather and preserve evidence from a particular computing device in a way that is suitable for presentation in a court of law. The goal of computer forensics is to perform a structured investigation and maintain a document of evidence to find out exactly what happened on a computing device and who was responsible for it. The forensic process is also used as part of data recovery processes to gather data from a crashed server, failed drive, reformatted operating system (OS), or other situations where a system has unexpectedly stopped working. We have proposed an analysis of the forensics report using some software techniques.

Computer forensics which is sometimes referred to as computer forensic science, essentially is data recovery with legal compliance guidelines to make the information admissible in legal proceedings. The terms digital forensics and cyber forensics are often used as synonyms for computer forensics.

Digital forensics starts with the collection of information in a way that maintains its integrity. Investigators then analyze the data or system to determine if it was changed, how it was changed, and who made the changes. The use of computer forensics isn't always tied to a crime. The forensic process is also used as part of data recovery processes to gather data from a crashed server, failed drive, reformatted operating system (OS), or other situations where a system has unexpectedly stopped working.

1.1 IMPORTANCE OF COMPUTER FORENSICS

In the civil and criminal justice system, computer forensics helps ensure the integrity of digital evidence presented in court cases. As computers and other data-collecting devices are used more frequently in every aspect of life, digital evidence and the forensic process used to collect, preserve and investigate it has become more important in solving crimes and other legal issues.

The average person never sees much of the information modern devices collect. For instance, the computers in cars continually collect information on when the driver brakes, shifts, and changes speed without the driver being aware. However, this information can prove

critical in solving a legal matter or a crime, and computer forensics often plays a role in identifying and preserving that information.Digital evidence isn't just useful in solving digital-world crimes, such as data theft, network breaches, and illicit online transactions. It's also used to solve physical-world crimes, such as burglary, assault, hit-and-run accidents, and murder.

Businesses often use a multi layered data management, data governance, and network security strategy to keep proprietary information secure. Having data that's well managed and safe can help streamline the forensic process should that data ever come under investigation.

1.2 TYPES OF COMPUTER FORENSICS

There are various types of computer forensic examinations. Each deals with a specific aspect of information technology. Some of the main types include the following:

- Database forensics. The examination of information contained in databases, both data, and related metadata.
- Email forensics. The recovery and analysis of emails and other information contained in email platforms, such as schedules and contacts.
- Malware forensics. They are sifting through code to identify possible malicious programs and analyzing their payload. Such programs may include Trojan horses, ransomware, or various viruses. Memory forensics. Collecting information stored in a computer's random access memory (RAM) and cache.
- Mobile forensics. The examination of mobile devices to retrieve and analyze the information they contain, including contacts, incoming and outgoing text messages, pictures and video files.

1.3 TECHNOLOGIES USED

1.3.1 HTML

- HyperText Markup Language (HTML) is a simple markup system used to create hypertext documents that are portable from one platform to another. HTML documents are SGML documents with generic semantics that are appropriate for representing information from a wide range of applications.

- HyperText Markup Language (HTML) is the set of markup symbols or codes inserted into a file intended for display on the Internet. The markup tells web browsers how to display a web page's words and images.
- HTML is a file extension used interchangeably with HTM. ... The HTML tags can be used to define headings, paragraphs, lists, links, quotes, and interactive forms. It can also be used to embed Javascript and CSS (cascading style sheets).
- HTML tags are like keywords that define that how a web browser will format and display the content. With the help of tags, a web browser can distinguish between HTML content and simple content. HTML tags contain three main parts: an opening tag, content, and closing tag. ... Every tag in HTML performs different tasks.

1.3.2 CSS

CSS stands for Cascading Style Sheets. It is the **l**anguage for describing the presentation of Web pages, including colours, layout, and fonts, thus making our web pages presentable to the users. CSS is designed to make style sheets for the web

- Inline CSS.
- Internal or Embedded CSS.
- External CSS.

1.3.3 JSP

- JavaServer Pages (JSP) is a Web page development technology that supports dynamic content. This allows programmers to use specific JSP tags to insert Java code into HTML pages.
- A part of JavaServer Pages is a type of Java servlet designed to perform the function of a Java web application user interface. JavaServer Pages (JSP) is a technology for developing Webpages that supports dynamic content.
- This helps developers insert java code in HTML pages by making use of special JSP tags, most of which start with <% and end with %>. A JSP file is a server-generated web page. It is similar to an. ASP or . ... Since the Java code is parsed on the webserver, the end-user never sees the JSP code, but only the HTML generated by the Java code on the page.
- JSP pages can be edited using a web development program or basic text editor.

1.3.4 MYSQL

- MySQL is an open-source relational database management system (RDBMS). It is the most popular database system used with Backend. MySQL is developed, distributed, and supported by Oracle Corporation. The data in a MySQL database are stored in tables which consists of columns and rows.
- MySQL is a relational database management system based on SQL Structured
- Query Language. The application is used for a wide range of purposes, including data warehousing, e-commerce, and logging applications.
- The most common use for mySQL however, is for the purpose of a web database.

1.4 PROJECT GOALS

Forensic investigators typically follow standard procedures, which vary depending on the context of the forensic investigation, the device being investigated, or the information investigators are looking for. In general, these procedures include the following three steps:

1. **Data collection.** Electronically stored information must be collected in a way that maintains its integrity. This often involves physically isolating the device under investigation to ensure it cannot be accidentally contaminated or tampered with. Examiners make a digital copy, also called a forensic image, of the device's storage media, and then they lock the original device in a safe or another secure facility to maintain its pristine condition. The investigation is conducted on the digital copy. In other cases, the publicly available information may be used for forensic purposes, such as Facebook posts or public Venmo charges for purchasing illegal products or services displayed on the Vice website.
2. **Analysis.** Investigators analyze digital copies of storage media in a sterile environment to gather the information for a case. Various tools are used to assist in this process, including Basis Technology's Autopsy for hard drive investigations and the Wireshark network protocol analyzer. A mouse jiggler is useful when examining a computer to keep it from falling asleep and losing volatile memory data that is lost when the computer goes to sleep or loses power.

3. **Presentation.** The forensic investigators present their findings in a legal proceeding, where a judge or jury uses them to help determine the result of a lawsuit. In a data recovery situation, forensic investigators present what they were able to recover from a compromised system.

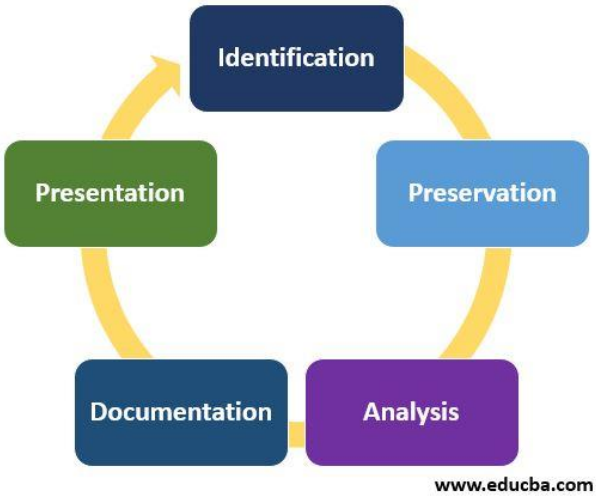


Fig- 1.1 Project goals and process

1.5 MOTIVATION

Due to the increasing dependency on digitalization and Internet-of-Things (IoT), various security incidents such as unauthorized access, malware attack, zero-day attack, data breach, denial of service (DoS), social engineering, or phishing, etc. have grown at an exponential rate in recent years.

According to Juniper Research, the number of records breached each year to nearly triple over the next five years. Thus, it's essential that organizations need to adopt and implement a strong cybersecurity approach to mitigate the loss. According to, the national security of a country depends on the business, government, and individual citizens having access to applications and tools which are highly secure and the capability of detecting and eliminating such cyber threats in a timely way. Therefore, to effectively identify various cyber incidents, either previously seen or unseen, and intelligently protect the relevant systems from such cyber-attacks is a key issue to be solved urgently.

1.6 ADVANTAGES

- To ensure the integrity of the computer system.
- To produce evidence in the court, which can lead to the punishment of the culprit.
- It helps the companies to capture important information if their computer systems or networks are compromised.
- Efficiently tracks down cybercriminals from anywhere in the world.
- Helps to protect the organization's money and valuable time.
- Allows to extract, process, and interpret the factual evidence, so it proves the cybercriminal action's in the court.

1.7 LIMITATIONS

- Digital evidence accepted into court. However, it is must be proved that there is no tampering
- Producing electronic records and storing them is an extremely costly affair
- Legal practitioners must have extensive computer knowledge
- Need to produce authentic and convincing evidence
- If the tool used for digital forensic is not according to specified standards, then in the court of law, the evidence can be disapproved by justice.
- Lack of technical knowledge by the investigating officer might not offer the desired result.

1.8 OBJECTIVES

- Identify, gather, and preserve the proof of a law-breaking.
- Track and prosecute the perpetrators in an exceedingly court of law.
- Interpret, document and gift the proof to be permissible throughout prosecution.
- Estimate the potential impact of a malicious activity on the victim and assess the intent of the offender.
- Find vulnerabilities and security loopholes that facilitate attackers.
- Understand the techniques and strategies utilized by attackers to avert prosecution, and overcome them.

- Recover deleted files, hidden files, and temporary information that would be used as proof.
- Perform incident response to forestall more loss of property, finances Associate in Nursing name throughout an attack.
- Have information concerning the laws of assorted regions and areas, as digital crimes are ubiquitous and remote in nature.
- Know the method of handling multiple platforms, information varieties and in operation systems.
- Understand the usage of correct tools for forensic investigations.

1.9 OUTCOMES

The Computer & Digital Forensics program is designed to provide you with the skills you need for career success. You'll learn the process of computer forensics, including topics within digital forensics and computer crimes. By graduation, you'll know the role digital evidence plays in criminal and civil investigations and incident response. The degree ensures that graduates will be able to:

- conduct digital investigations that conform to accepted professional standards and are based on the investigative process: identification, preservation, examination, analysis, and reporting;
- cite and adhere to the highest professional and ethical standards of conduct, including impartiality and the protection of personal privacy;
- identify and document potential security breaches of computer data that suggest violations of legal, ethical, moral, policy, and/or societal standards;
- apply a solid foundational grounding in computer networks, operating systems, file systems, hardware, and mobile devices to digital investigations and to the protection of computer network resources from unauthorized activity;
- work collaboratively with clients, management, and/or law enforcement to advance digital investigations or protect the security of digital resources;
- access and critically evaluate relevant technical and legal information and emerging industry trends; and
- communicate effectively the results of a computer, network, and/or data forensic analysis verbally, in writing, and in presentations to both technical and lay audiences.

1.10 APPLICATIONS

- Crime Detection- There are various malwares and malicious activities that happen over digital media and networks, such as phishing, spoofing, ransomware, etc.
- Crime Prevention- There are various cyber crimes that happen due to lack of security or existing unknown vulnerabilities, such as zero-day vulnerability. Hence, cyber forensics helps in finding out these vulnerabilities and avoiding such crimes to occur.
- Crime Analysis- This is the main application of digital forensics.
- Preservation- This process involves protecting the crime scene and the digital evidence or setup from further manipulation and photographing and video graphing the crime scene, for future reference. Also this process involves stopping any ongoing command that may be linked to the crime.
- Identification- This process involves identifying the digital media and devices that can serve as the potential evidence.
- Extraction- This process involves the imaging of the digital evidence, (to maintain the authenticity of the original evidence), for further analysis.
- Documentation- This involves maintaining the chain of custody and documenting all the evidence collected from the crime scene.
- Interpretation- This involves making of a report by the digital forensic expert about the analysis conducted on the digital evidence using various tools such as FTK (for imaging and mounting of evidences),Sleuth Kit and Autopsy (analyzes disk images and recover files from them) etc. and presenting it in the court of law. The conclusion is based on the evidence collected and reconstructing data fragments.

CHAPTER 2

2.LITERATURE SURVEY

[1] Author: A. Nieto

Year of publication: 2020

Title: Becoming judas: Correlating users and devices during a digital investigation (IEEE Transactions on Information Forensics and Security, vol. 15, pp. 3325–3334, 2020).

- One of the biggest challenges in IoT-forensics isthe analysis and correlation of heterogeneous digital evidence,to enable an effective understanding of complex scenarios.
- In order to illustrate this approach, the proposed methodology is implemented in the JSON Users and Devices analysis (JUDAS) tool, which is able to generate the context from JSON files, complete it, and show the whole context using dynamic graphs.
- The approach is validated using the files in an IoT-Forensic digital investigation where an important set of potential digital evidence extracted from Amazon's Alexa Cloud is analysed.
- This paper has proposed the JSON Users and Devicesanalysis (JUDAS) methodology to correlate users and devices,taking advantage of the JSON format widely used by manytools, either to save logs or to provide results of operations ondata during the analysis of digital evidence
- Last but not least, the context is delimited by the files ofa use case, or those belonging to a digital investigation. Forthe sake of simplicity a folder with all the potential digitalevidence is considered, but this can be much more complexin the case the digital information is distributed over severalsources.

[2] Author: B. Stojanovic, K. Hofer-Schmitz, and U. Kleb.

Year of publication: 2020

Title: APT datasets and attack modeling for automated detection methods: A review (Computers & Security, p. 101734, 2020).

- In order to test these methods properly, it is crucial to have a suitable dataset.
- This paper provides a review on datasets and their creation for use in APT detection in literature. A special focus is placed on feature engineering, including construction, selection and dimensionality reduction.
- Two use cases based on the underlying infrastructure are distinguished, large enterprise networks and Cyber Physical System, additionally including cloud computing systems, financial technology networks and Internet of Things networks. These datasets are usually based on an attack model.
- A description of different stages including approaches and goals of such attacks are given.
- The major achievement is the description and analysis of existing feature extraction methodologies and detailed overview of datasets used in APT detection related literature.
- This shows that the large enterprise network use case, has incorporated a much more frequent use of datasets with quite short periods of time. In the case of Cyber Physical System, a realistic dataset is publicly available

[3] Author: S. M. Milajerdi, R. Gjomemo, B. Eshete, R. Sekar, and V. Venkatakrishnan.

Year of publication: 2019.

Title: Holmes: real-time apt detection through correlation of suspicious information flows (IEEE symposium on security and privacy (sp). ieee, 2019, pp. 1137–1152).

- In this paper, we present HOLMES, a system that implements a new approach to the detection of Advanced and Persistent Threats (APTs). HOLMES is inspired by several case studies of real-world APTs that highlight some common goals of APT actors.
- In a nutshell, HOLMES aims to produce a detection signal that indicates the presence of a coordinated set of activities that are part of an APT campaign. One of the main challenges addressed by our approach involves developing a suite of techniques that make the detection signal robust and reliable
- At a high-level, the techniques we develop effectively leverage the correlation between suspicious information flows that arise during an attacker campaign.
- In addition to its detection capability, HOLMES is also able to generate a high-level graph that summarizes the attacker's actions in real-time. This graph can be used by an analyst for an effective cyber response.
- An evaluation of our approach against some real-world APTs indicates that HOLMES can detect APT campaigns with high precision and low false alarm rate.
- The compact high-level graphs produced by HOLMES effectively summarizes an ongoing attack campaign and can assist real-time cyber-response operations.

[4] Author: C. Yan, B. Li, Y. Vorobeychik, A. Laszka, D. Fabbri, and B. Malin,.

Year of publication: 2018.

Title: Get your workload in order: Game theoretic prioritization of database auditing (2018 IEEE 34th International Conference on Data Engineering (ICDE). IEEE, 2018, pp. 1304–1307).

- The quantity of personal data that is collected, stored, and subsequently processed continues to grow at a rapid pace.
- Given its potential sensitivity, ensuring privacy protections has become a necessary component of database management. To enhance protection, a number of mechanisms have been developed, such as audit logging and alert triggers.
- First, the volume of such alerts grows with the size of the database and is often substantially greater than the capabilities of resource-constrained organizations.
- Second, strategic attackers can attempt to disguise their actions or carefully choosing which records they touch, such as by limiting the number of database accesses they commit, thus potentially hiding illicit activity in plain sight.
- There are several limitations of our approach that we wish to highlight as opportunities for future investigations. First, there are limitations to the parameterization of the game. One not able aspect is that we assumed that the game has a zero-sum property.
- Specifically, we expected the interaction between the auditor and adversaries as fully rational. In reality, adversaries may be bounded in their rationality, and an important extension would be to generalize the model consider such behavior.

[5] Author: H. Studiawan, C. Payne, and F. Sohel.

Year of publication: 2017

Title: Graph clustering and anomaly detection of access control log for forensic purposes (Digital Investigation, vol. 21, pp. 76–87, 2017)

- Attacks on operating system access control have become a significant and increasingly common problem.
- This type of security threat is recorded in a forensic artifact such as an authentication log. Forensic investigators will generally examine the log to analyze such incidents.
- An anomaly is highly correlated to an attacker's attempts to compromise the system. In this paper, we propose a novel method to automatically detect an anomaly in the access control log of an operating system.
- The logs will be first preprocessed and then clustered using an improved MajorClust algorithm to get a better cluster.
- This technique provides parameter-free clustering so that it automatically can produce an analysis report for the forensic investigators.
- The clustering results will be checked for anomalies based on a score that considers some factors such as the total members in a cluster, the frequency of the events in the log file.

CHAPTER 3

3.PROBLEM ANALYSIS

3.1 PROBLEM DEFINATION

Cyber- crimes can be categorized as internal or external events. Typically, the largest threat to organizations has been employees and insiders that is why computer crime is often referred to as 'insider' crime. Internal crimes are committed by those with a substantial link to the intended victim. However, with advancement of remote networks, the threat from external source is increasing with a rapid pace. An external event is committed anonymously. Internal events can generally be contained within the attacked organization as it is easier to determine a motive and, therefore, simpler to identify the offender. However, when the person involved has used intimate knowledge of information technology infrastructure, obtaining digital evidence of the offense is quite difficult. An external event is hard to predict, yet can often be traced using evidence provided by the organization under attack. Typically, the offender has no motive and is not even connected with the organization, making it fairly straightforward to prove unlawful access to data or systems.

3.2 PROBLEM IDENTIFICATION

Cyber-crime occurs when information technology is used to commit or conceal an offense. Computer crimes include:

- Financial fraud Sabotage of data and/or networks;
- Theft of proprietary information;
- System penetration from the outside and denial of service;
- Unauthorized access by insiders and employee misuse of Internet access privileges
- Viruses, which are leading cause of unauthorized users gaining access to systems and networks through the Internet.

As "first wave" wars were fought for land and "second wave" wars were fought for control over productive capacity, the "third wave" wars are being fought for knowledge.

3.3 EXISTING SYSTEM

Cyber security tends to focus on how malicious actors use electronic assets to attack information, running the assets safely with security implementations of databases, networks, hardware, firewall. Though, it prevents individuals, organizations, financial institutions and universities from cyber-attack forensic investigation will help to identify and improve the security system, design and architecture. Generally, cyber forensic investigation happen after the incident. Sometime, it may delay the process to get the result for the investigations.

3.4 PROPOSED SYSTEM

Departments should create policies and procedures for the establishment and/or operation of a computer forensics unit. Cyber forensic will help to identify the attack and give the reports. From the report, analysis process provide the decision making report. This may happen daily or weekly or monthly basis based on the investigation reports. It will provide the instructions or any other kind of format to the staff to minimize their investigation process.

3.5 REQUIREMENTS ENGINEERING

These are the requirements for doing the project. Without using these tools and software's we can't do the project. So we have two requirements to do the project. They are

1. Hardware Requirements.
2. Software Requirements.

3.5.1 HARDWARE REQUIREMENTS

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It shows what the system does and not how it should be implemented.

Minimum Requirements:

PROCESSOR	:	PENTIUM IV 2.6 GHz, Intel Core 2 Duo.
RAM	:	4GB DD RAM
MONITOR	:	15" COLOR
HARD DISK	:	40 GB

3.5.2 SOFTWARE REQUIREMENTS

The software requirements document is the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the team's and tracking the team's progress throughout the development activity.

Front End	:	J2EE (JSP, SERVLETS) JAVASCRIPT
Back End	:	MY SQL 5.5
Operating System	:	Windows, Mac
IDE	:	Eclipse

3.6 FEATURES OF JAVA

3.6.1 THE JAVA FRAMEWORK

Java is a programming language originally developed by James Gosling at Sun Micro systems and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++ but has a simpler object model and fewer low-level facilities. Java applications are typically compiled to byte code that can run on any Java Virtual Machine (JVM) regardless of computer architecture. Java is general-purpose, concurrent, class-based, and object-oriented, and is specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere".

Java is considered by many as one of the most influential programming languages of the 20th century, and is widely used from application software to web applications. The java framework is a new platform independent that simplifies application development internet .Java technology's versatility, efficiency, platform portability, and security make it the ideal technology for network computing. From laptops to datacenters, game consoles to scientific supercomputers, cell phones to the Internet, Java is everywhere!

3.6.2 OBJECTIVES OF JAVA

Java has been tested, refined, extended, and proven by a dedicated community. And numbering more than 6.5 million developers, it's the largest and most active on the planet. With its versatility, efficiency, and portability, Java has become invaluable to developers by enabling them to:

- Write software on one platform and run it on virtually any other platform
- Create programs to run within a Web browser and Web services
- Develop server-side applications for online forums, stores, polls, HTML forms processing, and more
- Combine applications or services using the Java language to create highly customized applications or services
- Write powerful and efficient applications for mobile phones, remote processors, low-cost consumer products, and practically any other device with a digital heartbeat
- Today, many colleges and universities offer courses in programming for the Java platform. In addition, developers can also enhance their Java programming skills by reading Sun's java.sun.com Web site, subscribing to Java technology-focused newsletters, using the Java Tutorial and the New to Java Programming Center, and signing up for Web, virtual, or instructor-led courses.

3.6.3 OBJECT ORIENTED

To be an Object Oriented language, any language must follow at least the four characteristics.

1.Inheritance: It is the process of creating the new classes and using the behavior of the existing classes by extending them just to reuse the existing code and adding addition a features as needed.

2.Encapsulation: It is the mechanism of combining the information and providing the abstraction.

3.Polymorphism: As the name suggest one name multiple form, Polymorphism is the way of providing the different functionality by the functions having the same name based on the signatures of the methods.

4.Dynamic binding: Sometimes we don't have the knowledge of objects about their specific types while writing our code. It is the way of providing the maximum functionality to a program about the specific type at runtime.

3.7 COMPONENTS OF JAVA FRAMEWORK

3.7.1 JAVASERVER PAGES

Java Server Pages or JSP for short is Sun's solution for developing dynamic web sites. JSP provide excellent server side scripting support for creating database driven web applications. JSP enable the developers to directly insert java code into jsp file, this makes the development process very simple and its maintenance also becomes very easy.

JSP pages are efficient, it loads into the web servers memory on receiving the request very first time and the subsequent calls are served within a very short period of time.

In today's environment most web sites servers dynamic pages based on user request. Database is very convenient way to store the data of users and other things. JDBC provide excellent database connectivity in heterogeneous database environment. Using JSP and JDBC its very easy to develop database driven web application.

Java is known for its characteristic of "write once, run anywhere." JSP pages are plat Java Server Pages.

Java Server Pages (JSP) technology is the Java platform technology for delivering dynamic content to web clients in a portable, secure and well-defined way. The Java Server Pages specification extends the Java Servlet API to provide web application developers with a robust framework for creating dynamic web content on the server using HTML, and XML templates, and Java code, which is secure, fast, and independent of server platforms.

JSP has been built on top of the Servlet API and utilizes Servlet semantics. JSP has become the preferred request handler and response mechanism. Although JSP technology is going to be a powerful successor to basic Servlets, they have an evolutionary relationship and can be used in a cooperative and complementary manner.

Servlets are powerful and sometimes they are a bit cumbersome when it comes to generating complex HTML. Most servlets contain a little code that handles application logic and a lot more code that handles output formatting. This can make it difficult to separate and reuse portions of the code when a different output format is needed. For these reasons, web application developers turn towards JSP as their preferred servlet environment.

One of the main reasons why the Java Server Pages technology has evolved into what it is today and it is still evolving is the overwhelming technical need to simplify application design by separating dynamic content from static template display data. Another benefit of utilizing JSP is that it allows to more cleanly separate the roles of web application/HTML

designer from a software developer. The JSP technology is blessed with a number of exciting benefits, which are chronicled as follows:

1. The JSP technology is platform independent, in its dynamic web pages, its web servers, and its underlying server components. That is, JSP pages perform perfectly without any hassle on any platform, run on any web server, and web-enabled application server. The JSP pages can be accessed from any web server.

2. The JSP technology emphasizes the use of reusable components. These components can be combined or manipulated towards developing more purposeful components and page design. This definitely reduces development time apart from the At development time, JSPs are very different from Servlets, however, they are precompiled into Servlets at run time and executed by a JSP engine which is installed on a Web-enabled application server such as BEA WebLogic and IBM WebSphere.

3.7.2 SERVLETS

Earlier in client- server computing, each application had its own client program and it worked as a user interface and need to be installed on each user's personal computer. Most web applications use HTML/XHTML that are mostly supported by all the browsers and web pages are displayed to the client as static documents.

A web page can merely displays static content and it also lets the user navigate through the content, but a web application provides a more interactive experience.

Any computer running Servlets or JSP needs to have a container. A container is nothing but a piece of software responsible for loading, executing and unloading the Servlets and JSP. While servlets can be used to extend the functionality of any Java- enabled server.

They are mostly used to extend web servers, and are efficient replacement for CGI scripts. CGI was one of the earliest and most prominent server side dynamic content solutions, so before going forward it is very important to know the difference between CGI and the Servlets.

Java Servlet is a generic server extension that means a java class can be loaded dynamically to expand the functionality of a server. Servlets are used with web servers and run inside a Java Virtual Machine (JVM) on the server so these are safe and portable.

CHAPTER 4

4.DESIGN

4.1 GENERAL

Design Engineering deals with the various UML [Unified Modeling language] diagrams for the implementation of project. Design is a meaningful engineering representation of a thing that is to be built. Software design is a process through which the requirements are translated into representation of the software. Design is the place where quality is rendered in software engineering. Design is the means to accurately translate customer requirements into finished product.

4.2 USE CASE

The use case diagram is the main building block of object oriented modeling. It is used both for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code. For this in our component diagram first propose a data In this proposed method we are using Hash-Solomon Code Algorithm to encrypt the data. Then user2 want to access the file by the permission of user1 share the authenticated key

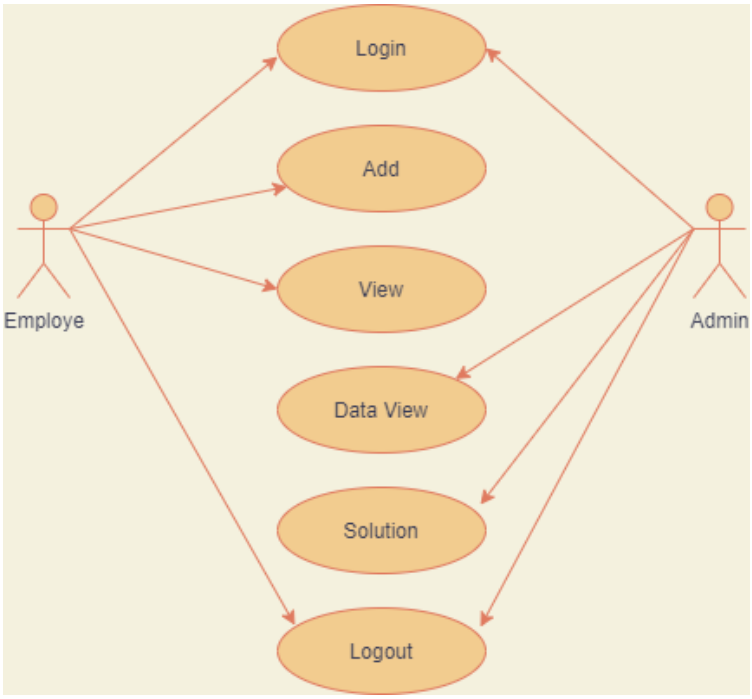


Fig- 4.1 Use case diagram

4.3 STATE DIAGRAM

State diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction. Many forms of state diagrams exist, which differ slightly and have different semantics. In our state diagram first propose For this in our component diagram first propose a data In this proposed method we are using Hash-Solomon Code Algorithm to encrypt the data. Then user2 want to access the file by the permission of user1 share the authenticated key.

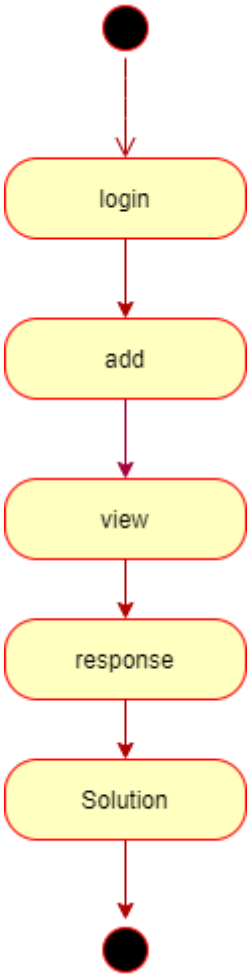


Fig- 4.2 State diagram

4.4 CLASS DIAGRAM

Class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, and the relationships between the classes. The classes in a class diagram represent both the main objects and or interactions in the application and the objects.

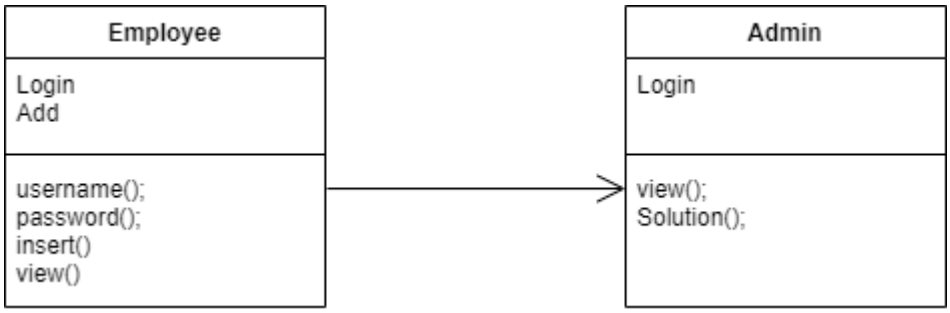


Fig- 4.3 Class diagram

4.5 SEQUENCE DIAGRAM

In our sequence diagram specifying processes operate with one another and in order. In our sequence diagram first propose a For this in our component diagram first propose a data In this proposed method we are using Hash-Solomon Code Algorithm to encrypt the data. Then user2 want to access the file by the permission of user1 share the authenticated key.

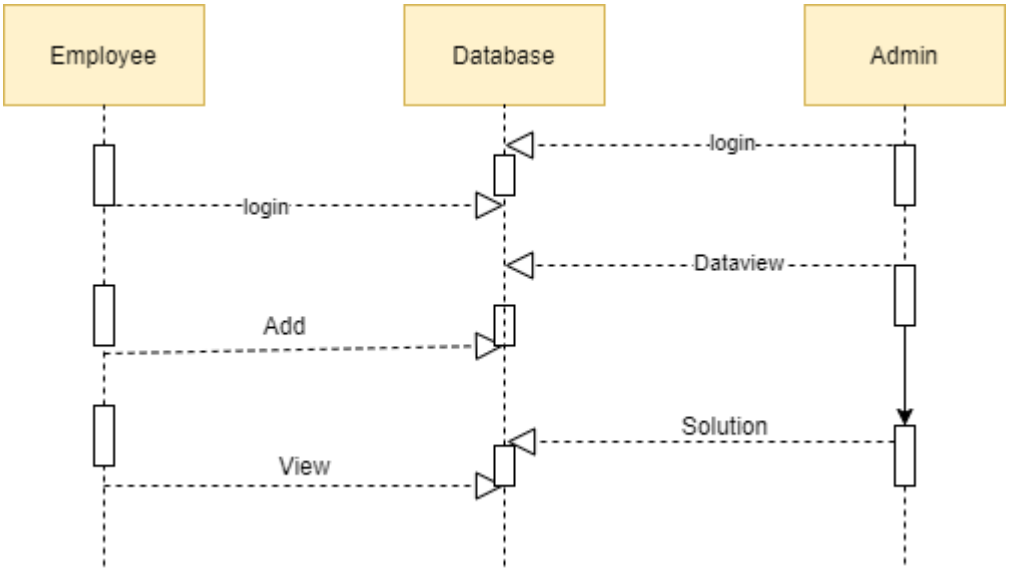


Fig- 4.4 Sequence diagram

4.6 E-R DIAGRAM

Entity-Relationship Model (ERM) is an abstract and conceptual representation of data. Entity-relationship modeling is a database modeling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database. In our ER diagram . For this in our component diagram first propose a data In this proposed method we are using Hash-Solomon Code Algorithm to encrypt the data. Then user2 want to access the file by the permission of user1 share the authenticated key.

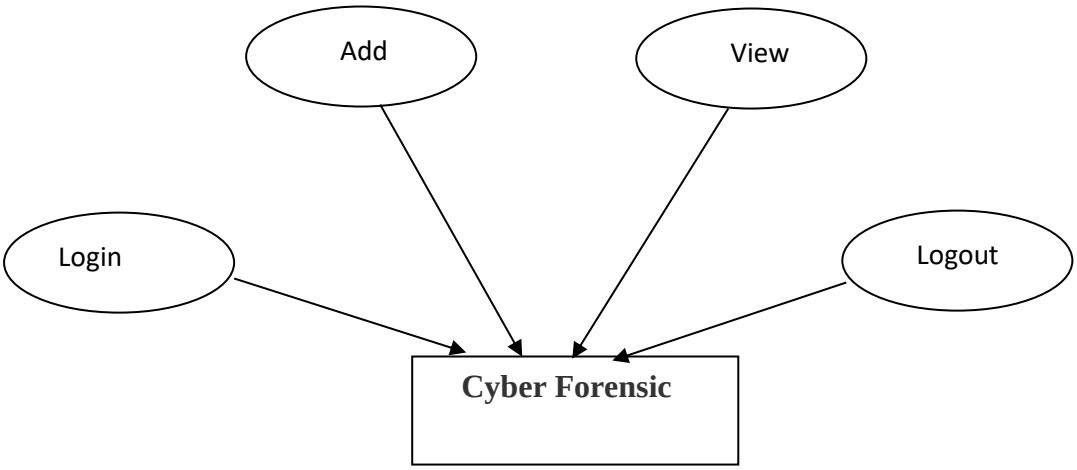


Fig- 4.5 E-R diagram

4.7 SYSTEM ARCHITECTURE

The systems architect establishes the basic structure of the system, we propose a Hashcode Solomon algorithm and a we can put a small part of data in local machine and fog server in order to protect the privacy. Moreover, based on computational intelligence, this algorithm can compute the distribution proportion stored in cloud, fog, and local machine, respectively. Through the theoretical safety analysis and experimental evaluation, the feasibility of our scheme has been validated, which is really a powerful supplement to existing cloud storage scheme

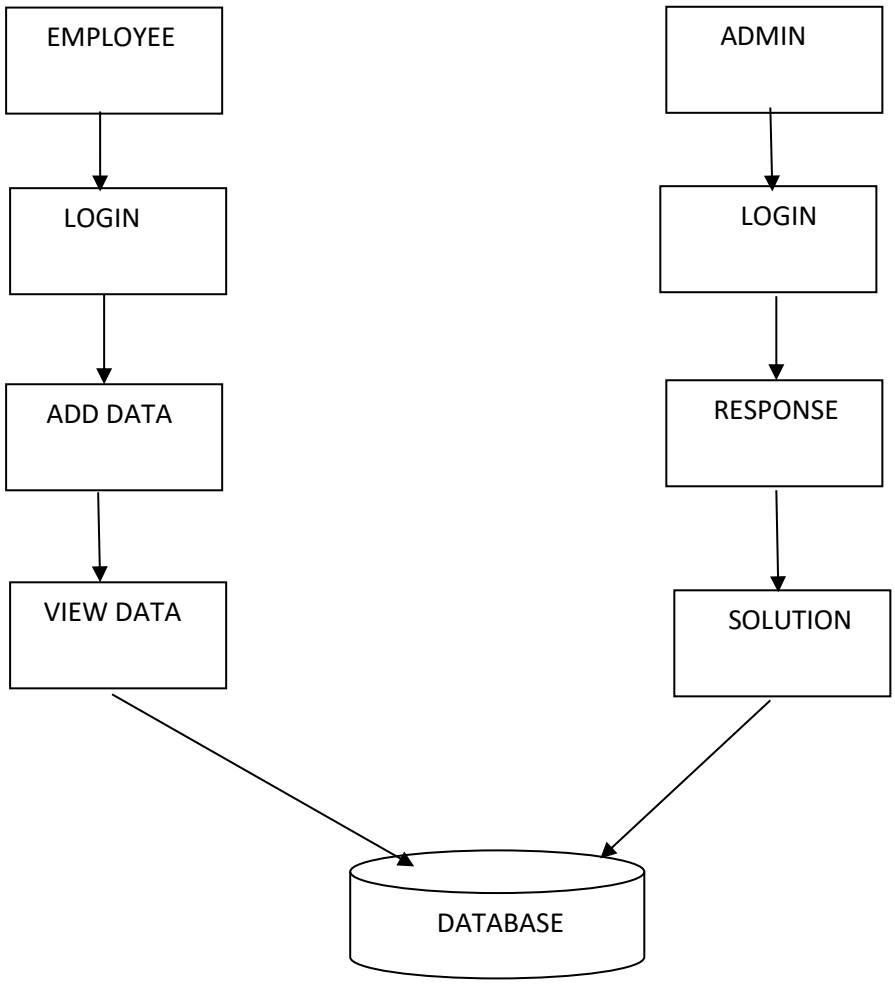


Fig- 4.6 System Architecture

4.8 ARCHITECTURE DIAGRAM

The structures architect establishes the primary shape of the system, we advise a Hashcode Solomon set of rules and a we will positioned a small a part of information in neighborhood gadget and fog server so that it will shield the privacy. Moreover, primarily based totally on computational intelligence, this set of rules can compute the distribution share saved in cloud, fog, and neighborhood gadget, respectively. Through the theoretical protection evaluation and experimental evaluation, the feasibility of our scheme has been validated, that is clearly a effective complement to present cloud garage scheme

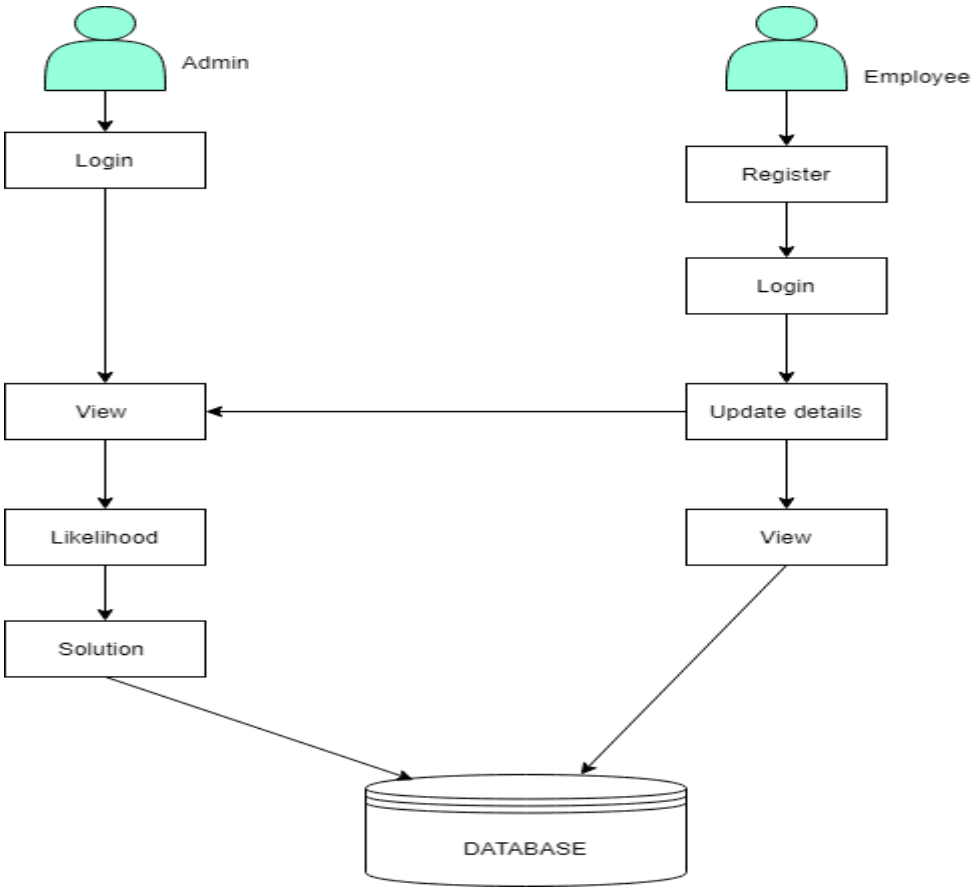


Fig- 4.7 Architecture Diagram

4.9 PROBLEM STATEMENT

"DATA-DRIVEN DECISION SUPPORT FOR OPTIMIZING CYBER FORENSIC INVESTIGATION" to solve that computer system as it secure, unusual, or Possible to Hack using Learning Algorithm:

➔ Naïve Bayes algorithm

- Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems.
- It is mainly used in *text classification* that includes a high-dimensional training dataset.
- Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.
- It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.
- Some popular examples of Naïve Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.

BAYES' THEOREAM:

- Bayes' theorem is also known as **Bayes' Rule** or **Bayes' law**, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.
- The formula for Bayes' theorem is given as:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Where,

P(A|B) is Posterior probability: Probability of hypothesis A on the observed event B.

P(B|A) is Likelihood probability: Probability of the evidence given that the probability of a hypothesis is true.

P(A) is Prior Probability: Probability of hypothesis before observing the evidence.

P(B) is Marginal Probability: Probability of Evidence.

CHAPTER 5

5.IMPLEMENTATION

5.1 USER INTERFACE DESIGN

5.1.1 Employee login

This module gives the way to enter into the main employee page after login with valid input such as username or email id and password. An employee can enter along with the valid text.

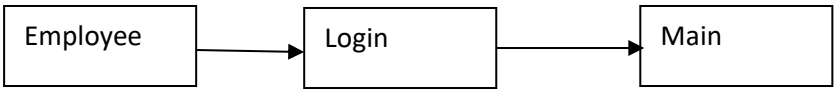
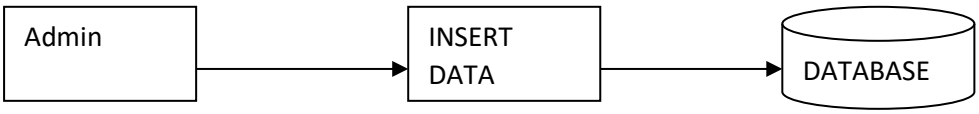


Fig- 5.1 Employee login

5.1.2 Add Report

This module is used to enter details into the database.



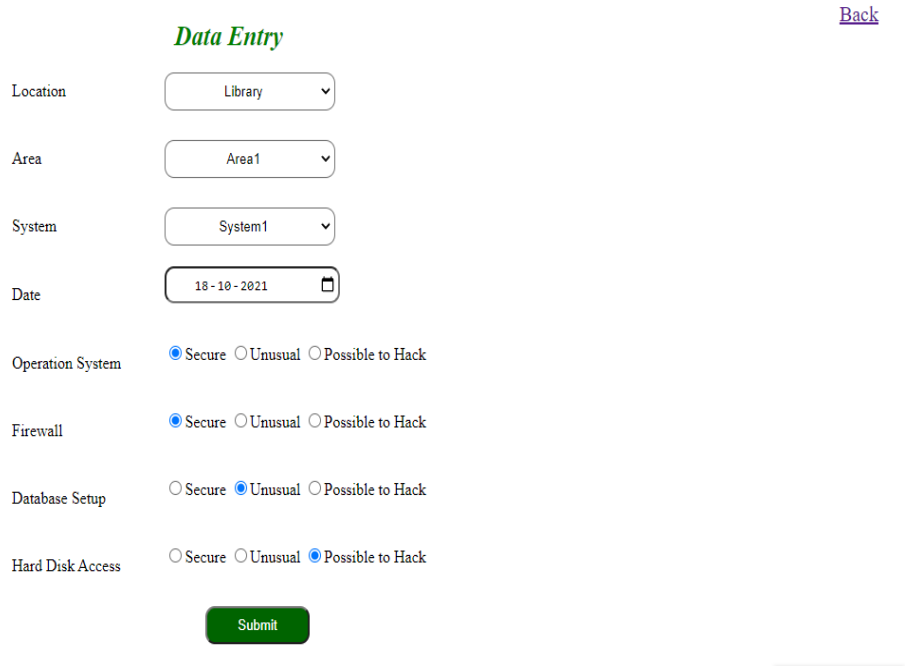
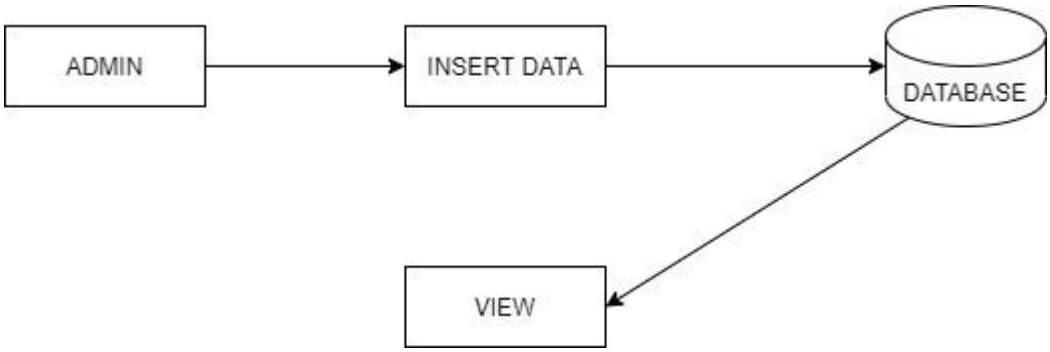


Fig- 5.2 Add Report

5.1.3 View Report

This module gives the way to use view the Details.



Back

Service Details

Location	Area	System	Date	OperatingSystem	Firewall	Database	Harddisk
library	area1	system1	2021-09-09	unusual	unusual	possible	possible
library	area2	system2	2021-09-09	secure	secure	unusual	possible
library	area4	system4	2021-09-09	possible	unusual	secure	possible
library	area1	system1		secure	unusual	unusual	secure
library	area1	system1	2021-09-09	unusual	possible	unusual	secure
library	area1	system1		unusual	secure	secure	unusual
library	area1	system1	2021-09-09	unusual	secure	secure	unusual
library	area1	system1	2021-09-10	unusual	secure	possible	secure
library	area1	system1	2021-09-10	possible	unusual	unusual	secure
eservicecentre	area1	system1	2021-09-17	secure	unusual	secure	unusual

Fig- 5.3 View report

5.1.4 Admin Login

This module gives way to This module gives the way to enter into Admin main page after login with valid input such as username or password use view the Details.

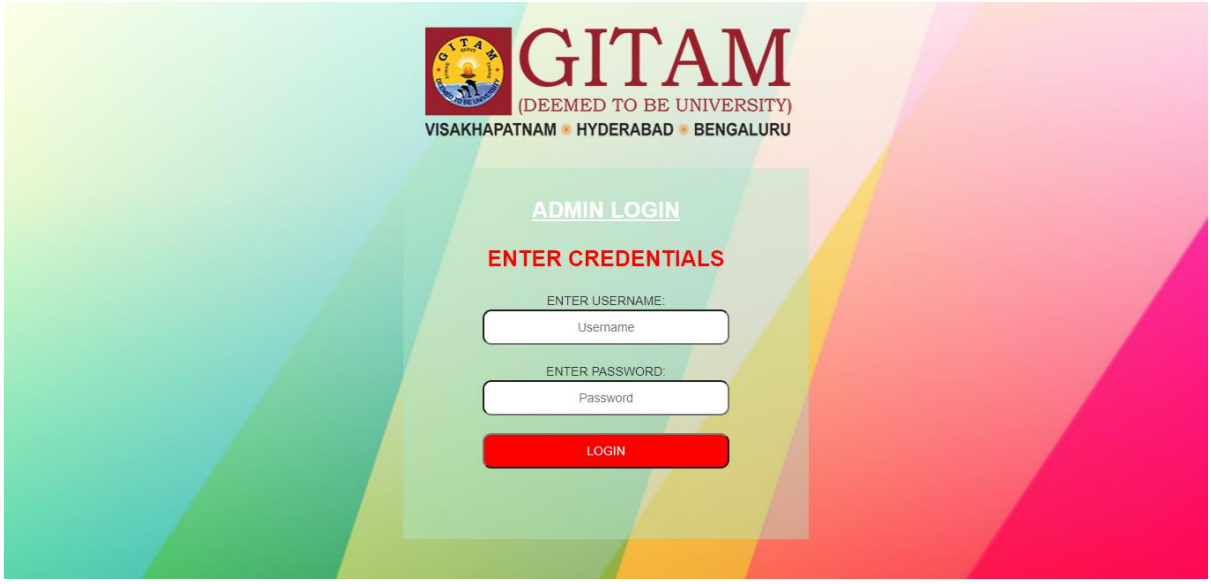
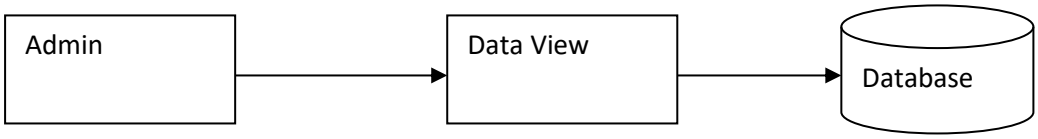


Fig- 5.4 Admin login

5.1.5 Admin View

This module helping the admin check and view the details in the database.



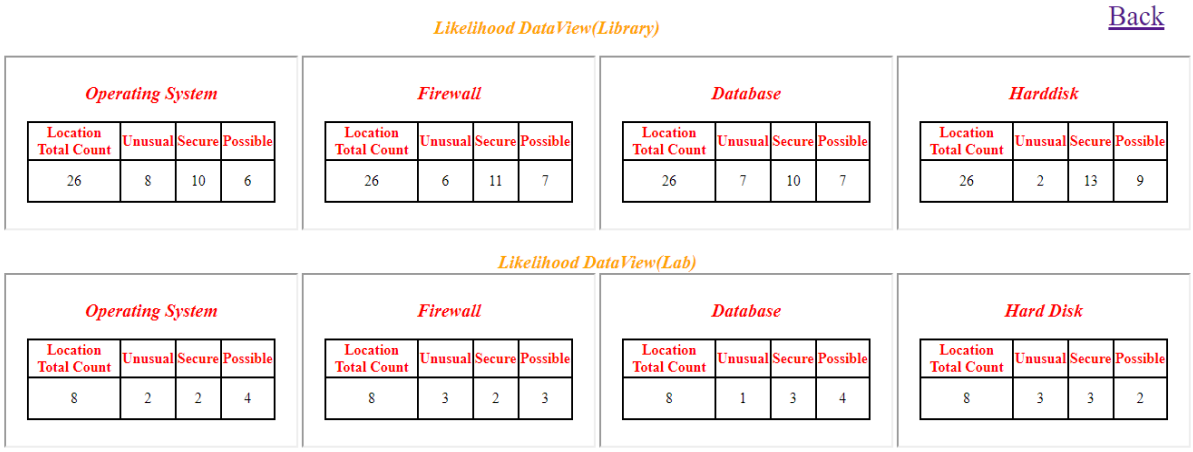
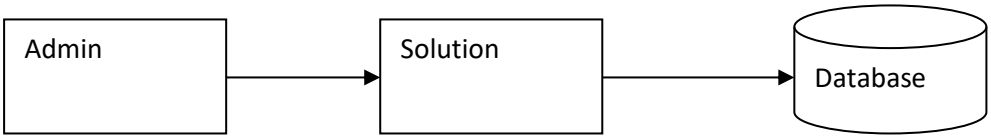


Fig- 5.5 Admin view

5.1.6 Admin Response

This module helping the check and response the files in the database.



Back

Solution View(Library)

Operating System

Unusual	Secure	Possible
0.9230769	0.7692308	0.9230769

Firewall

Unusual	Secure	Possible
0.9230769	0.84615386	0.8076923

Database

Unusual	Secure	Possible
0.8076923	0.7692308	0.8076923

Hard Disk

Unusual	Secure	Possible
1.0	1.0	0.6923077

Solution View(Lab)

Operating System

Unusual	Secure	Possible
1.0	1.0	1.0

Firewall

Unusual	Secure	Possible
0.75	1.0	0.75

Database

Unusual	Secure	Possible
1.0	0.75	1.0

Hard Disk

Unusual	Secure	Possible
0.75	0.75	1.0

Fig- 5.6 Admin response

5.2 ALGORITHM

The Naïve Bayes algorithm is comprised of two words Naïve and Bayes, Which can be described as:

- **Naïve**: It is called Naïve because it assumes that the occurrence of a certain feature is independent of the occurrence of other features. Such as if the fruit is identified on the bases of color, shape, and taste, then red, spherical, and sweet fruit is recognized as an apple. Hence each feature individually contributes to identify that it is an apple without depending on each other.
- **Bayes**: It is called Bayes because it depends on the principle of Bayes' Theorem.
-

5.2.1 Working of Naïve Bayes' Classifier

Working of Naïve Bayes' Classifier can be understood with the help of the below example:

Suppose we have a dataset of **weather conditions** and corresponding target variable "**Play**". So using this dataset we need to decide that whether we should play or not on a particular day according to the weather conditions. So to solve this problem, we need to follow the below steps:

1. Convert the given dataset into frequency tables.
2. Generate Likelihood table by finding the probabilities of given features.
3. Now, use Bayes theorem to calculate the posterior probability.

5.2.2 Advantages of Naïve Bayes Classifier

- Naïve Bayes is one of the fast and easy ML algorithms to predict a class of datasets.
- It can be used for Binary as well as Multi-class Classifications.
- It performs well in Multi-class predictions as compared to the other Algorithms.
- It is the most popular choice for **text classification problems**.

5.2.3 Disadvantages of Naïve Bayes Classifier

- Naive Bayes assumes that all features are independent or unrelated, so it cannot learn the relationship between features.

5.2.4 Applications of Naïve Bayes Classifier

- It is used for Credit Scoring.
- It is used in medical data classification.
- It can be used in real-time predictions because Naïve Bayes Classifier is an eager learner.
- It is used in Text classification such as Spam filtering and Sentiment analysis.

5.3 ALGORITHM IMPLIMENTATION

```
try{
        abyb=Integer.parseInt(locatlist)/Integer.parseInt(osunuselist);
        multia=abyb*Integer.parseInt(osunuselist);
        divb=multia/Integer.parseInt(locatlist);
        finallist=String.valueOf(divb);

}catch(ArithmeticException e){
        e.printStackTrace();
}
```

CHAPTER 6

6.TESTING AND TRAINING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

6.1 DEVELOPING METHODOLOGIES

The test process is initiated by developing a comprehensive plan to test the general functionality and special features on a variety of platform combinations. Strict quality control procedures are used.

The process verifies that the application meets the requirements specified in the system requirements document and is bug free. The following are the considerations used to develop the framework from developing the testing methodologies.

6.2 TYPES OF TESTS

6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program input produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.2.2 FUNCTIONAL TEST

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures : interfacing systems or procedures must be invoked.

6.2.3 SYSTEM TEST

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

6.2.4 PERFORMANCE TEST

The Performance test ensures that the output be produced within the time limits,and the time taken by the system for compiling, giving response to the users and request being send to the system for to retrieve the results.

6.2.5 INTEGRATION TESTING

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

6.2.6 ACCEPTANCE TESTING

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

- The Acknowledgements will be received by the Sender Node after the Packets are received by the Destination Node
- The Route add operation is done only when there is a Route request in need
- The Status of Nodes information is done automatically in the Cache Updation process

6.2.7 BUILD THE TEST PLAN

Any project can be divided into units that can be further performed for detailed processing. Then a testing strategy for each of this unit is carried out. Unit testing helps to identity the possible bugs in the individual component, so the component that has bugs can be identified and can be rectified from errors.

6.3 PERFORMED TESTCASES:

6.3.1 EMPLOYEE DATA:

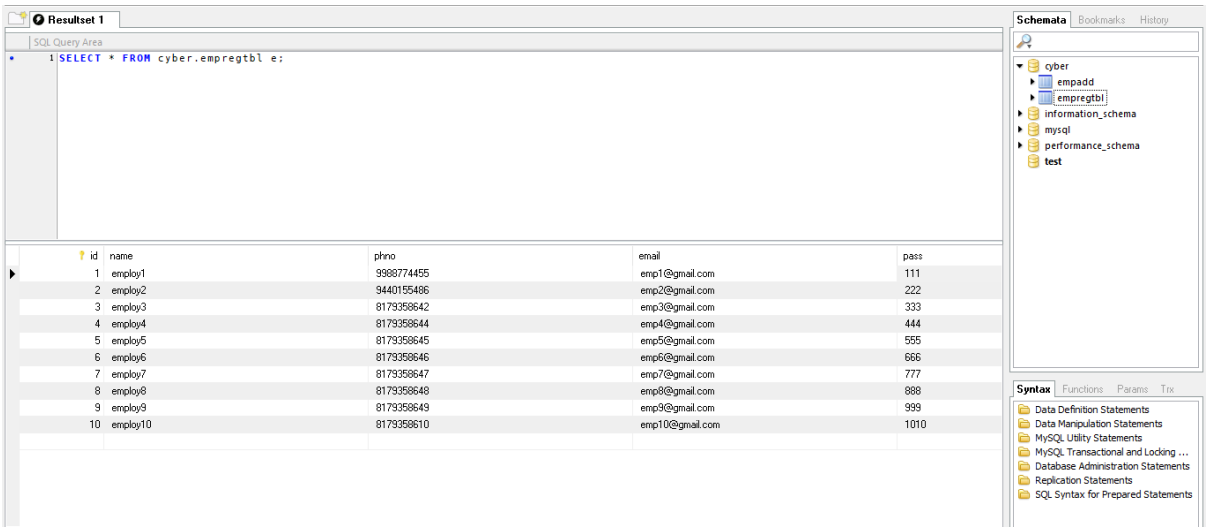


Fig- 6.1 EMPLOYEE DATA

6.3.2 EMPLOYEE DATA VIEW

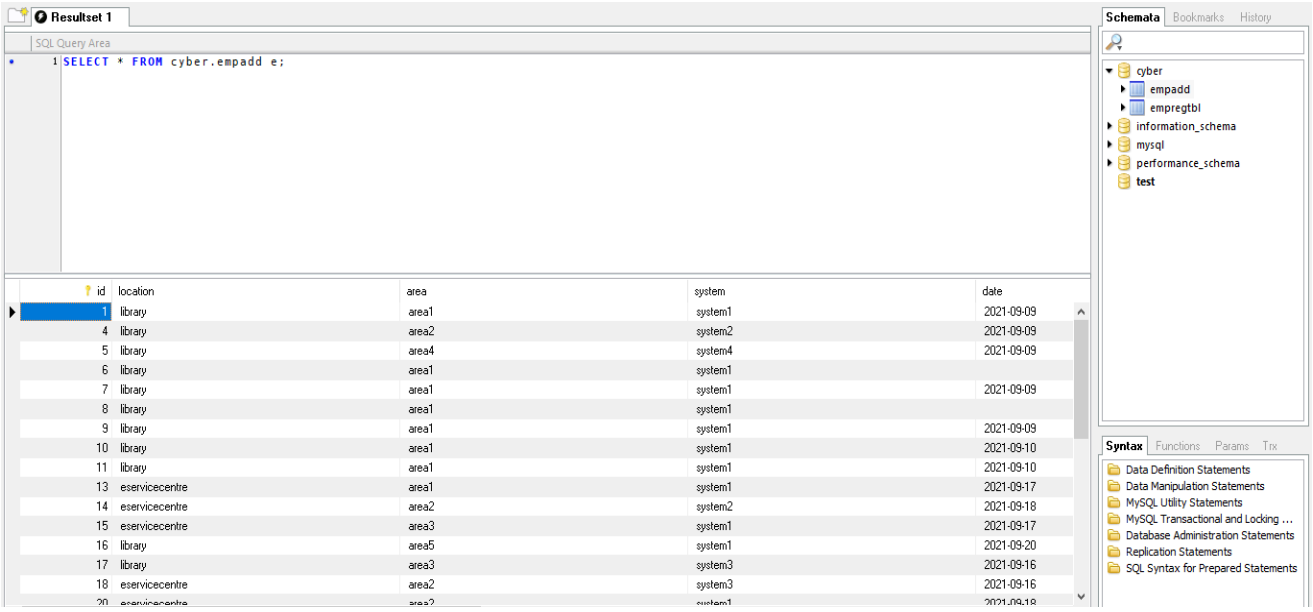


Fig- 6.2 EMPLOYEE DATA VIEW

CHAPTER 7

7.RESULT ANALYSIS

7.1 ANALYSIS

1. Missing Data

Naive Bayes can handle missing data.Attributes are handled separately by the algorithm at both model construction time and prediction time.As such, if a data instance has a missing value for an attribute, it can be ignored while preparing the model, and ignored when a probability is calculated for a class value.

2. Use Log Probabilities

Probabilities are often small numbers. To calculate joint probabilities, you need to multiply probabilities together. When you multiply one small number by another small number, you get a very small number.It is possible to get into difficulty with the precision of your floating point values, such as under-runs. To avoid this problem, work in the log probability space (take the logarithm of your probabilities).

This works because to make a prediction in Naive Bayes we need to know which class has the larger probability (rank) rather than what the specific probability was.

3. Use Other Distributions

To use Naive Bayes with categorical attributes, you calculate a frequency for each observation.To use Naive Bayes with real-valued attributes, you can summarize the density of the attribute using a Gaussian distribution. Alternatively you can use another functional form that better describes the distribution of the data, such as an exponential.

Don’t constrain yourself to the distributions used in examples of the Naive Bayes algorithm. Choose distributions that best characterize your data and prediction problem.

4. Use Probabilities For Feature Selection

Feature selection is the selection of those data attributes that best characterize a predicted variable.

In Naive Bayes, the probabilities for each attribute are calculated independently from the training dataset. You can use a search algorithm to explore the combination of the probabilities of different attributes together and evaluate their performance at predicting the output variable.

5. Segment The Data

Identifying and separating out segments that are easily handled by a simple probabilistic approach like Naive Bayes can give you increase performance and focus on the elements of the problem that are more difficult to model.

Explore different subsets, such as as the average or popular cases that are very likely handled well by Naive Bayes.

6. Re-compute Probabilities

Calculate the probabilities for each attribute is very fast.This benefit of Naive Bayes means that you can re-calculate the probabilities as the data changes. This may be monthly, daily, even hourly.

This is something that may be unthinkable for other algorithms, but should be tested when using Naive Bayes if there is some temporal drift in the problem being modeled.

7. Use as a Generative Model

The Naive Bayes method characterizes the problem, which in turn can be used for making predictions about unseen data.This probabilistic characterization can also be used to generate instances of the problem.

In the case of a numeric vector, the probability distributions can be sampled to create new fictitious vectors.

In the case of text (a very popular application of Naive Bayes), the model can be used to create fictitious input documents.

8. Remove Redundant Features

The performance of Naive Bayes can degrade if the data contains highly correlated features.This is because the highly correlated features are voted for twice in the model, over inflating their importance.Evaluate the correlation of attributes pairwise with each other using a correlation matrix and remove those features that are the most highly correlated.

Nevertheless, always test your problem before and after such a change and stick with the form of the problem that leads to the better results.

9. Parallelize Probability Calculation

The probabilities for each attribute are calculated independently. This is the independence assumption in the approach and the reason why it has it's name "naive".You can exploit this assumption to further speed up the execution of the algorithm by calculating attribute probabilities in parallel.

Depending on the size of the dataset and your resources, you could do this using different CPUs, different machines or different clusters.

10. Less Data Than You Think

Naive Bayes does not need a lot of data to perform well.It needs enough data to understand the probabilistic relationship of each attribute in isolation with the output variable.Given that interactions between attributes are ignored in the model, we do not need examples of these interactions and therefore generally less data than other algorithms, such as logistic regression.

Further, it is less likely to overfit the training data with a smaller sample size.

11. Zero Observations Problem

Naive Bayes will not be reliable if there are significant differences in the attribute distributions compared to the training dataset.An important example of this is the case where a categorical attribute has a value that was not observed in training. In this case, the model will assign a 0 probability and be unable to make a prediction.

These cases should be checked for and handled differently. After such cases have been resolved (an answer is known), the probabilities should be recalculated and the model updated.

12. It Works Anyway

An interesting point about Naive Bayes is that even when the independence assumption is violated and there are clear known relationships between attributes, it works anyway.

Importantly, this is one of the reasons why you need to spot check a variety of algorithms on a given problem, because the results can very likely surprise you.

7.1.1 LIKELIHOOD VIEW

```
<%
String idlist="";
String locatlist="";
String arealist="";
String systlist="";
String datelist="";
String oslist="";
String firelist="";
String dblist="";
String hdisklist="";
String osunuselist="";
String osseclist="";
String possiblist="";
String finallist="";
String libcount="";
String libcountsec="";
String libcountunuse="";
String libcountposs="";
String govthospcount="";
String powerstationcount="";
String eservicecount="";
String govtofficecount="";
```

```
float useint=0;
float possint=0;
float secint=0;
float abyb=0,multia=0,divb=0;
float solution=0;


%>

<%

Connection d4 = Dbconn.create();
PreparedStatement p4 = d4.prepareStatement("SELECT count(location) FROM
`cyber`.`empadd` where location='library'");
ResultSet rp4 = p4.executeQuery();

while (rp4.next()){

locatlist=rp4.getString(1);

%>

<%
}
%>

<%

Connection d = Dbconn.create();
PreparedStatement p = d.prepareStatement("SELECT count(location) FROM
`cyber`.`empadd` where location='library' and osystem='unusual'");
ResultSet rp = p.executeQuery();

while (rp.next()){

libcountunuse=rp.getString(1);

%>

<%
}
%>

<%

Connection d2 = Dbconn.create();
PreparedStatement p2 = d2.prepareStatement("SELECT count(location) FROM
`cyber`.`empadd` where location='library' and osystem='secure'");
ResultSet rp2 = p2.executeQuery();
```

```
while (rp2.next()){

libcountsec=rp2.getString(1);

%>

<%
}
%>

<%

Connection d3 = Dbconn.create();
PreparedStatement p3 = d3.prepareStatement("SELECT count(location) FROM
`cyber`.`empadd` where location='library' and osystem='possible'");
ResultSet rp3 = p3.executeQuery();

while (rp3.next()){

libcountposs=rp3.getString(1);

%>

<%
}
%>
```

OUTPUT:

Back

Likelihood DataView(Library)

Operating System

Location Total Count	Unusual	Secure	Possible
24	6	10	6

Firewall

Location Total Count	Unusual	Secure	Possible
24	6	11	5

Database

Location Total Count	Unusual	Secure	Possible
24	6	10	6

Harddisk

Location Total Count	Unusual	Secure	Possible
24	2	11	9

Likelihood DataView(Lab)

Operating System

Location Total Count	Unusual	Secure	Possible
6	2	2	2

Firewall

Location Total Count	Unusual	Secure	Possible
6	2	2	2

Database

Location Total Count	Unusual	Secure	Possible
6	0	3	3

Hard Disk

Location Total Count	Unusual	Secure	Possible
6	1	3	2

Fig- 7.1 LIKELIHOOD VIEW

7.1.2 ALGORITHM SOLUTION

```
<%
String idlist="";
String locatlist="";
String arealist="";
String systlist="";
String datelist="";
String oslist="";
String firelist="";
String dblist="";
String hdisklist="";
String osunuselist="";
String osseclist="";
String possiblist="";
String finallist="";
String finallist2="";
String finallist3="";

float useint=0;
float possint=0;
float secint=0;
float abyb=0,multia=0,divb=0;
float abyb2=0,multia2=0,divb2=0;
float abyb3=0,multia3=0,divb3=0;
float solution=0;



%>

<%

        Connection d4 = Dbconn.create();
        PreparedStatement p4 = d4.prepareStatement("SELECT count(location) FROM
`cyber`.`empadd` where location='library'");
       ResultSet rp4 = p4.executeQuery();

                while (rp4.next()){

                        locatlist=rp4.getString(1);

       %>

       <%
       }
       %>

<%
```

```
 Connection d = Dbconn.create();
 PreparedStatement p = d.prepareStatement("SELECT count(*) FROM
`cyber`.`empadd` where location='library' and osystem='unusual'");
ResultSet rp = p.executeQuery();

        while (rp.next()){

                osunuselist=rp.getString(1);

%>

<%
}
%>

<%

 Connection d2 = Dbconn.create();
 PreparedStatement p2 = d2.prepareStatement("SELECT count(*) FROM
`cyber`.`empadd` where location='library' and osystem='secure'");
ResultSet rp2 = p2.executeQuery();

        while (rp2.next()){

                osseclist=rp2.getString(1);

%>

<%
}
%>

<%

 Connection d3 = Dbconn.create();
 PreparedStatement p3 = d3.prepareStatement("SELECT count(*) FROM
`cyber`.`empadd` where location='library' and osystem='possible'");
ResultSet rp3 = p3.executeQuery();

        while (rp3.next()){

                possiblist=rp3.getString(1);

%>

<%
}
%>
```

```
<%


try{
        abyb=Integer.parseInt(locatlist)/Integer.parseInt(osunuselist);
        multia=abyb*Integer.parseInt(osunuselist);
        divb=multia/Integer.parseInt(locatlist);
        finallist=String.valueOf(divb);

}catch(ArithmeticException e){
        e.printStackTrace();
}




%>

<%
try{
abyb2=Integer.parseInt(locatlist)/Integer.parseInt(osseclist);
multia2=abyb2*Integer.parseInt(osseclist);
divb2=multia2/Integer.parseInt(locatlist);
finallist2=String.valueOf(divb2);
}catch(ArithmeticException e){
        e.printStackTrace();
}

%>

<%
try{
abyb3=Integer.parseInt(locatlist)/Integer.parseInt(possiblist);
multia3=abyb3*Integer.parseInt(possiblist);
divb3=multia3/Integer.parseInt(locatlist);
finallist3=String.valueOf(divb3);

}catch(ArithmeticException e){
        e.printStackTrace();
}


%>
```

OUTPUT:

Back

Solution View(Library)

Operating System

Unusual	Secure	Possible
1.0	0.8333333	1.0

Firewall

Unusual	Secure	Possible
1.0	0.9166667	0.8333333

Database

Unusual	Secure	Possible
1.0	0.8333333	1.0

Hard Disk

Unusual	Secure	Possible
1.0	0.9166667	0.75

Solution View(LAB)

Operating System

Unusual	Secure	Possible
1.0	1.0	1.0

Firewall

Unusual	Secure	Possible
1.0	1.0	1.0

Database

Unusual	Secure	Possible
	1.0	1.0

Hard Disk

Unusual	Secure	Possible
1.0	1.0	1.0

Fig- 7.2 ALGORITHM SOLUTION

CHAPTER 8

8.CONCLUSION & FUTURE SCOPE

Digital forensics involves the process of identifying, collecting, acquiring, preserving. analysing, and presenting of digital evidence. Digital evidence must be authenticated to ensure its admissibility in a court of law. Ultimately, the forensic artefacts and forensic methods used to static or live acquisition depend on the device, its operating system, and its security features. Proprietary operating systems and security features serve as impediments to digital forensics. Real evidence must be competent (authenticated), relevant, and material. For example, a computer that was involved in a court matter would be considered real evidence provided that it has not been changed, altered, or accessed in a way that destroyed the evidence. Current web application has designed with employees and single expert or administrator to check and give the suggestion to the employees from their report to optimize their process. First employee login and make the report to authority person. Then, they may give their response from these report.

Future enhancement of project have designed the model with double expert or administrator to check and give the suggestion to the employees with the authorization services. One phase will be implemented under various business sectors such as banking and finance, healthcare, transportation, manufacturing, etc. Another phase cyber crime report will be added.

REFERENCES

[1] K. Finnerty, S. Fullick, H. Motha, J. N. Shah, M. Button, and V. Wang, “Cyber security breaches survey 2019,” 2019.

[2] I. Security, “Cost of a data breach report 2019,” 2019.

[3] V. Diaz, D. Emm, and C. Raiu, “Kaspersky security bulletin 2019: Advanced threat predictions for 2020,” 2019.

[4] J. Navarro, A. Deruyver, and P. Parrend, “A systematic survey on multistep attack detection,” Computers & Security, vol. 76, pp. 214–249, 2018. [11] A. L. B

[5] V. S. Harichandran, F. Breitinger, I. Baggili, and A. Marrington, “A cyber forensics needs analysis survey: Revisiting the domain’s needs a decade later,” Computers & Security, vol. 57, pp. 1–13, 2016.

[6] L. Martin, “Cyber kill chain®,” URL: http://cyber. lockheedmartin. com/hubfs/Gaining the Advantage Cyber Kill Chain. pdf, 2014.

[7] S. Barnum, “Standardizing cyber threat intelligence information with the structured threat information expression (stix),” Mitre Corporation, vol. 11, pp. 1–22, 2012.

[8] J. Williams, “Acpo good practice guide for digital evidence,” Metropolitan Police Service, Association of chief police officers, GB, 2012

[9] A. Brinson, A. Robinson, and M. Rogers, “A cyber forensics ontology: Creating a new approach to studying cyber forensics,” Digital Investigation, vol. 3, pp. 37–43, 2006.

[10] K. Kent, S. Chevalier, T. Grance, and H. Dang, “Guide to integrating forensic techniques into incident response,” NIST Special Publication, vol. 10, no. 14, pp. 800–86, 2006.

REFERENCE LINKS

i. https://www.javatpoint.com/machine-learning-naive-bayes-classifier
ii. https://machinelearningmastery.com/better-naive-bayes/
iii. https://austinpublishinggroup.com/forensicscience-criminology/fulltext/ajfsc-v5-id1076.pdf
iv. https://link.springer.com/chapter/10.1007/0-387-24230-9_13
v. https://www.computer.org/publications/tech-news/research/digital-forensics-security-challenges-cybercrime
vi. https://www.researchgate.net/publication/328520438_Get_Your_Workload_in_Order_Game_Theoretic_Prioritization_of_Database_Auditing
vii. https://www.itgovernance.co.uk/what-is-cybersecurity#:~:text=Cyber%20security%20is%20the%20application,of%20systems%2C%20networks%20and%20technologies.
viii. https://innocenceproject.org/forensic-science-problems-and-solutions/
ix. https://austinpublishinggroup.com/forensicscience-criminology/fulltext/ajfsc-v5-id1076.pdf
x. https://www.uowdubai.ac.ae/degrees/bachelors/computer-science/bachelor-computer-science-cyber-security