ABSTRACT

To reduce the leaking of sensitive and important data in a cloud-based computing environment, an improved data leakage detection system is designed. In general, data leakage in computing systems has devastated or caused great irreversible harm to many institutions or organizations around the world. Therefore, the goal of this research is to use a dynamic password or key for data decryption security measures to identify and stop any purposeful or unintentional data leakages. The OOADM methodology was used to accomplish this. The new system's backend was created using Microsoft SQL Server Management Studio and ASP.net MVC. Additionally, the new system keeps track of events inside and outside the computing environment using a date and time stamp by integrating an audit trail/transaction log mechanism.

CHAPTER 1

INTRODUCTION

A data leak occurs when confidential information is improperly transferred from a computer or datacenter to an unauthorized third party. Simple mental tricks, physical removal of discs or reports, or covert methods like data concealment can all be used to leak data. The quest to prevent intrusions, viruses, or spam has given way to the fight against data leakage. Data leaks can happen accidentally or on purpose. Intellectual property, trade secrets, or personally identifiable information may be exposed. The number one hazard for many businesses today is the disclosure of sensitive information. When private corporate data, such as patient or customer information, ASCII text files or design specifications, tariffs, trade secrets, and spreadsheet predictions and budgets, leaks, it is considered a data leak. When these are disclosed, the business is no longer protected and is no longer under the organization's control. The business community is exposed as a result of this unchecked data leaking. The company faces significant danger once this data is no longer within the domain. Data breaches today can affect hundreds of thousands or even millions of individual customers and much more individual information, all as a result of a single attack on a single business. To offer protection against the potential of data leakage, many prior researchers have implemented a variety of enabling security technologies, including firewalls, encryption techniques, access control techniques, identity management, and context-based/machine learning detectors. To ensure that data does not leak out, some researchers created a three-tier application that uses watermarking. With this technique, the organization embeds a different code in each copy of the data. In case a distributed copy of the confidential data is later discovered in an unapproved area, the guilty agent can be caught relatively quickly. Data leakage continues to be a major concern, especially for many companies and institutions, despite the efforts of past researchers to stop it.

CHAPTER 2

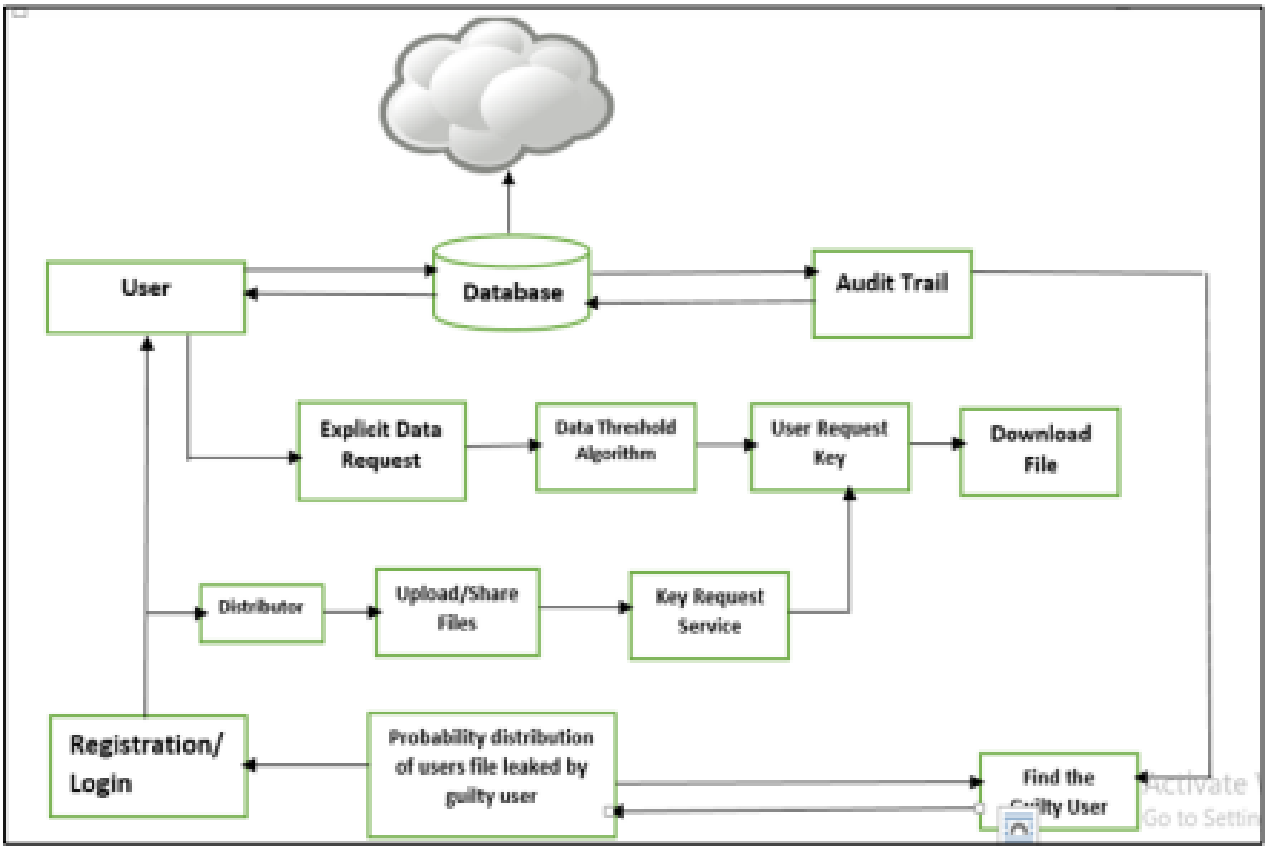
LITERATURE SURVEY

Data leakage didn't start when businesses started storing their secured data online. Data leaks have actually existed for as long as people and businesses have kept records and stored sensitive information. Before computers were widely used, a data breach might be as simple as looking at someone's medical records. Without official consent or looking at private records that weren't properly destroyed. However, the 1980s saw a sharp increase in the number of publicly revealed data leaks. The public's awareness of the possibility of data leaks also started to increase in the 1990s and the early 2000s. The majority of data leaking material focuses on the time period from 2005 to the present. This is largely because of the advancement of technology and rapid increase of electronic data throughout the world Data leaks have become a major problem for both businesses and consumers due to the rapid expansion of electronic data globally and the improvement of technology. Intellectual Property (IP), financial information, patient information, personal credit card data, and other sensitive data are examples of data that businesses and organizations must protect. Data leakages have gained widespread attention as businesses of all sizes become increasingly reliant on digital data, cloud computing, and workforce mobility. Data leakage also is a situation whereby crucial information is illegally transferred to the outside world. Traditionally, leakage detection is handled by watermarking, e.g., a singular code is embedded in each distributed copy. In case a replicate is found later within the hands of a third or an unauthorized party, the leaker can be identified. Watermarks are often very useful in some cases, but again, involve some modification of the first data. Furthermore, watermarks can sometimes be destroyed if the data recipient is malicious. E.g. A hospital may give patient records to researchers who will devise new treatments.

CHAPTER 3

PROPOSED MODEL

3.1 ARCHITECTURE



3.2 EXPLANATION OF THE PROPOSED MODEL

The methodology adopted for this research work is the Object-Oriented Analysis and Design methodology (OOAD). The existing system from the perspective of objects and similar objects are grouped as classes and their characteristics are handled as properties while their behaviors are treated as the actions or methods within the same bundle of object. This methodology was chosen because it is best used to handle a system where different objects exist. It is a structured approach that is used in analyzing and designing a system. It applies object-oriented concepts and develop a set of graphical system model during the development lifecycle of the software. The New application system has in it an audit trail/transaction log system which checks also monitors user transactions and activities to computing resource at each stage of the process. The system administrator registers each user and assigns roles to them. The roles include an administrator, a distributor and a user. The distributor uploads new files into the system, approves, sends key request to the user, also shares file to each user.

CHAPTER 4

RESULTS

Input Interface:

In this system, data is supplied by the user during registration which in turn is used to create login details and the secret key of the user. The interface makes use of Graphical User Interface components such as radio buttons, text boxes to accept input from the users into the system.

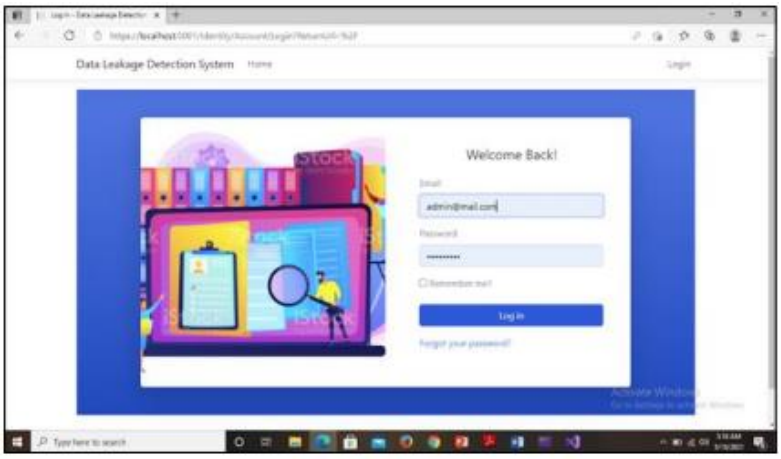


Figure 2 Admin sign in Module

Figure-2: Shows the module where the admin logs in to the system in order to create new users and assign role to the users.

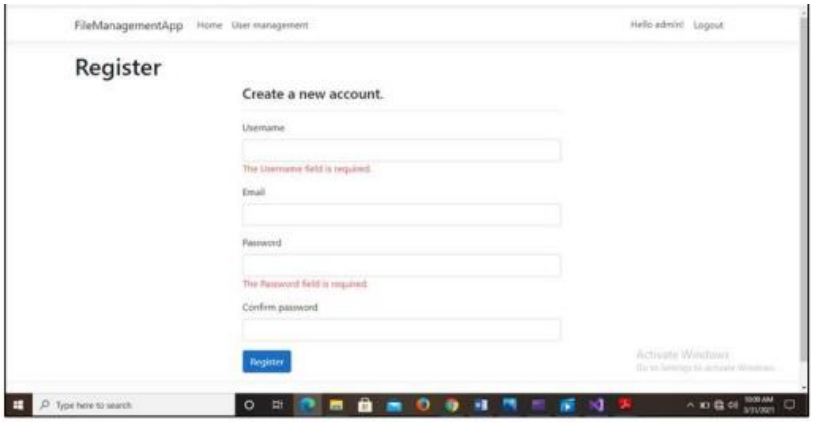


Figure 3 Admin registers a new user

Figure-3: Illustrates where the administrator creates a new user. The “Register” button is clicked after entering the fields.

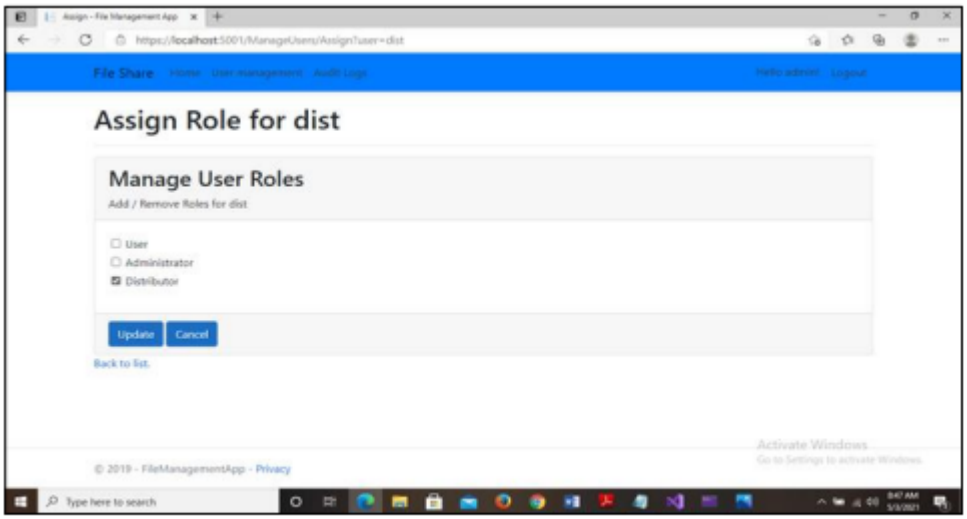


Figure 4 Admin assigns roles

Figure 4: Shows where the administrator assigns role to the users, one user can be a distributor of files, or another administrator or only a user.

Output Interface/ Test Results:

Firstly, the administrator logs into the system and has the privileges of registering users, viewing the list of all registered users, assign roles to members, and checking the audit trail to identify a leaker.

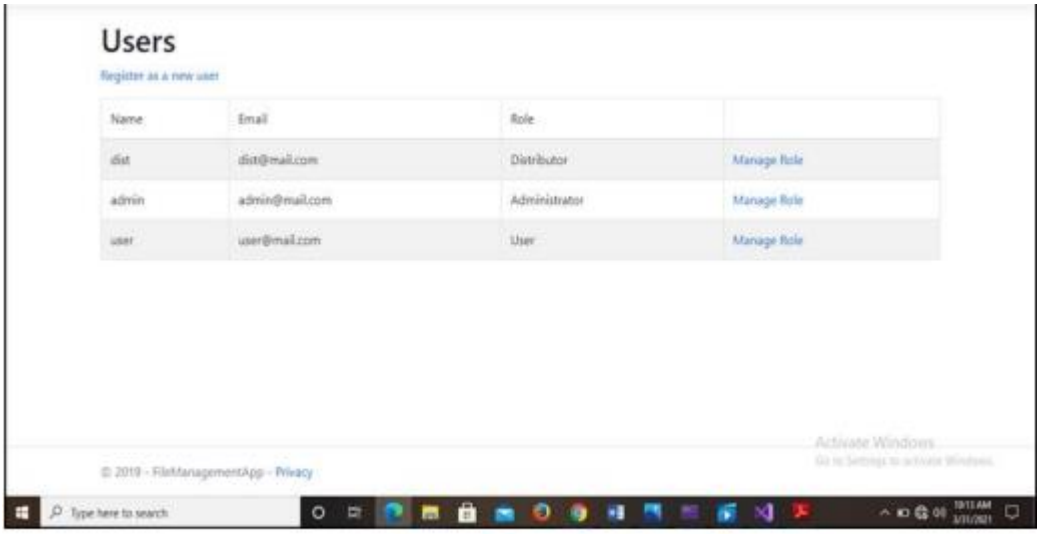


Figure 5 Admin dashboard

Figure-5: The admin views the set of all users that have registered and also can assign roles to them.

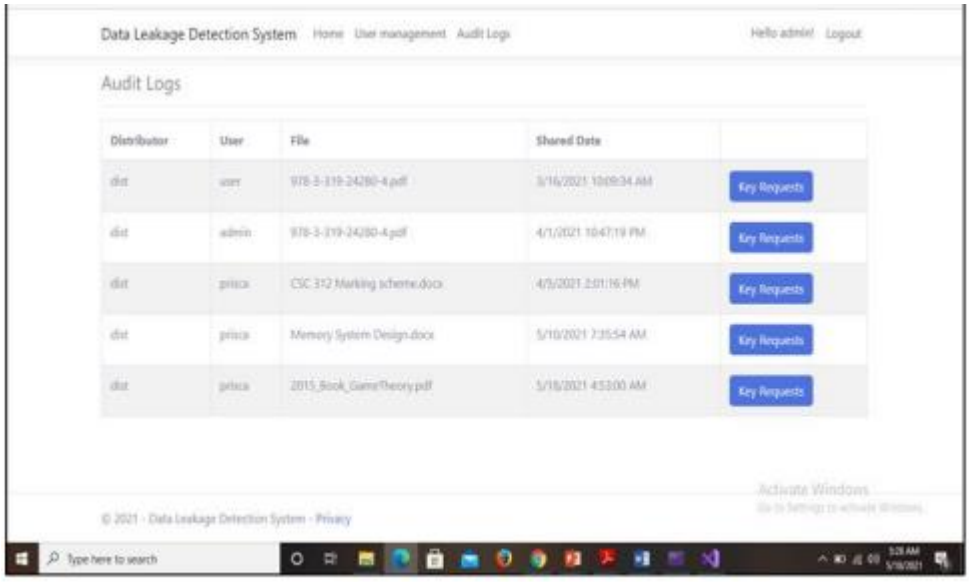


Figure 6 Audit Trail/Transaction log module

Figure-6: Shows the audit log or trail that tracks user activities in the system.

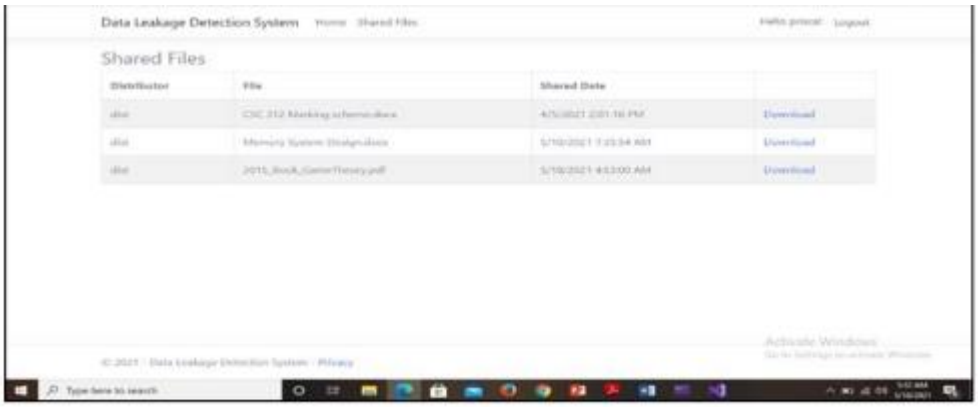


Figure 7 Shared file module

Figure-7: Shows the files shared to a particular user and can be downloaded through the approval of key request by the distributor or administrator.

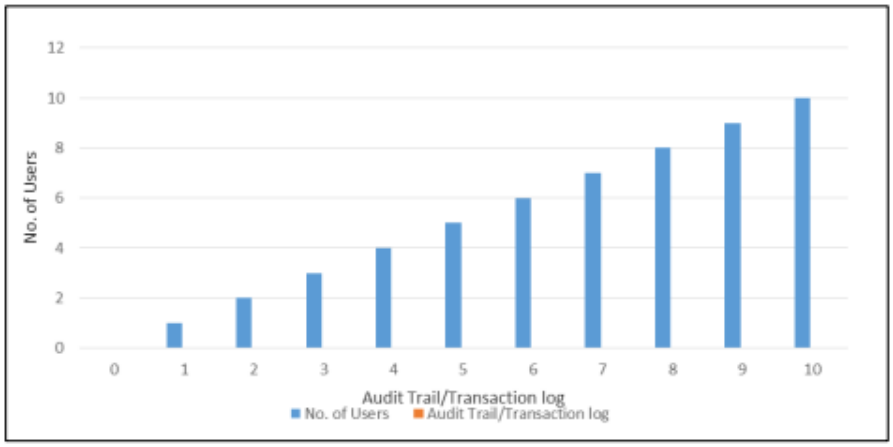


Figure 8 Graph of Entries without Transaction Log/Audit Trail

Figure-8:Depict graphically, 10 clients entered in to the without transaction log/Audit trail system. The graph shows that for any member of clients entered there is no any mechanism to profile user activity in the system.

No.of users	0	1	2	3	4	5	6	7	8	9	10
No,of Audit trail	0	0	0	0	0	0	0	0	0	0	0

Table of Entries without Transaction Log/Audit Trail

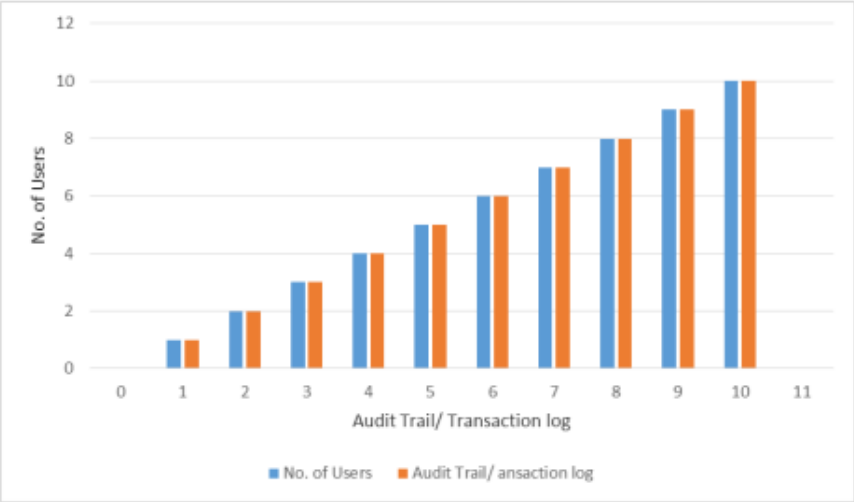


Figure 9 Graph of Entries with Transaction Log/ Audit Trail

Figure-9: Shows that for each client entered there is a corresponding transaction log/Audit Trail System that profile user activity in the system.

CHAPTER 5

CONCLUSION

Data security is not an easy task even top leading organizations suffer the fear of data leakage. This research work discussed some of the latest work carried out in the area of detection of data leakage and prevention algorithms, tools and technologies supported. And the researchers were able to develop a system that provides security to overcome issues related to cloud environment. This helps to protect the data from leakage by tokenization of files before distributing and setting the time bound for each and every file that particular user needs to download from the cloud storage. The system protects the data leaked from guilty agent who act as a third party and security is provided using dynamic key generation which is an auto generated random unique number for every file when user or an employee makes attempt to view the content of file. Incorporated in it also is an Audit Trail/ Transaction log which profiles user activities in the system as against the old system that does not have an Audit trail. The audit trail will monitor when a user sends out organization’s information with date and time stamp.

REFERENCES

1) What is Data Leakage? Defined, Explained, and Explored | Forcepoint

2) Data Leakage in Machine Learning - MachineLearningMastery.com

3) Top 7 Data Loss Prevention Tools for 2022 (techtarget.com)

4) 16 Best Data Loss Prevention Software Tools 2022 (Free + Paid) (comparitech.com)

5) What is Data Loss Prevention (DLP)? | Digital Guardian