Group No. 1

Section L2
Instructor: Dr Shazi Shah Jabeen

	Group Members	ID Number
1.	Anoushka Martin	2022A9PS0129U
2.	Dev Dahiya	2022A7PS0319U
3.	Dhanush Krishna	2022A7PS0302U
4.	Drisya Manikkath	2022A9PS0254U
5.	Karthik Narayanan Sudheer	2022A7PS0321U
6.	Mohamad Shamshuddin Ganagavali	2022A7PS0333U
7.	Sivaa Balamurugan	2022A7PS0323U
8.	Xec Abdul Kader	2022A7PS0252U

Data Leak Detection System
Seattle, Washington
United States

A Report

on

Social Cybersecurity and Data Leak Detection

Prepared
for
Dr. Margaret Chan
Director General

Prepared
by
Dr. James Wilson
Head, Research Department

Approved by
Dr. Margaret Chan
Director General

April, 2023

ABSTRACT

Data leaks can result in significant reputational damage and financial losses for organizations. Detecting and preventing data loss is a pressing security concern, and despite numerous research efforts, it remains an active problem. This study covers enterprise data leakage dangers, recent occurrences, prevention and detection strategies, as well as new problems and solutions. Although effective protection can be deployed during data storage, usage, or transit, it is crucial to understand the threats and limitations of various solutions to make informed security decisions. The study examines big data leaks, systematizes knowledge on protection and detection, and emphasizes new research prospects in deep learning, privacy, and experimental reproducibility. The research offers a new model that merges a file-management system with a data leakage detection system in the background using a daemeon thread algorithm, giving a one-stop solution for data leakage detection.

ACKNOWLEDGEMENTS

We express our sincere and heartfelt gratitude to Dr. Srinivasan Madapusi, Director of BITS Pilani, Dubai Campus for giving us this opportunity to apply and understand engineering concepts in a practical atmosphere.

We thank Dr. Shazi Shah Jabeen for her valuable insights and continuous support throughout this project. Her guidance and encouragement were instrumental in its successful completion.

We are highly indebted to Ms. Sapna Sadhwani, for her support and advice which pointed us in the right direction for further research in the computational part of this project.

Finally, we would like to thank everyone who supported us during the completion of this report, from loved ones to survey participants, without whom this report would not have been possible.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1. Authorization

This report on Social Cybersecurity and Data Leak Detection was approved and authorized by Dr. Shazi Shah Jabeen, Professor of Humanities and Social Sciences, BPDC on February 24th, 2023.

1.2. Background of the Study

With most enterprises going online, there has been a growth in the number of data leaks and cyber attacks. Data leakage can be caused due to malware hacking, employee error, or negligence. It involves the loss of personal information, financial data , intellectual property and other sensitive data. Its consequences involve financial loss, reputational damage, legal liabilities, and regulatory penalties.

IDES (Intrusion Detection Expert System) developed in the 1980s by Dororthy Denning and Peter Neumann, is one of the earliest data leakage detection models. It was a rule based detection software which monitored computer systems and networks. The 1990s saw the development of anomaly detection techniques which addressed the limitations of IDES. However, these techniques showed too many false alerts to be implemented on a wide scale. Machine learning algorithms such as neural networks and support vector machines were introduced in the early 2000s. They were able to classify data as normal or anomalous based on training data. Deep Learning algorithms, which were developed in recent years, provide protection against complex threats and mitigate threats in many data leak situations. The constant growth of data leakage detection models shows the requirement for constant research in this department as these models need to grow with the increasing complexity of threats.

A prominent example of data leakage occurred in 2016, when attackers gained access to the personal GitHub account of an engineer who worked at Uber. They were able to access datastores and one of its internal repositories. These data repositories held unencrypted personal information on 57 million drivers and riders of Uber. They later contacted Uber and demanded a ransom to delete stolen data. Uber chose not to declare the breach and pay the ransom to keep the incident quiet. Later, the breach was disclosed following the appointment of a new CEO in 2017. Uber had to pay $100,000 in ransom and $148,000,000 as part of a settlement in court.

The Uber data loss incident highlights the importance of strong cybersecurity measures including proper access controls and incident response protocols as well as the ethical and legal responsibilities of companies to protect their customers' data and disclose security incidents in a timely and transparent manner. It also shows the need for further research in the field of data leak prevention and detection.

1.3. Objectives

The following are the report's objectives :

- To explore the causes and methods of data leakage.
- To study pre-existing data leakage models, such as encryption, access controls and watermarking.
- To build an application [DLPS – Data Leak Prevention System] that helps in detecting data which has been leaked and finding the guilty agent from the given set of agents using probability distribution and algorithms.
- To propose a better model that serves as both a file management system as well as a data leak detection software that works on daemeon thread running in the background.

1.4. Scope

- To study various types of data leakage- insider threats, external attacks, and accidental leakage.
- To explore the possibilities of using machine learning algorithms in increasing the efficiency and practicality of data leakage detection.
- To study the risks associated with data leakage and the development of risk assessment and management strategies to mitigate losses.

1.5. Limitations

- The search period of approximately four months is relatively short and does not cover all the research done on the matter.
- Lack of transparency in the AI decision making process. In the decision making process of AI, all the participants, including programmers, do not know why the AI model gives the final decision results. Although the AI model can achieve high accuracy, the tests are all implemented in the test set. Therefore, when facing unknown events, whether the AI model can achieve such a high accuracy remains to be verified.
- The AI models need a lot of data to complete the training. Before using data, they may do operations which mainly include a series of steps such as data noise reduction, normalization, missing value filling, etc.

1.6. Methods and Sources of Data Collection

Since data leakage prevention and social cybersecurity are relatively new topics, there is little printed material on them. Hence, the main source of information has been research papers published on the internet on websites such as IEEE Xplore and Academia.

Case studies of prominent data leakage incidents of various companies have also been a great resource in identifying types of data leakage. They have given important insights on detecting the channels through which data is being leaked and the methods to prevent the leaks.

1.7. Report Preview

Data leakage is a significant security threat that can originate from both malicious external actors and insiders. Effective prevention of data breaches requires the implementation of technical and administrative controls such as access control, network segmentation, and user security awareness. Data Leakage Prevention and Detection (DLPD) tools such as encryption, firewalls, Intrusion Detection Systems (IDS), and content-based methods can help detect and prevent data leaks.

A proposed system for detecting data leaks combines user authentication, file management, and email functionality, which improves the efficiency and security of user operations. It is crucial for organizations to implement comprehensive security measures and training programs to mitigate data leakage threats. Prioritizing data security is essential to minimize the risk of data leakage and improve overall security posture.

CHAPTER 2: DISCUSSION

2.1. Understanding Data Leakage:

Data leaking is the unethical movement of private information by a person to an outside recipient or destination. It may be explained as when a distributor of data gives information to a presumably reliable agent, that is then leaked and discovered in an unapproved area.

2.1.1. Types of Leaked Data

1. Personal Identifiable Information such as name, date of birth, home address, and email address are examples of PII that may be used to identify a person.
2. Financial data, which may be used for fraudulent acts if gained by unauthorized people, including details such as bank account and credit card numbers.
3. Intellectual property, which includes private company data, including business plans, trade secrets, financial reports, patents, and information on research and development.
4. Health-related information, genetic data, and prescription drugs, which are all considered to be medical data.

2.1.2. Modes of Transmission

1. Email : Unsecured email communication, since emails can be intercepted, forwarded, or accessed by unauthorized people.
2. Storage Devices : Private information is kept on portable storage devices, such as USB drives, which are prone to theft or loss.
3. Cloud Services: If not adequately protected, cloud-based systems for collaboration and storage are susceptible to data leakage.

4. Viruses : Trojans, worms, and other types of malware may infect systems and provide unauthorized users access to private information.
5. Social Engineering: It involves techniques like phishing and pharming to deceive people into exposing private information.
6. Employees or anyone with access to sensitive information may accidentally or purposefully leak information, posing an insider threat.

2.2. Classification of Data Leakage Threats:

Threats from data leaks are categorized in a variety of ways using different taxonomies. These taxonomies will be used in this section to identify and characterize the main risks that might result in data leaking. Then, after analyzing and debating a number of data breach cases in businesses, we will draw out the most important conclusions and lessons.

2.2.1. Based on Cause:

1. Intentional: Involves insiders or outsiders with access to the data purposefully disclosing private information. This kind of data leaking could be carried out for nefarious purposes that include espionage, financial gain, and retaliation. Through a variety of methods, such as email, portable storage devices, cloud services, and social engineering techniques, intentional data leakage can happen. A loss of competitive advantage, damage to an organization's reputation, and legal culpability are just a few examples of the serious harm it can do.

2. Inadvertent : An unintentional action by insiders or workers that results in an inadvertent data leakage; a sort of data breach. This involves disclosing private information on accident, sending emails to the incorrect recipient, or misplacing a laptop or storage device that contained private information. Accidental data leakage frequently results from human error, ignorance, and inadequate training.

2.2.2. Based on Party that is responsible for the Leak:

1. Insider threats: Involve insiders who have access to the data purposefully disclosing critical information. This includes cyber sabotage and espionage.

2. Outsider Threats : Brought on by malware, viruses, and social engineering; an attacker could use a system's backdoor or improperly set access restrictions, for instance, to get around authentication and access private data. Attacks on businesses using social engineering techniques like phishing and pharming are getting increasingly complex, deceiving people into giving sensitive corporate information to criminals.

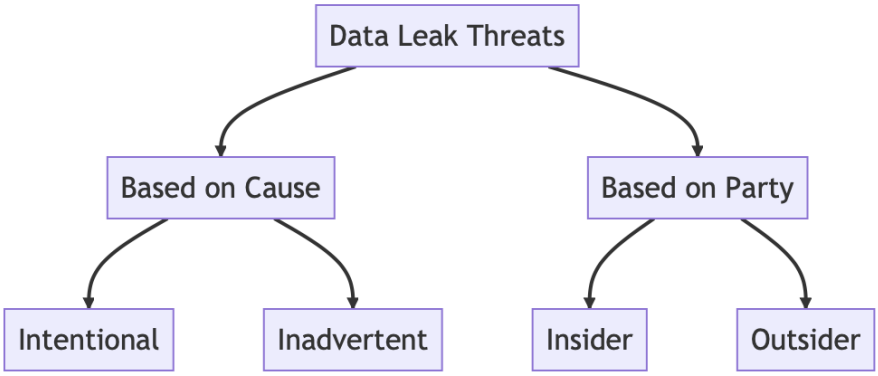


Figure 2.1: Classification of Data Leakage Threats

2.2.3. Based on Industry Sector:

The overall number of significant data leakage events has increased over the last five years, as shown by the ITRC in Figure 2.2. For instance, compared to 2015, the amount of data leaks increased by 40% in 2016.

An industry-specific data leak is shown in Figure 2.2(a) using a stacked histogram display. The bulk of the leaks occur in the commercial and healthcare sectors. Business data leaks accounted for 45.2% of all breaches in 2016 while medical/healthcare breaches accounted for 34.5 percent.

2.2.4. Based on Type of Occurence:

Figure 2.2(b) shows data leaks by type of occurrence, with the 'other' category including employee error, email/internet exposure, etc. The graph indicates that in 2016, over 55% of all leaks were caused by malevolent outsiders. All of these reports, including the ITRC statistics, confirm the trend that insider threats emerge as the leading cause of enterprise data leak threats, with more than 40% of breaches being perpetrated from within a company. This is true even though different cybersecurity reports may produce different results because they use nonidentical datasets.

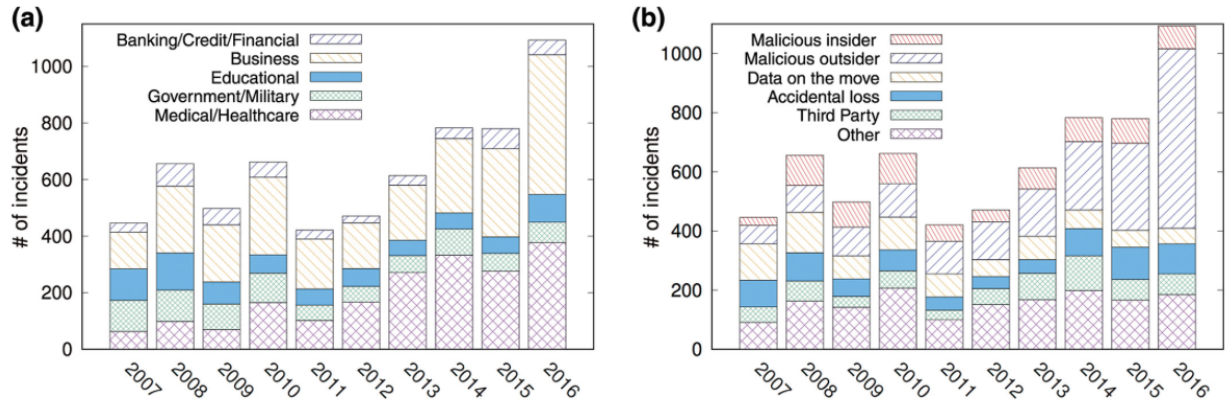


Figure 2.2 : Statistics of Data Leakage Reports from 2007 - 2016
(a) Industry Sector and (b) Type of Occurrence

2.3. Incidents of Data Leakage:

As demonstrated in Table 1, large-scale data leaks have increased in frequency recently, causing huge monetary losses and the release of millions of people's personal information. The main causes of these occurrences are insider threats and external cyberattacks. We will carefully explore the Target data breach as a significant instance of an outside assault that led to data leakage.

Organization	Records	Breach Date	Type	Source	Industry	Estimated Cost
Anthem insurance	78 million	January 2015	Identify theft	Malicious outsider	Healthcare	$100 million
Yahoo	500 million	December 2014	Account access	State sponsored [1]	Business	$350 million
Home depot	109 million	September 2014	Financial access	Malicious outsider	Business	$28 million
JPMorgan chase	83 million	August 2014	Identify theft	Malicious outsider	Financial	$13 billion
Benesse	49 million	July 2014	Identify theft	Malicious insider	Education	$138 million
Korea credit bureau	104 million	January 2014	Identify theft	Malicious insider	Financial	$100 million
Target	110 million	November 2013	Financial access	Malicious outsider	Business	$252 million
Adobe System	152 Million	September 2013	Financial access	Malicious outsider	Business	$714 Million

Figure 2.3 : Compiled List of the Biggest Enterprise Data Leaks

2.3.1. Internal Data Leak Incidents:

Accidental data breaches have increased in frequency during the last several years, as shown by cases like the Australian Red Cross Blood Service data leak in 2016, in which the personal information of nearly 550,000 blood donors was inadvertently posted on an unprotected online directory. Sensitive information including medical histories, drug usage, and sexual activities were exposed in the incident. Another instance occurred in 2011, when a Texas state owned state server unintentionally posted the personal data of 3.5 million individuals online for a year.

There have been allegations of a number of harmful insider threats, including the removal and exfiltration of private information, listening in on conversations, and installing software with backdoors. Over 250,000 classified U.S. diplomatic cables were leaked in 2010 by an insider group using a secondary hard drive, and they were subsequently sent to WikiLeaks. A Vodafone Germany IT contractor broke into the company's database system in 2013 and stole the personal information and bank account numbers of almost two million consumers. As medical records have been digitized, there have been examples of insiders stealing patient data, such as the theft of 14,000 patient records by a former worker at UMass Memorial Medical Center in 2015.

Insiders frequently have legitimate access to data infrastructure, and they may employ covert channels and steganography techniques to avoid discovery, making it difficult to identify internal data breaches. To stop accidental data leakage, workplace security knowledge needs to be raised.

2.3.2. External Data Leak Incidents:

Organizations like Yahoo and Target suffered major financial losses as a result of many high-profile data dumps. By 2016, Yahoo had had two significant data leaks, the first of which happened in late 2014 and affected over 500 million account holders. Over 1 billion user accounts were compromised by the second, which was found in December 2016, in August 2013. Due to these events, Verizon was able to purchase Yahoo for $350 million less than the initial sale price. One of the biggest stores in the United States, Target Corporation, had its data security compromised in 2013, allowing cybercriminals to obtain PII from almost 70 million users.

The actions taken by the perpetrators in the Target data breach are shown in Figure 2.4. Attackers began a phishing assault in September 2013 on Fazio Mechanical Services, a third-party vendor with access privileges to Target's network. The attackers were able to access weak workstations in Target's network through Fazio. To steal sensitive data, they later installed the data-stealing virus BlackPOS on point-of-sale (POS) systems. Prior to being transmitted to servers outside of the Target network, the stolen information was encrypted and delivered to internal servers that were compromised.

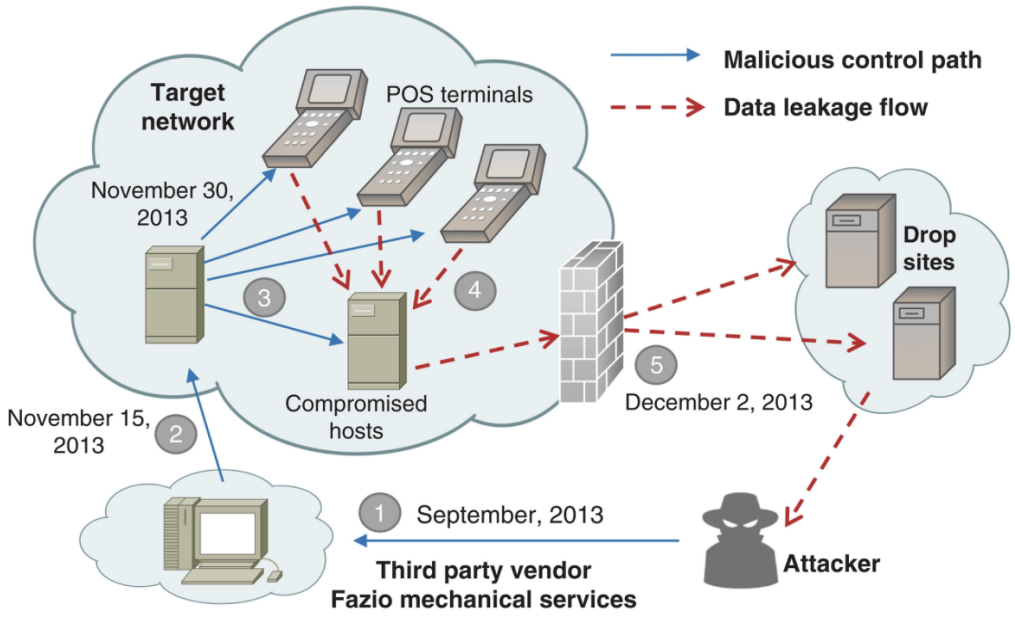


Figure 2.4 : Analysis of a Typical Data Breach

The Target data breach serves as a reminder of the significance of implementing suitable Data Loss Prevention and Detection (DLPD) technological and administrative measures to stop external threats posing a risk of data leak. There are a number of technical reasons why Target was unable to stop or identify the breach at various times. First off, the initial hackers' break-ins were allowed by Target not using proper access control methods on third-party partners. Furthermore, it did not separate delicate systems for payment from the rest of its infrastructure. Thirdly, Target failed to secure the POS systems, allowing illegal software to be set up and installed.

2.4. Existing Data Leak Prevention and Detection Models:

Basic security precautions and defined approaches are two categories under which DLPD tactics fall. DLPD technologies are especially made to identify and prevent data leaks using content and context monitoring, while basic security measures offer generic protection. They're aimed to protect private information from unauthorized use. Tools for designated DLPD are gaining popularity and will be a crucial part of organizational security.Techniques for detecting and preventing data leakage in a company's system are shown in Figure 4. Data at rest is protected by fundamental security methods including access restriction, encryption, and safe data publication. Network access is restricted by firewalls, and intrusion detection systems (IDS) keep an eye out for unwanted access. Data-stealing malware is detected by antivirus software, and private files are safeguarded by trusted computing technologies and virtual computers. However, IDS has a significant potential for false positives.

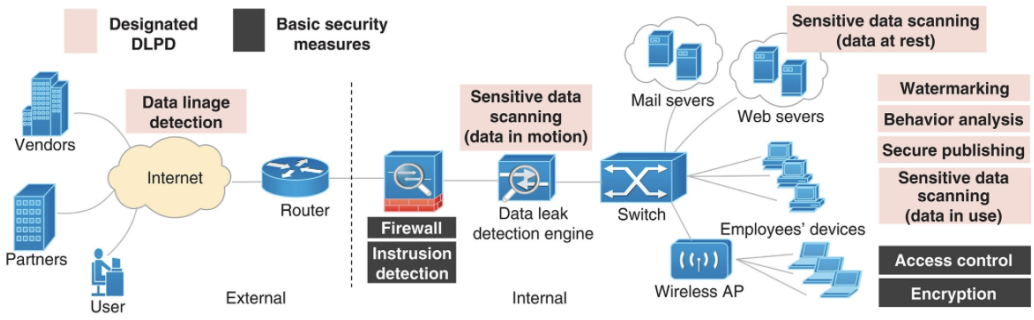


Figure 2.5 : There exists multiple points for deploying data leak prevention and detection techniques.

2.4.1. Content-based Data Loss Prevention

Content-based techniques, also known as sensitive data scans, look at the data's actual contents to avoid unintentionally disclosing private information in a variety of contexts such as while it is idle, in use, or in transmission. Although this method can be useful in limiting unintentional data leakage, attackers from the inside and the outside who try to obfuscate the data can get around it.

These techniques are employed to look for sensitive data on gadgets or in network activity. These techniques rely on statistical analysis, semantic content analysis, and data fingerprinting. Lexical analysis employs regular expressions to locate sensitive material that follows basic patterns, whereas data fingerprinting recovers traces of known potentially sensitive information to detect data leaks. Examining the frequency of successive byte sequences in a document is known as statistical analysis. The intersection rate between monitored content sequences and sensitive data sequences is calculated using a statistical analysis technique called collection intersection. When balanced by the size of the smallest collection, the intersection rate is calculated as the total of all item occurrence frequencies in the intersection.

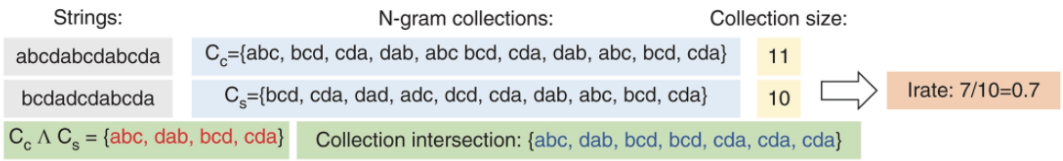


Figure 2.6 : Demonstrates calculation of the intersection rate between the two 3-gram collections.

Unlike set intersection, collection intersection counts duplicated elements.

An increasing number of methods based on machine learning are being used to identify sensitive data that has to be protected. For instance, Symantec employs vector machine learning technology, which gains accuracy via training, to find sensitive data in unstructured data. In order to recognize and differentiate between enterprise documents that contain sensitive information and those that do not, a number of organizations have developed text classification algorithms based on machine learning. Some organizations utilize statistical analysis techniques to interpret the significance of confidential data embedded within complex data sets, which may not be apparent through other means.

2.4.2. Context-based Data Loss Prevention

Contrary to content-based techniques, context-based approaches concentrate on examining the meta data linked to the data or the environment in which the data is found rather than on determining the existence of sensitive material.

Context-based methods for detecting insider threats concentrate on examining user behavior and activity patterns in order to spot abnormalities or departures from the norm. These methods involve monitoring user activity, profiling users' typical behavior, and spotting unusual behavior using data mining or machine learning methods. Machine learning-based techniques have the benefit of automatically identifying outliers without the need to fully define aberrant behaviors. The absence of training data is a problem for employing machine learning or data mining for anomaly detection.

Vulnerable data points are marked with watermarking to prevent and identify data breaches, whereas trap-based defenses utilize honeypots to lure and deceive insiders into disclosing their nefarious intentions. These techniques can also be utilized for post-incident forensic investigation to locate the leaker. Incorporating bogus information into the distributed data can assist in identifying the leaker, and data allocation algorithms can be used to determine the chance that an agent is to blame for a leak.

2.4.3. Outcome

Since the emphasis is on identifying the actual information itself, content-based approaches are typically superior to context-based methods in detecting the leakage of sensitive material. Therefore, content analysis for the identification of sensitive data has been the focus of the majority of the research in this field.

Data scanning can be used at several phases of deployment to secure data. Finding possible data leak threats within the enterprise can be aided by scanning data that is at rest or kept on servers. Monitoring data use can stop sensitive data from being handled carelessly and can stop efforts to transmit sensitive data inside the company network. Finally, monitoring network traffic while they are in transit can assist in preventing the entry and exit of sensitive data from the company network.

Figure 2.5 above provides a visual representation of these different stages.

Some DLP programs employ hybrid strategies that look at both content and context. Overall, companies may be better protected against insider threats by using an array of content- and context-based strategies, watermarking, and trap-based defenses.

2.5. Data Leak Prevention Techniques

1. Signature-based: A key method in Data Loss Prevention (DLP) is signature-based detection. By using common hashing algorithms, fingerprint databases are created. This method can find all secret material, making it simple to use and giving it good coverage. However, if the sensitive data is changed or updated, traditional hashing can be readily circumvented and false negatives can happen. Due to the substantial data indexing and matching that is necessary, processing large amounts of material can also be computationally expensive.

2. Regular expressions: Many DLPD systems employ regular expressions to carry out accurate and limited string matching. The use of wildcards in regular expression-based comparisons allows for the partial capture of updated data breaches. However, this approach offers poor protection for information and can produce high rates of false positives. As a consequence, it can only locate data breaches with predictable patterns.

3. Collection Insertion: A common technique for finding sensitive information in fragmented text input is collection intersection. This method can tolerate slight modifications to sensitive data, such as the addition of tags, character replacements, and minimal data reformatting. However, it incurs significant processing and storage costs. Basic n-gram detection may generate false alarms since the comparison is not in any particular sequence. Higher detection rates are achieved by alignment-based algorithms that can evaluate the order of n-grams at collection intersections.

4. Behavior analysis : It is crucial for deciphering user intent and solving insider attack issues. Although there are several behavior models and audit sources accessible in the literature, current behavior analysis-based methodologies are error-prone because of the temporal dynamics of context information, which results in high false positive and poor detection rates. The practical use of watermarking in DLPD is constrained by its susceptibility to malicious removal, distortion, and even change of the original data. Approaches using honeypots have inherent flaws since insiders might not use or engage with them.

2.6. Limitations of the Existing Models

A practical DLPD system should:

1. DLP systems should restrict sensitive data flows while allowing regular traffic.
2. Protect against malevolent or careless insiders causing data loss.
3. Stop data exfiltration if conventional security measures fail. Despite several studies, finding and preventing data breaches remains a challenging research issue.

Technique	Analysis	Pros	Cons
Fingerprinting	Content	Simple, Better coverage	Very sensitive to data modification
Regular expressions	Content	Simple, Tolerate certain noises	Limited data protection, High false positive
Collection intersection	Content	Wide data protection, Capture local features	High computation and storage cost, Inapplicable to evolved or obfuscated data
Machine learning	Content /Context	Resilient to data modifications, High accuracy	Large training data, Complicated
Behavior analysis	Context	Mitigate insider threats	Large training data, High false positives
Watermarking	Context	Forensics analysis	Vulnerable to malicious removal or distortion
Honeypots	Context	Detect malicious insiders	Limited applications

Figure 2.7 : Table depicting the different techniques, its pros and cons

Advanced content analysis techniques, such as machine learning-based approaches, have been suggested to address these problems. When analyzing vast volumes of data, however, these technologies run into scalability issues.

The benefits and drawbacks of several DLPD approaches used in scholarly research are shown in Figure 2.7.

2.7. Our Proposed Model

Our code implements a data leak detection system with user authentication and email functionality. The main menu allows the user to sign up or log in, and then access a submenu where they can create, open, and upload files. There is also an email submenu where users can send and receive emails. Additionally, there is a background task that checks if any of the files have been uploaded to the web and if so, sends an email warning the admin of a leak.

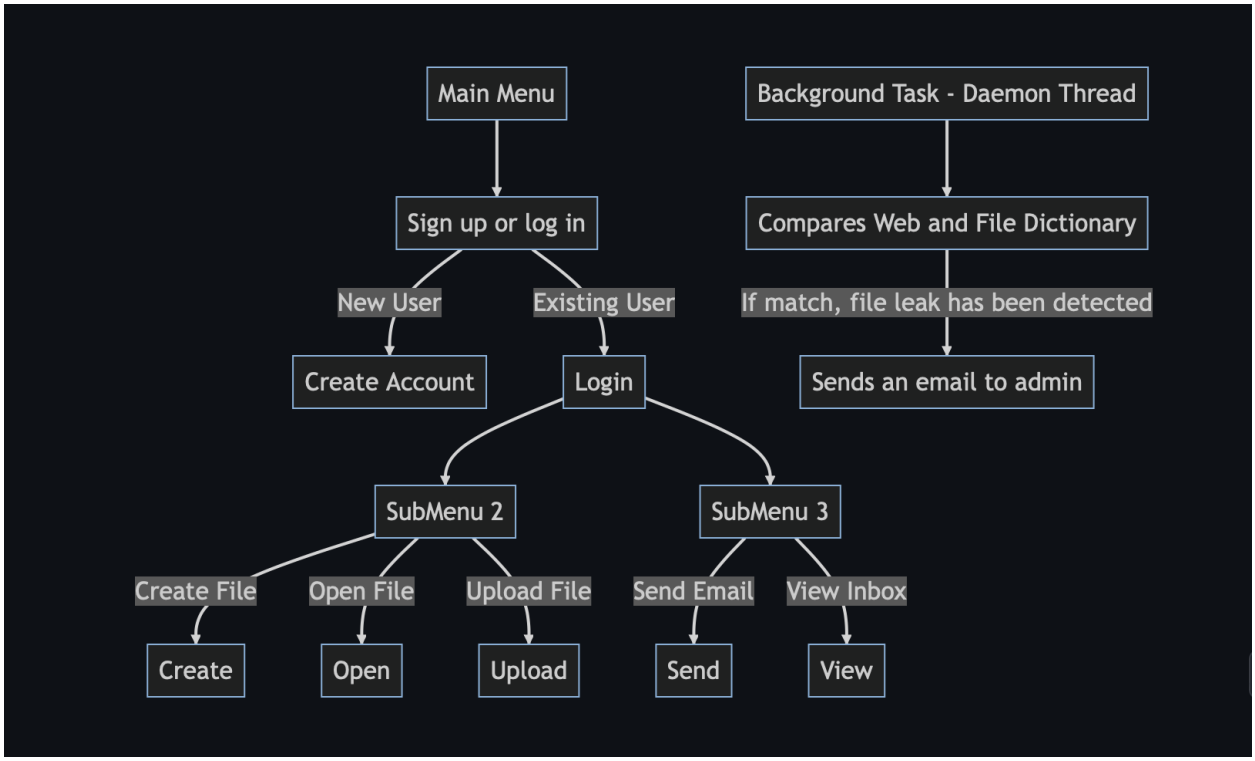


Figure 2.8 : Flowchart depicting the flow of command within the algorithm.

2.7.1. Function Definitions

The script starts by importing the necessary modules and defining a few global dictionaries to store user data, company files, and web files. It then defines functions to save and load these dictionaries to/from binary files using the pickle module. It also defines functions to save and load individual emails to/from binary files, and a function to view a user's emails.

```
45    web={}
46    web=load_web()
47    # Load the existing users from the binary file
48    users = load_users()
49    company=load_company()
50
51    def save_mail(usera,subject,data):
52
53        globals()[usera] = {}
54        a=globals()[usera]
55        a[subject]=data
56        print(type(a))
57        with open(usera,"wb") as f:
58            pickle.dump(a,f)
59
60    def load_mail(user):
61        a=globals()[user]
62        try:
63            with open(user,"rb") as f:
64                return pickle.load(f)
65        except FileNotFoundError:
66            return {}
67
68    def view_mail(user):
69        a=load_mail(user)
70        for key in a:
71            print(key)
72        subject=input('enter name of mail to open')
73        print(a[subject])
```

2.7.2. Main Menu

The main menu function prompts the user to sign in or login with an existing account. If the user signs in, they are prompted to enter a username and password, and their information is stored in the user dictionary. If the user logs in with an existing account, they are prompted to enter their username and password, and the script checks if the username exists and if the password matches the one associated with that username. If the login is successful, the user is taken to a submenu that allows them to manage their files.

1.py > main_menu

```python
#main menu
def main_menu():
    while True:
        print("Main Menu:")
        print("1. Sign-in")
        print("2. Login")
        print("3. Exit")
        choice = int(input("Enter your choice: "))
        if choice == 1:
            username = input("Enter your username: ")
            if username not in users:
                password = input("Enter your password: ")
                users[username] = password
                save_users()
                with open (username,"w") as f:
                    f.close()
                print("User created!")
            else:
                print('User already exists')
        elif choice == 2:
            username = input("Enter your username: ")
            if username in users:
                password = input("Enter your password: ")
                # If the username exists, check if the password matches
                if users[username] == password:
                    print("Login successful!")
                    submenu2()
                else:
                    print("Incorrect password")
            else:
                print('User does not exist')
        elif choice == 3:
            exit()
        else:
            print("Invalid choice.")
```

2.7.3. Sub Menu

The submenu allows the user to create new files, view and edit existing files, and manage their emails. If the user selects the ‘mail’ option, they are taken to a submenu that allows them to send and receive emails. If the user selects the ‘create’ option, they are prompted to enter a filename and data, and a new file with that name and data is added to the company dictionary. If the user selects the "open" option, they are prompted to enter the name of a file they want to open, and the script checks if the file exists in the company dictionary. If the file exists, the script displays its contents and prompts the user to upload it to the web. If the user chooses to upload the file, it is added to the web dictionary. If the user chooses not to upload the file, they are taken back to the list of files.

```python
#submenu2
def submenu2():
    while True:
        print("Submenu 2:")
        print("1. mail")
        print("2. create")
        print("3. open")
        print("4. Back to previuos menu")

        choice = int(input("Enter your choice: "))

        if choice == 1:
            submenu3()
        elif choice == 2:
            print("create")
            filename = input("Enter your filename: ")
            if filename not in company:
                data = input("Enter your data: ")
                company[filename] = data
                save_company()
                print("File created!")
            else:
                print('Filename already exists')
        elif choice == 3:
            print('open')
            for key in company.keys():
                print(key)
            filename=input('enter filename to open')
            print(company[filename])
            reply=input('Do you want to upload it to the web, reply with yes or no')
            if reply=='yes':
                web[filename]=company[filename]
                upload()
                sleep(1)
            elif reply=='no':
                break
                #exit to list of files
            else:
                print('you have entered incorrect option')
        elif choice==4:
            main_menu()
        else:
            print("Invalid choice.")

```

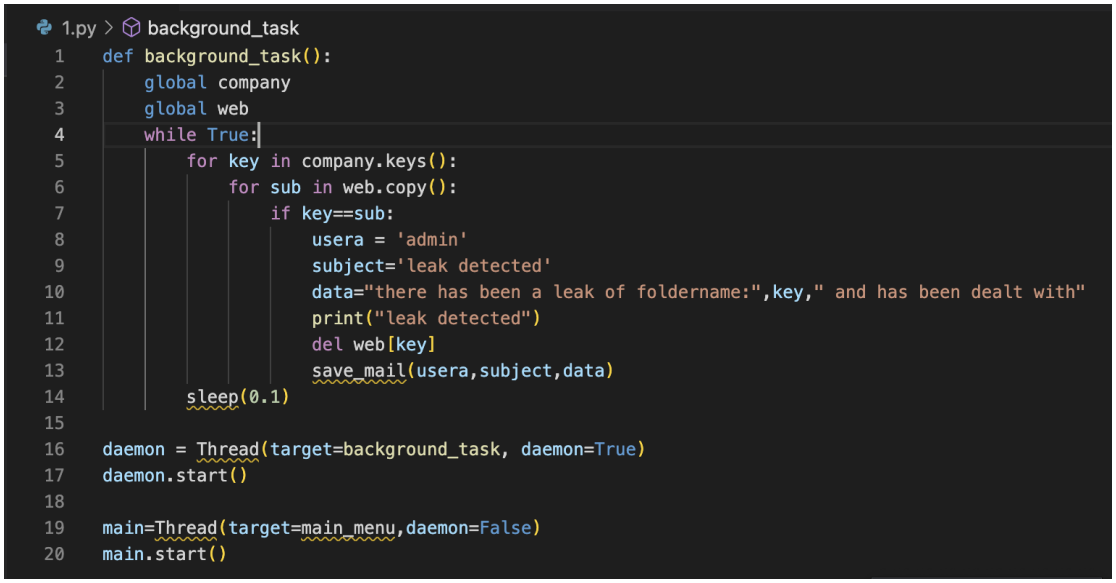
1.py > submenu3

```python
#submenu3
def submenu3():
    while True:
        print("Submenu 3:")
        print("1. send mail")
        print("2. recieve mail")
        print("3. quit")

        choice=int(input("Enter your choice: "))
        if choice==1:
            print('send mail')
            print('select the user you want to send the mail to')
            for keys in users:
                print(keys)
            usera = input("Enter a username ")
            subject=input('enter the subject')
            data=input("enter your data ")
            save_mail(usera,subject,data)
        elif choice==2:
            print('recieve mail')
            user=input('enter your username')
            view_mail(user)
        elif choice==3:
            break
        else:
            print("Invalid choice")
```

2.7.4. Daemon Thread

The daemon thread runs indefinitely in the background, checking for any file leaks between the company and the web. If a file exists in both dictionaries, the script sends a notification email to the admin and removes the file from the web dictionary. The task runs every 0.1 seconds.



2.7.5. Outcome

- save_users(): saves the user dictionary to a binary file
- load_users(): loads the user dictionary from the binary file
- save_company(): saves the company dictionary to a binary file
- load_company(): loads the company dictionary from the binary file
- upload(filename): uploads a file to the web
- load_web(): loads the web dictionary from the binary file
- save_mail(user, subject, data): saves an email to a binary file, with the recipient and subject as the keys and the message body as the value
- load_mail(user): loads all emails for a user from the binary file
- view_mail(user): prints the subjects of all emails for a user and allows the user to select one to view the message body

- main_menu(): the main menu, where the user can sign up or log in
- submenu2(): allows the user to create, open, and upload files
- submenu3(): allows the user to send and receive emails

Lastly, there is a background task running on a separate daemon thread that periodically checks if any files have been uploaded to the web. It does this by comparing the existing file dictionary and web dictionary and sending an email warning the administrator if a match is found.

Our proposed model is truly unique in its ability to integrate multiple functionalities within a single solution. By seamlessly combining a powerful file management system with a secure portal for leak detection, our algorithm offers a comprehensive solution that caters to a range of user needs. What's more, we have gone the extra mile by integrating email functionality that promptly notifies owners of any potential leaks, thereby significantly reducing the risk of data loss or breaches. The integration of multiple functionalities within our solution offers an unparalleled user experience, improving efficiency while enhancing security. We are confident that our model's unique combination of features will transform the way organizations manage their data, ensuring optimal protection of their sensitive information.

CHAPTER 3: CONCLUSION

Data leakage is a major security concern that can have serious consequences for both organizations and individuals. Sensitive information, such as PII, financial data, intellectual property, and medical information, can be leaked through various modes of transmission, such as email, instant messaging, file transfer, and social media. It is crucial for organizations to implement appropriate security measures and educate themselves on the best practices to prevent data leaks and protect sensitive data.

Data leaks can be caused by malicious outsiders or insiders, with the healthcare and business industries being the most affected. To prevent data leakage, organizations should adopt appropriate technical and administrative measures, such as access control mechanisms, network segmentation, system hardening, and user security awareness. DLPD tools such as encryption, access control, firewalls, IDS, antiviruses, content-based methods, context-based approaches, and machine learning-based approaches can help to detect and prevent data leaks.

A combination of DLP techniques, including signature-based detection, regular expression-based matching, collection intersection, and behavior analysis, can provide a more comprehensive and effective approach to data leak prevention. However, organizations should carefully consider which techniques to implement based on their needs, as each technique has its advantages and limitations. Machine learning-based methods have been proposed to overcome some limitations, but scalability remains a challenge. Detecting and preventing data leaks is an ongoing challenge that requires continued research and innovation.

To address the challenge of data leakage, a proposed data leak detection system incorporates user authentication, file management, and email functionality to meet the needs of multiple users. The algorithm utilizes functions such as saving and loading data dictionaries, creating and uploading files, and sending and receiving emails to ensure the security of sensitive information. A background task periodically checks for any file leaks and sends an email to the admin in case of a breach. By integrating multiple functionalities within a single solution, the proposed model can provide a more efficient and secure experience for users.

Organizations must adopt comprehensive security measures and training programs to prevent and mitigate data leakage threats. These measures include access control mechanisms, network segmentation, system hardening, and detection and prevention tools such as firewalls and intrusion prevention systems. User security awareness in the workplace is crucial to prevent inadvertent data leakage. By adopting these measures, organizations can reduce the likelihood of data leaks and protect sensitive information.

In conclusion, data leakage is a significant threat to organizations and individuals alike, leading to potential harm such as financial loss, reputational damage, and legal liability. Preventing and detecting data leaks requires a combination of technical and administrative measures, including the use of DLPD techniques, user security awareness, and comprehensive security measures. Continued research and innovation are necessary to address the evolving nature of data leakage threats and to develop more effective solutions to safeguard sensitive information. It is essential for organizations to prioritize data security and take proactive steps to protect sensitive information from unauthorized access, transmission, and use. By doing so, organizations can minimize the risk of data leakage and enhance their overall security posture.

CHAPTER 4: RECOMMENDATIONS

As concluded, the existing DLD models come with a number of drawbacks. It is essential to keep enhancing our file leak detection models and integrating the above solutions is the one step toward ensuring the same.

With the use of the daemon thread running in the background, we are able to compare the existing file dictionary and the web dictionary for any match. Once a match has been identified and a data leak has been detected, an email is sent to the admin warning them. This ‘comparison’ of dictionaries occurs every 0.1 seconds. When compared to the traditional method of DLD, this is far more beneficial, reduces the risk of computational errors, and is more efficient and accurate, leading to quicker decision making.

REFERENCES

[1] K. Wagh, “A Survey: Data Leakage Detection Techniques,” *International Journal of Electrical and Computer Engineering*, Aug. 2018, doi: 10.11591/ijece.v8i4.pp2247-2253.

[2] D. Gupta and U. Chellapandy, “Research Paper on Detection and Prevention of Data Leakage,” *International Journal for Research in Applied Science and Engineering Technology*, vol. 10, no. 4, pp. 2171–2176, Apr. 2022, doi: 10.22214/ijraset.2022.41731.

[3] J. Kim, J.-S. Kim, and H. Chang, “Research on Behavior-Based Data Leakage Incidents for the Sustainable Growth of an Organization,” *Sustainability,* vol. 12, no. 15, p. 6217, Aug. 2020, doi: 10.3390/su12156217.

[4] D. Liu *et al., Research on Leakage Prevention Technology of Sensitive Data based on Artificial Intelligence*. 2020. doi: 10.1109/iceiec49280.2020.9152286.

[5] A. Shabtai, Y. Elovici, and L. Rokach, *A Survey of Data Leakage Detection and Prevention Solutions.* Springer Science & Business Media, 2012.

[6] L. Cheng, F. Liu, and D. Yao, “Enterprise data breach: causes, challenges, prevention, and future directions,” *Wiley Interdisciplinary Reviews-Data Mining and Knowledge Discovery*, vol. 7, no. 5, p. e1211, Sep. 2017, doi: 10.1002/widm.1211.

[7] I. H. Montano, J. L. Aranda, J. C. Rodríguez, S. Cardín, I. De La Torre Díez, and J. J. P. C. Rodrigues, “Survey of Techniques on Data Leakage Protection and Methods to address the Insider threat,” *Cluster Computing*, vol. 25, no. 6, pp. 4289–4302, Jul. 2022, doi: 10.1007/s10586-022-03668-2.

[8] Y. Ji, A. Sun, J. Zhang, and C. Li, “A Critical Study on Data Leakage in Recommender System Offline Evaluation,” *ACM Transactions on Information Systems*, vol. 41, no. 3, pp. 1–27, Oct. 2020, doi: 10.1145/3569930.

[9] V. Sundareswaran, “STUDY OF CYBERSECURITY IN DATA BREACHING,” *ResearchGate*, Mar. 2018, [Online]. Available: https://www.researchgate.net/publication/325300571_STUDY_OF_CYBERSECURITY_IN_DATA_BREACHING

[10] H. Hammouchi, O. Cherqi, G. Mezzour, M. Ghogho, and M. E. Koutbi, “Digging Deeper into Data Breaches: An Exploratory Data Analysis of Hacking Breaches Over

Time," *Procedia Computer Science*, vol. 151, pp. 1004–1009, Jan. 2019, doi: 10.1016/j.procs.2019.04.141.

[11] A. H. Juma'h and Y. Alnsour, "The effect of data breaches on company performance," *International Journal of Accounting and Information Management*, vol. 28, no. 2, pp. 275–301, Mar. 2020, doi: 10.1108/ijaim-01-2019-0006.