Department of Computer Science

Kanpur-208024

2021-22

Mini project

Under the guidance of

Sri Nath Dwivedi

Head of Department Computer Science &Engineering

DATA LEAKAGE DETECTION
[DLD]

Submitted by:

Ashutosh Singh-	1901660100019
Umang Raj Chaurasia-	1901660100063
Swapnil Siddhartha-	1901660100058
Dinisha Jaiswal-	1901660100024

5th Semester

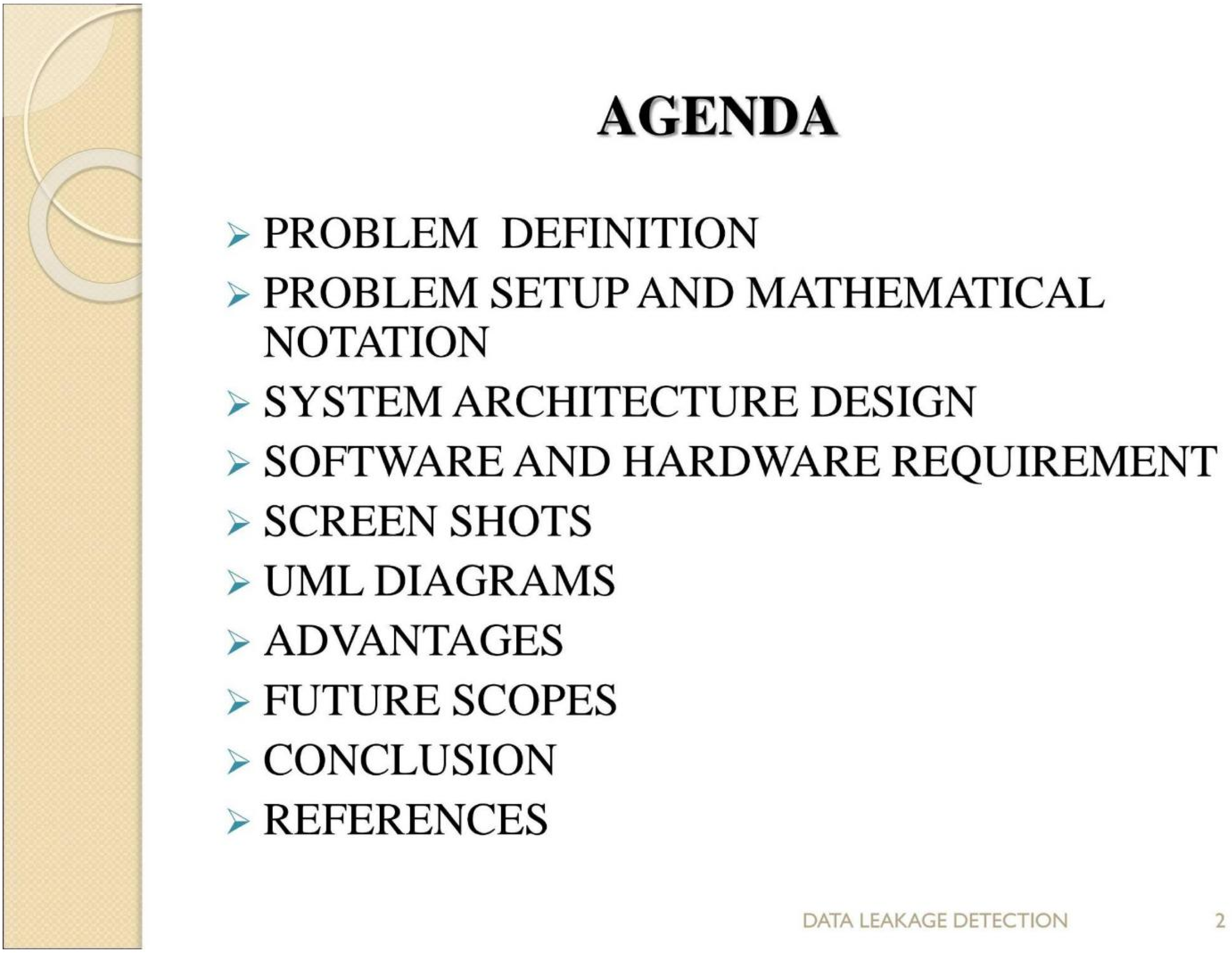
Dr. AITH, Kanpur

Sign:________

Date:__/__/__

Candidate’s Declaration

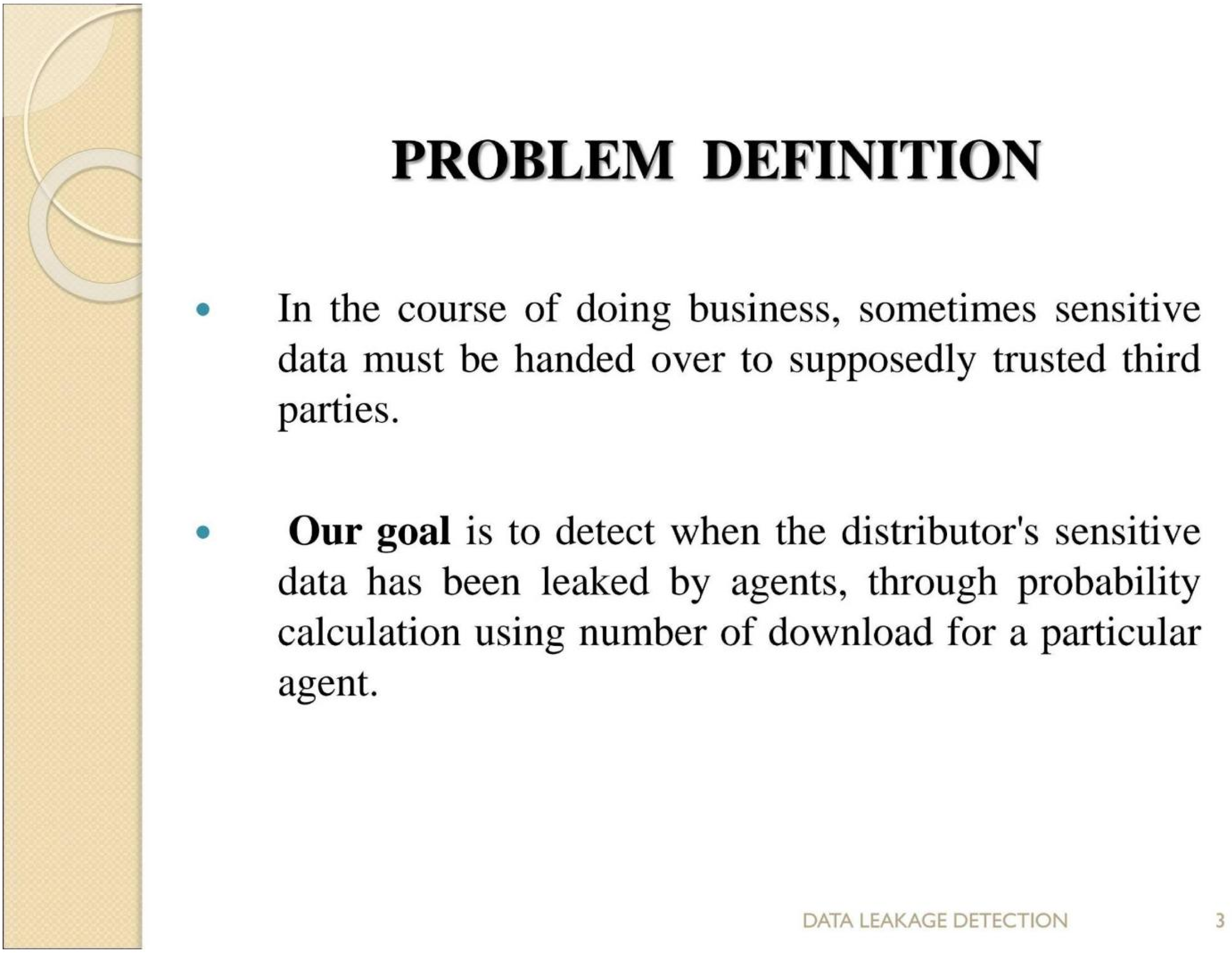
I hereby declare that the mini project work being presented in this report entitled ”DATA LEAKAGE DETECTION “ submitted in the department of computer Science ”Dr. AITH ,Kanpur “ is the authentic work carried out by be us under the guidance of Shri Nath Dwivedi Sir ,Head of Department Computer Science Engineering ,Dr. AITH , Kanpur



AGENDA

- PROBLEM DEFINITION
- PROBLEM SETUP AND MATHEMATICAL NOTATION
- SYSTEM ARCHITECTURE DESIGN
- SOFTWARE AND HARDWARE REQUIREMENT
- SCREEN SHOTS
- UML DIAGRAMS
- ADVANTAGES
- FUTURE SCOPES
- CONCLUSION
- REFERENCES

Scanned by TapScanner

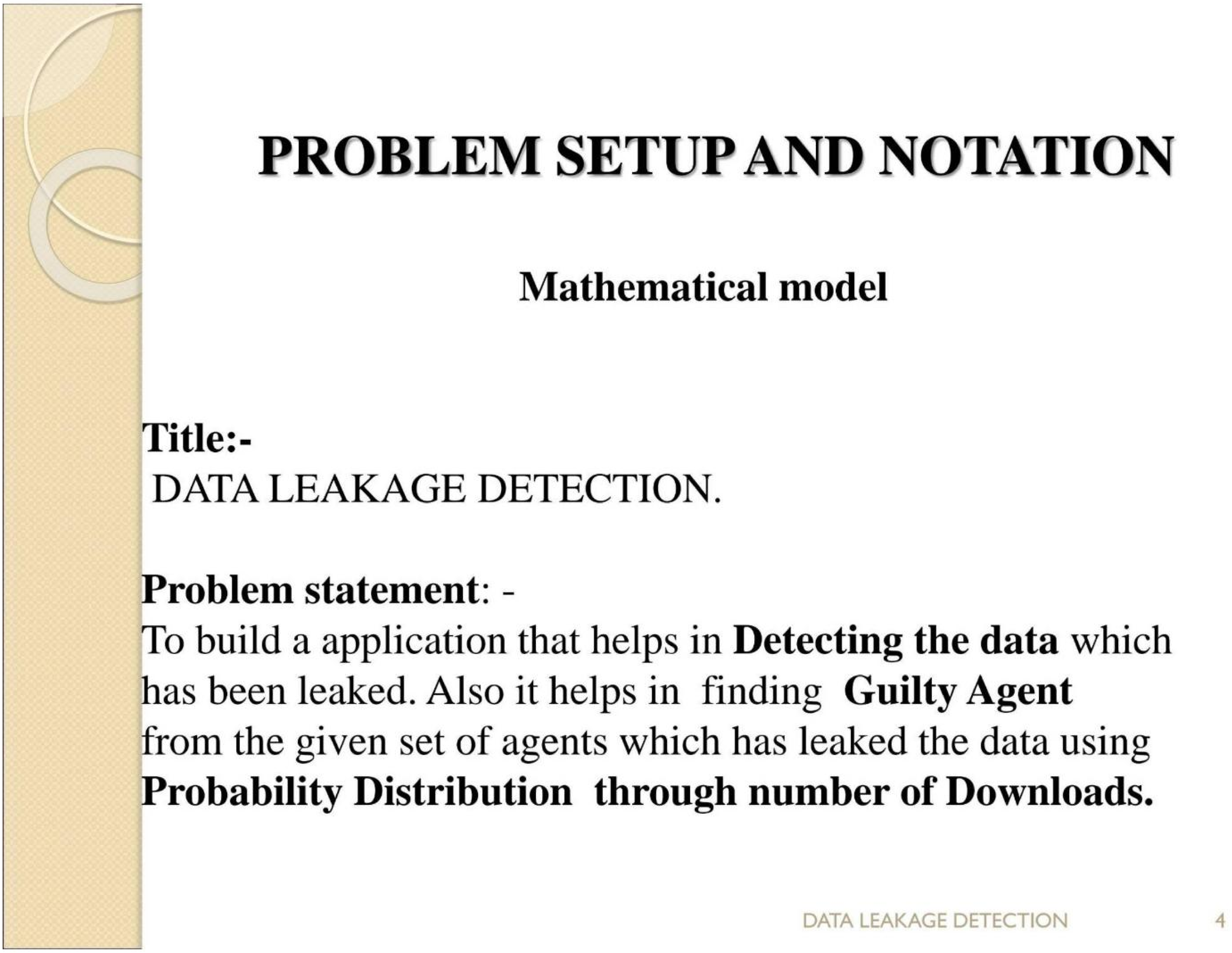


PROBLEM DEFINITION

- In the course of doing business, sometimes sensitive data must be handed over to supposedly trusted third parties.

- **Our goal** is to detect when the distributor's sensitive data has been leaked by agents, through probability calculation using number of download for a particular agent.

Scanned by TapScanner

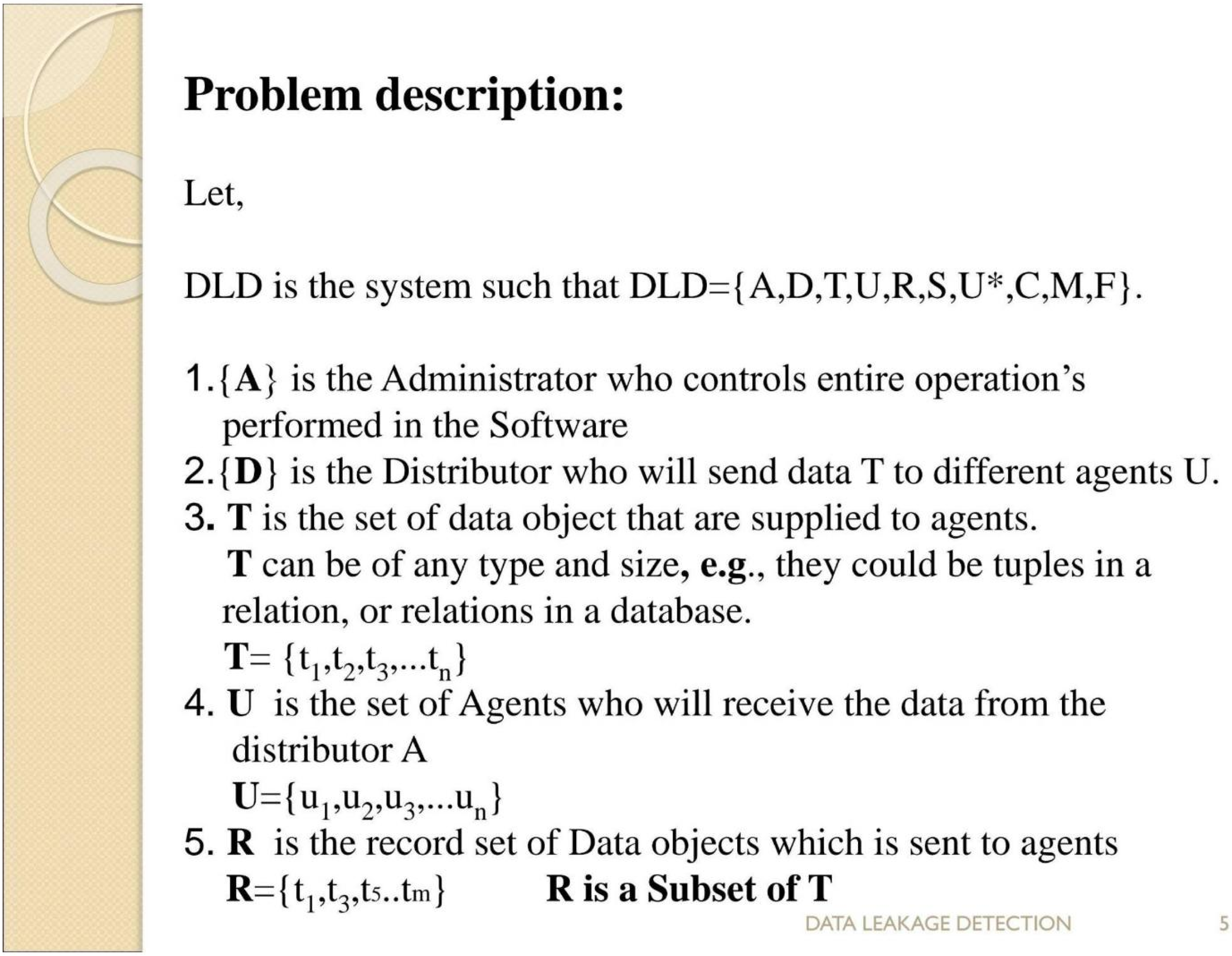


PROBLEM SETUP AND NOTATION

Mathematical model

Title:-
DATA LEAKAGE DETECTION.

Problem statement: -
To build a application that helps in **Detecting the data** which has been leaked. Also it helps in finding **Guilty Agent** from the given set of agents which has leaked the data using **Probability Distribution through number of Downloads.**



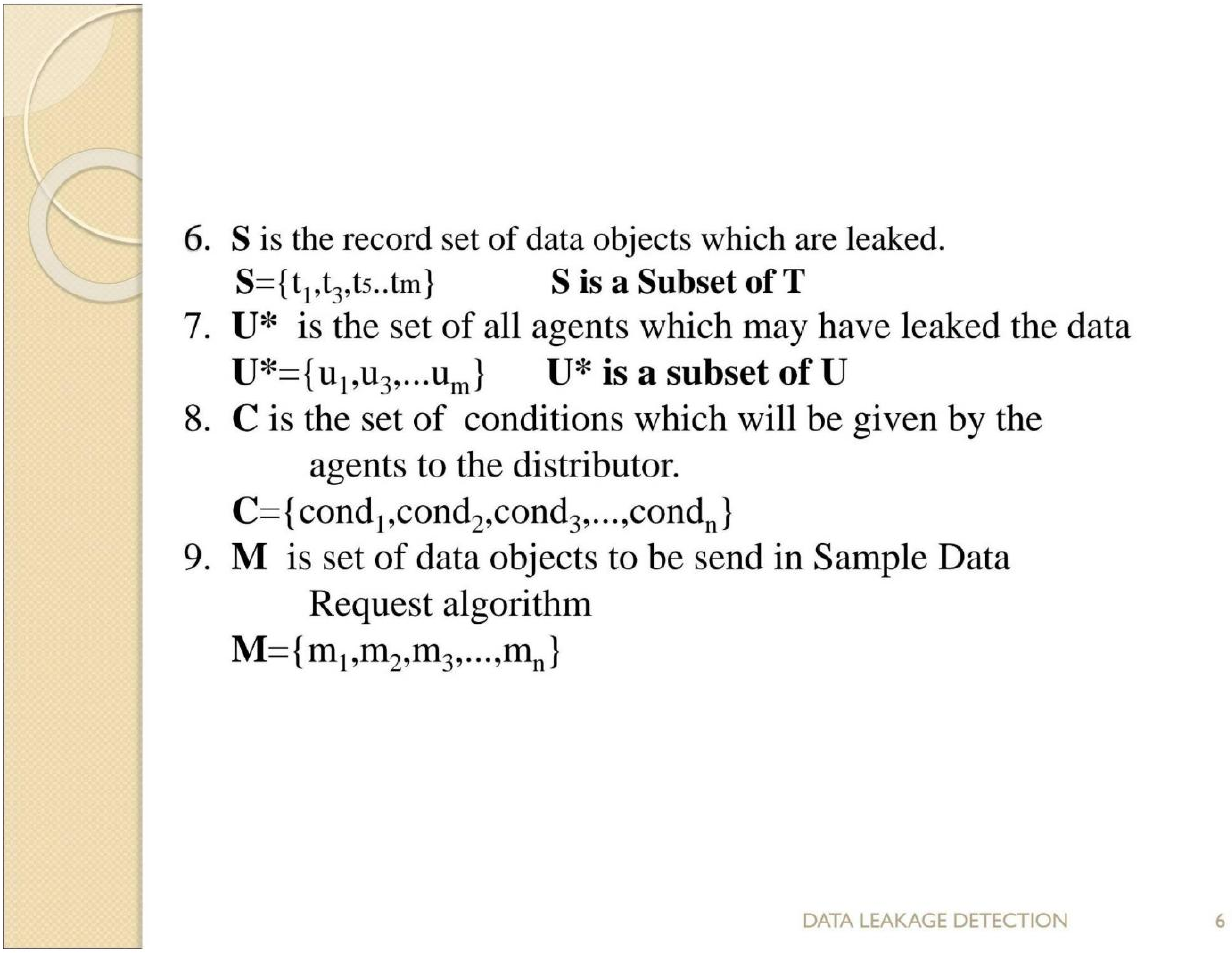
Problem description:

Let,

DLD is the system such that DLD={A,D,T,U,R,S,U*,C,M,F}.

1. {**A**} is the Administrator who controls entire operation's performed in the Software
2. {**D**} is the Distributor who will send data T to different agents U.
3. **T** is the set of data object that are supplied to agents.
   T can be of any type and size**, e.g.**, they could be tuples in a relation, or relations in a database.
   T= $\{t_1,t_2,t_3,...t_n\}$
4. **U** is the set of Agents who will receive the data from the distributor A
   U=$\{u_1,u_2,u_3,...u_n\}$
5. **R** is the record set of Data objects which is sent to agents
   R=$\{t_1,t_3,t_5..t_m\}$ **R is a Subset of T**

Scanned by TapScanner



6. **S** is the record set of data objects which are leaked.

   S={t_1,t_3,t5..tm} **S is a Subset of T**

7. **U*** is the set of all agents which may have leaked the data

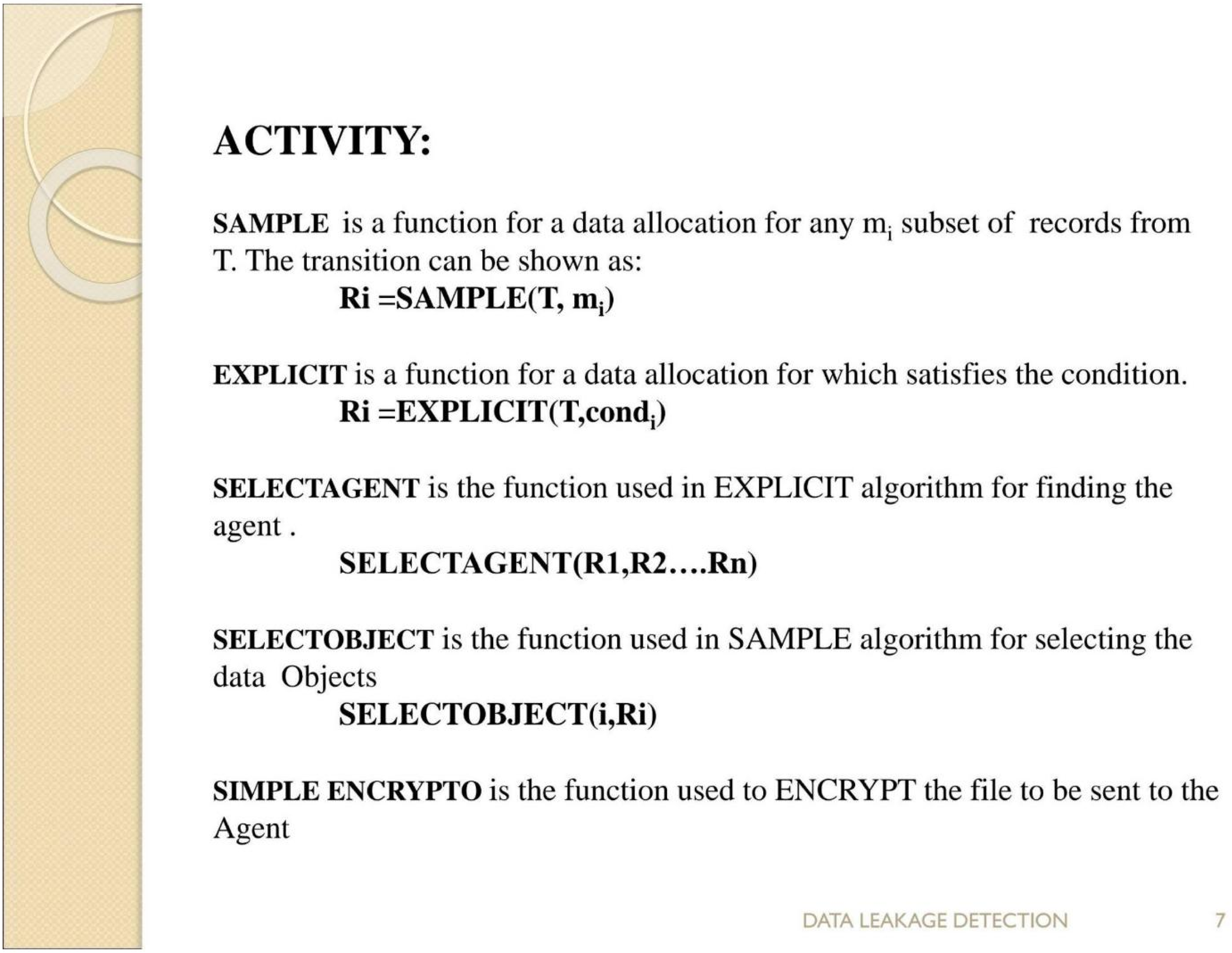
   U*={u_1,u_3,...u_m} **U* is a subset of U**

8. **C** is the set of conditions which will be given by the agents to the distributor.

   C={$cond_1$,$cond_2$,$cond_3$,...,$cond_n$}

9. **M** is set of data objects to be send in Sample Data Request algorithm

   M={m_1,m_2,m_3,...,m_n}



ACTIVITY:

SAMPLE is a function for a data allocation for any m_i subset of records from T. The transition can be shown as:

$$\mathbf{Ri = SAMPLE(T, m_i)}$$

EXPLICIT is a function for a data allocation for which satisfies the condition.

$$\mathbf{Ri = EXPLICIT(T, cond_i)}$$

SELECTAGENT is the function used in EXPLICIT algorithm for finding the agent .

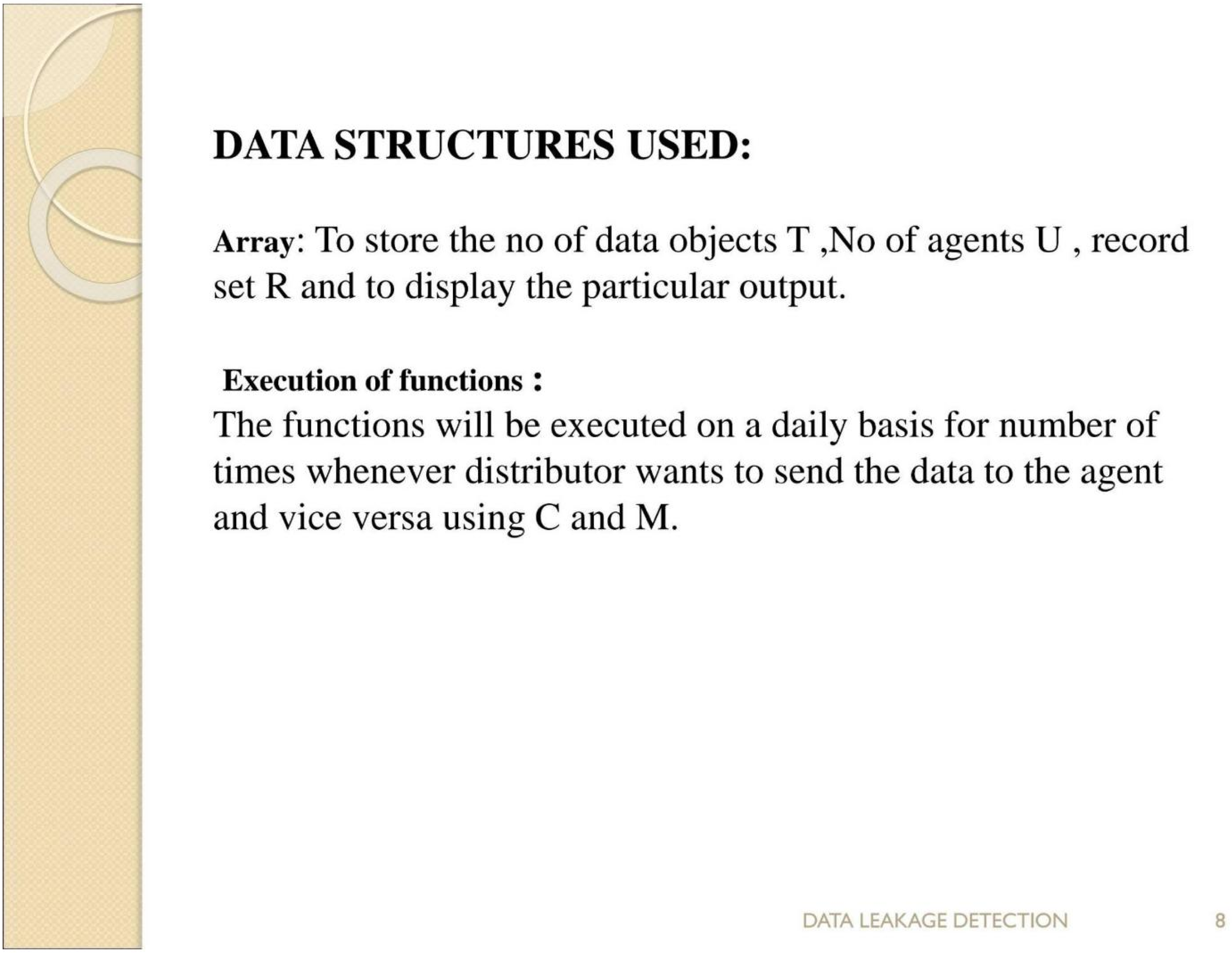
SELECTAGENT(R1,R2….Rn)

SELECTOBJECT is the function used in SAMPLE algorithm for selecting the data Objects

SELECTOBJECT(i,Ri)

SIMPLE ENCRYPTO is the function used to ENCRYPT the file to be sent to the Agent

Scanned by TapScanner



DATA STRUCTURES USED:

Array: To store the no of data objects T ,No of agents U , record set R and to display the particular output.

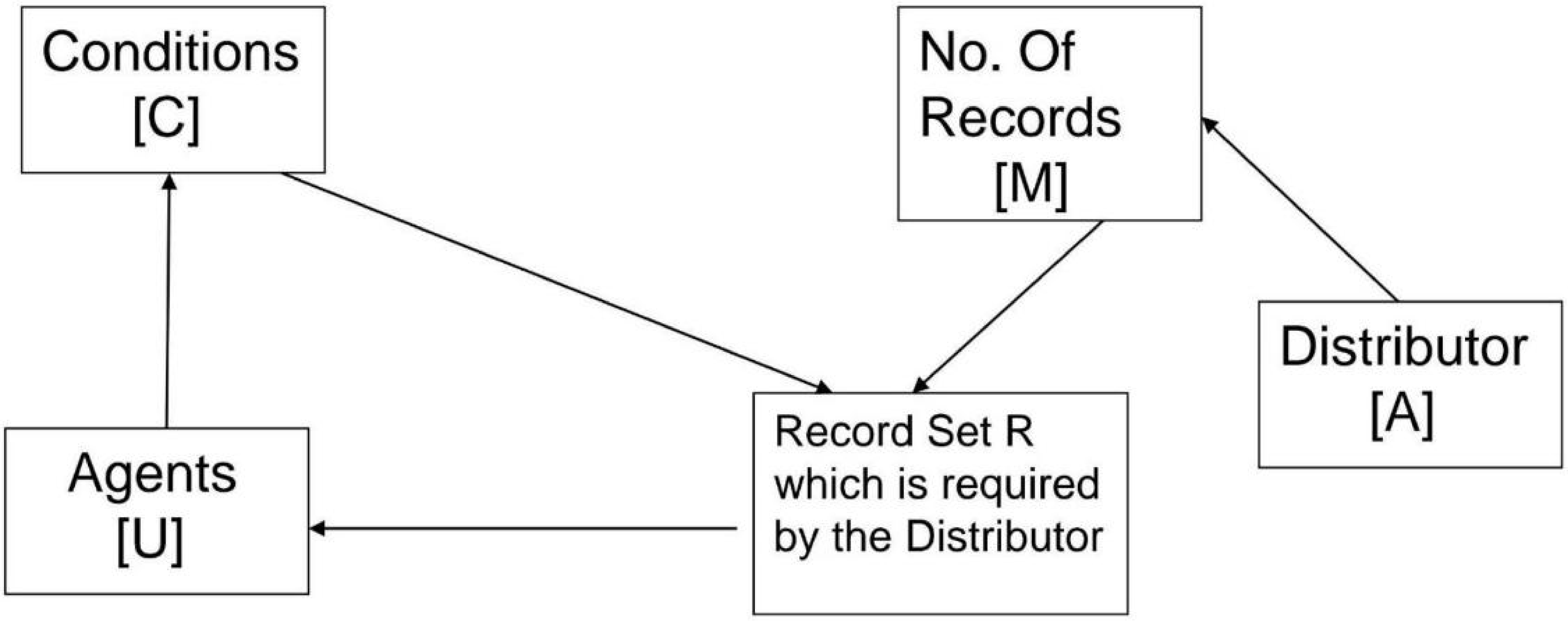
Execution of functions :

The functions will be executed on a daily basis for number of times whenever distributor wants to send the data to the agent and vice versa using C and M.

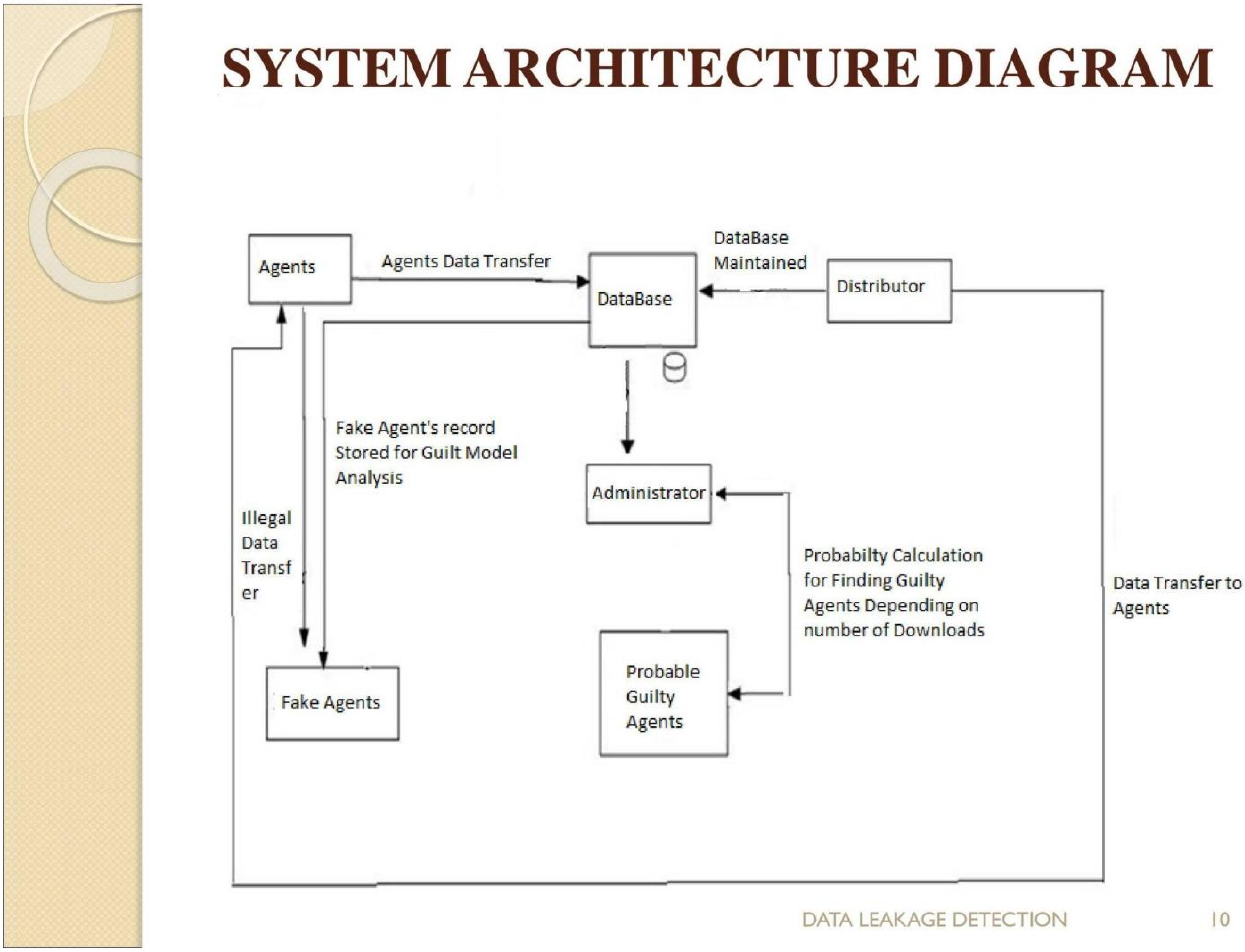
Scanned by TapScanner

FUNCTIONAL DEPENDENCY DIAGRAM:

The functional dependency of the system depends upon the conditions which are given by the agent and no of records which distributor decides to send to the agents.

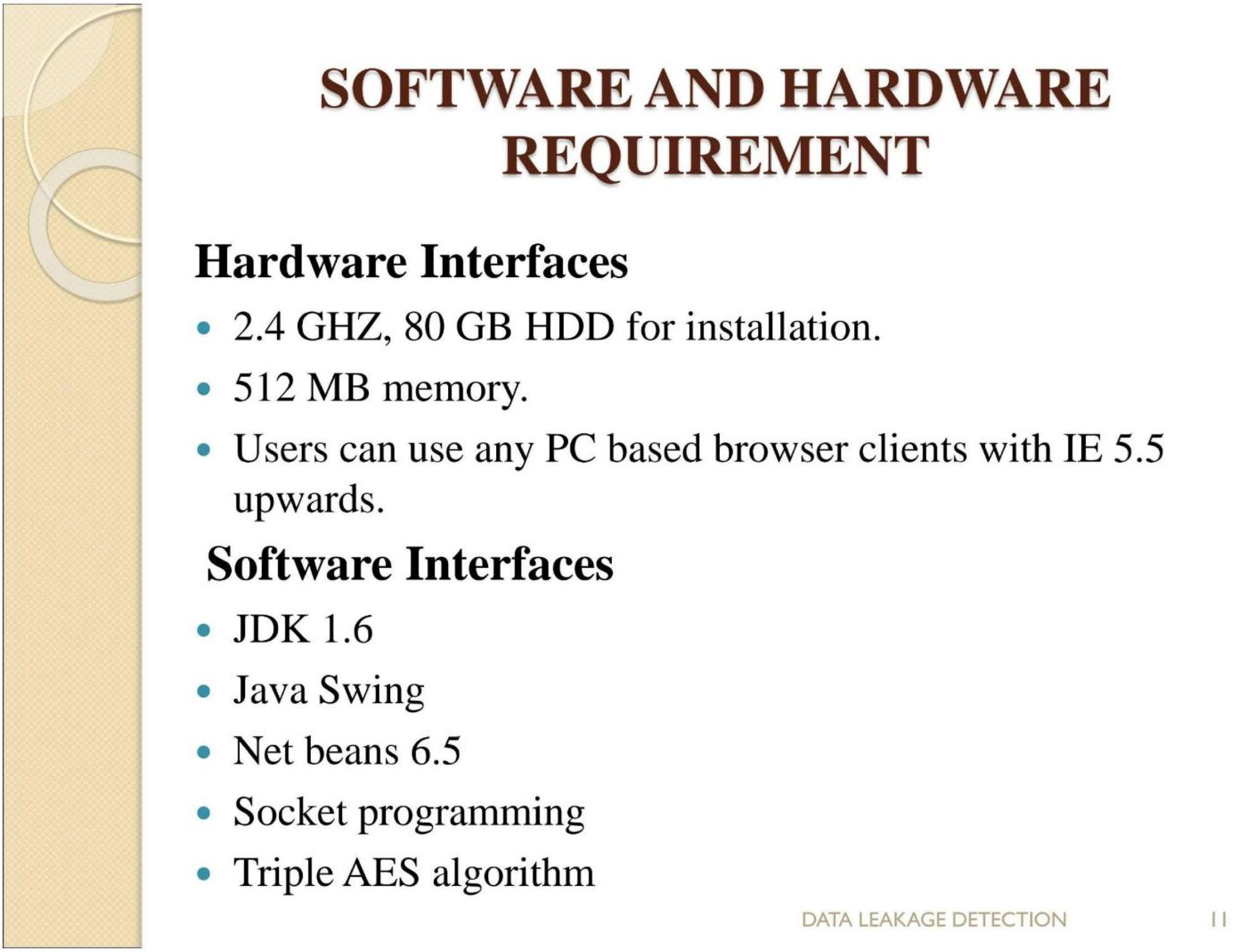


Scanned by TapScanner



SYSTEM ARCHITECTURE DIAGRAM

Scanned by TapScanner



SOFTWARE AND HARDWARE REQUIREMENT

Hardware Interfaces

- 2.4 GHZ, 80 GB HDD for installation.
- 512 MB memory.
- Users can use any PC based browser clients with IE 5.5 upwards.

Software Interfaces

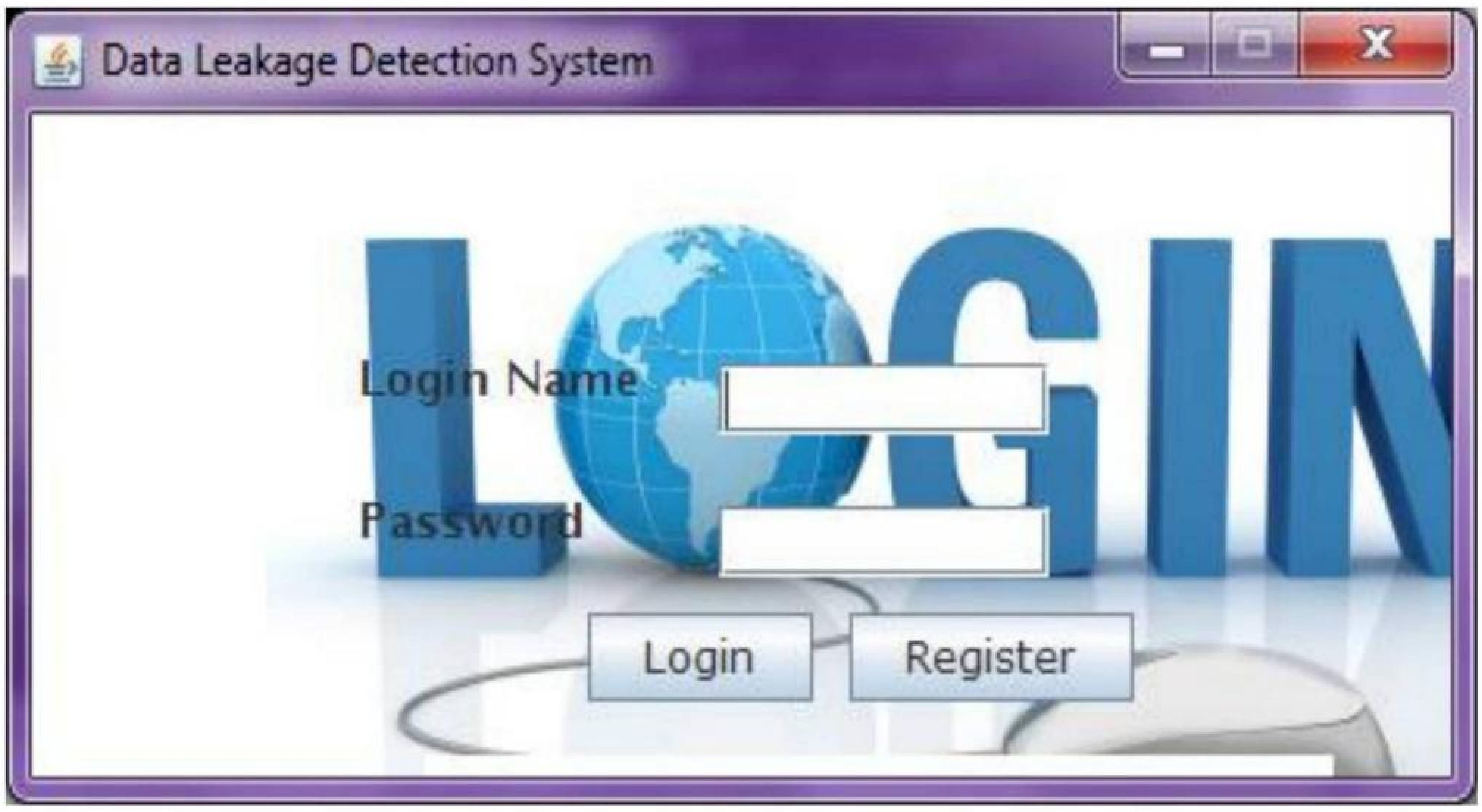
- JDK 1.6
- Java Swing
- Net beans 6.5
- Socket programming
- Triple AES algorithm

Scanned by TapScanner

SCREEN SHOTS

Scanned by TapScanner

1.User Login



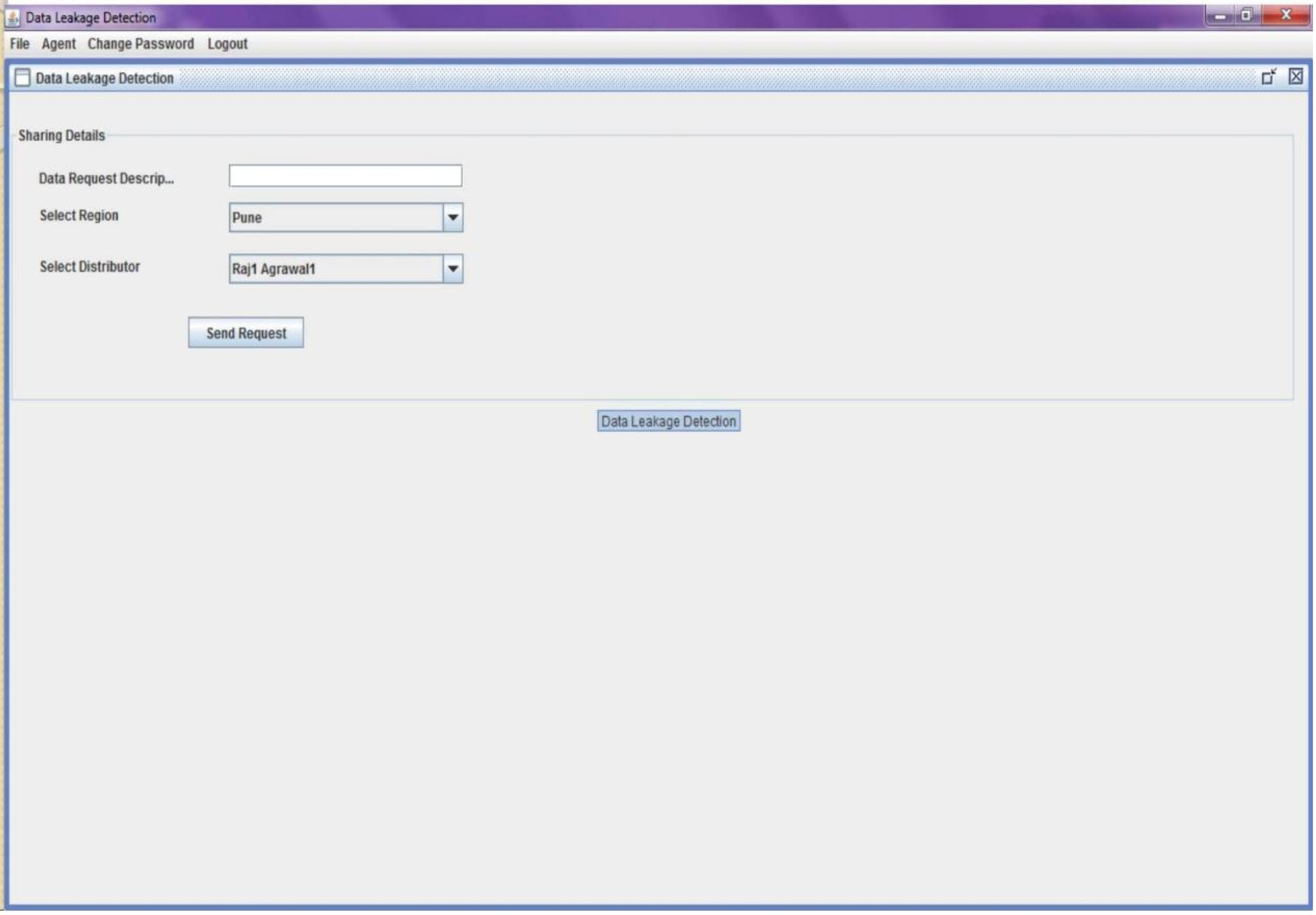
Data Leakage Detection System

Login Name

Password

Scanned by TapScanner

2. Agent Form(Request)



Sharing Details

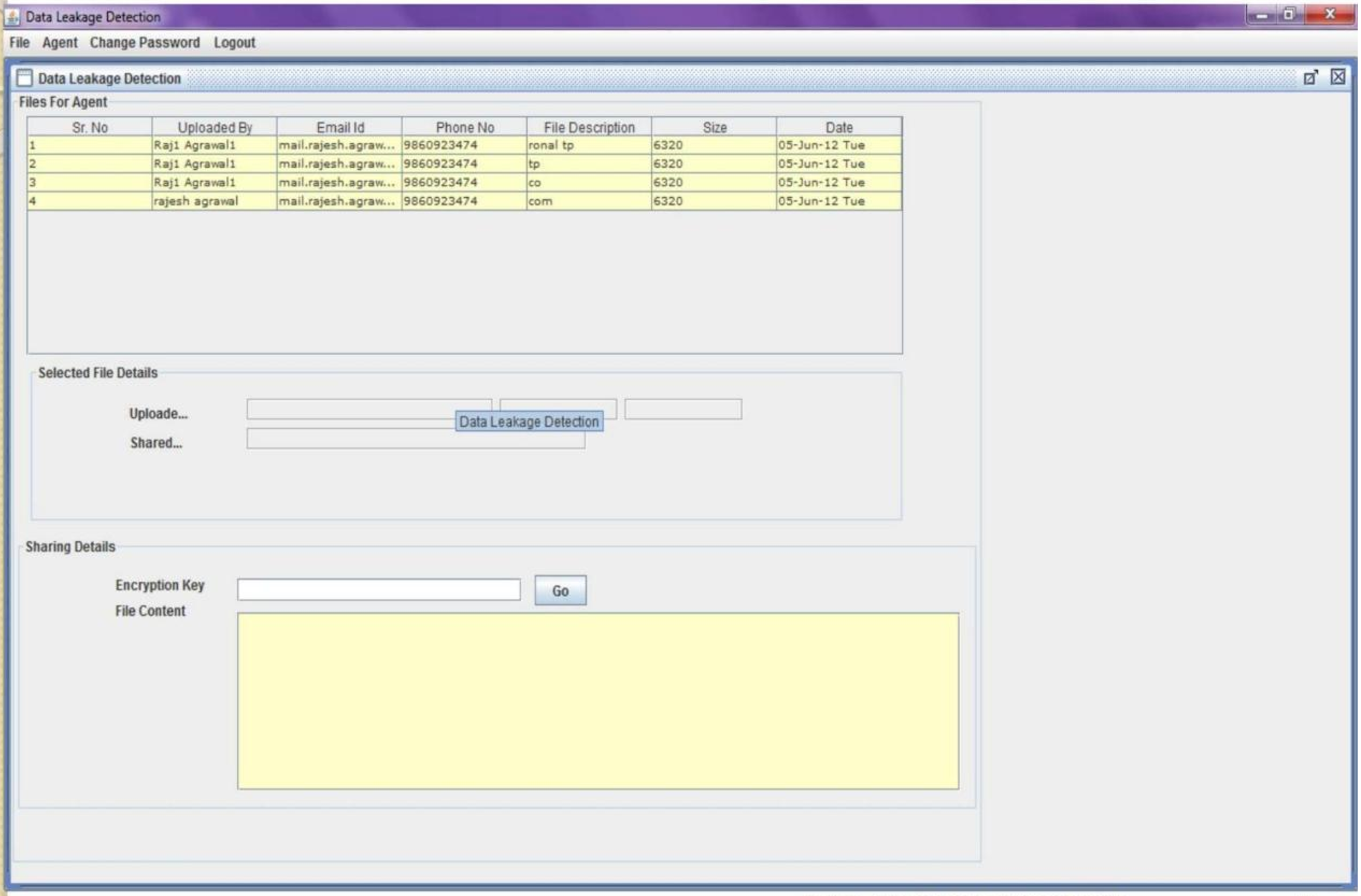
Data Request Descrip...

Select Region: Pune

Select Distributor: Raj1 Agrawal1

Scanned by TapScanner

3. Agent Form(Download Form)



Files For Agent

Sr. No	Uploaded By	Email Id	Phone No	File Description	Size	Date
1	Raj1 Agrawal1	mail.rajesh.agraw...	9860923474	ronal tp	6320	05-Jun-12 Tue
2	Raj1 Agrawal1	mail.rajesh.agraw...	9860923474	tp	6320	05-Jun-12 Tue
3	Raj1 Agrawal1	mail.rajesh.agraw...	9860923474	co	6320	05-Jun-12 Tue
4	rajesh agrawal	mail.rajesh.agraw...	9860923474	com	6320	05-Jun-12 Tue

Selected File Details

Uploade...

Shared...

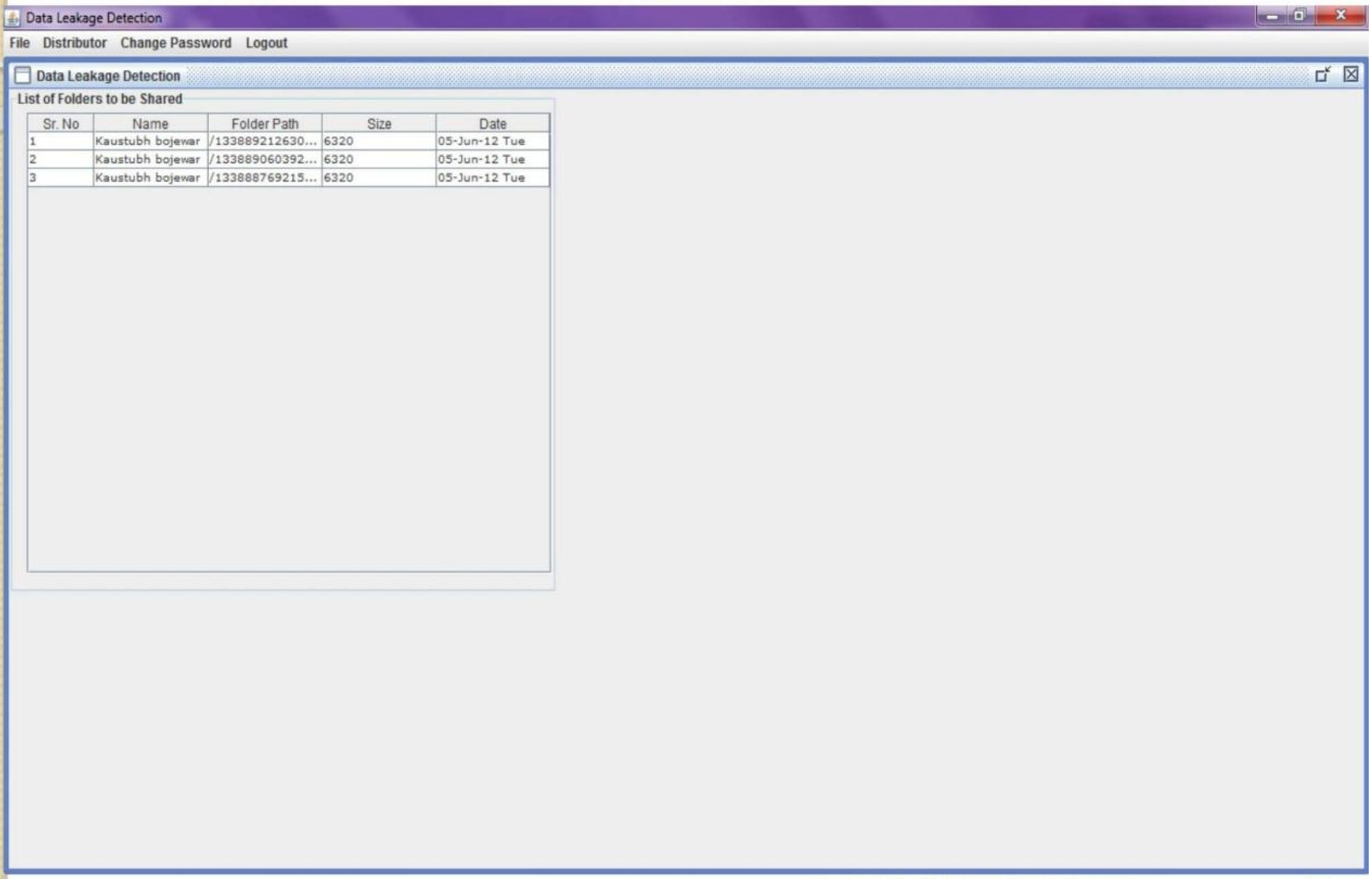
Sharing Details

Encryption Key

File Content

Scanned by TapScanner

4.Distributor(View shared files)

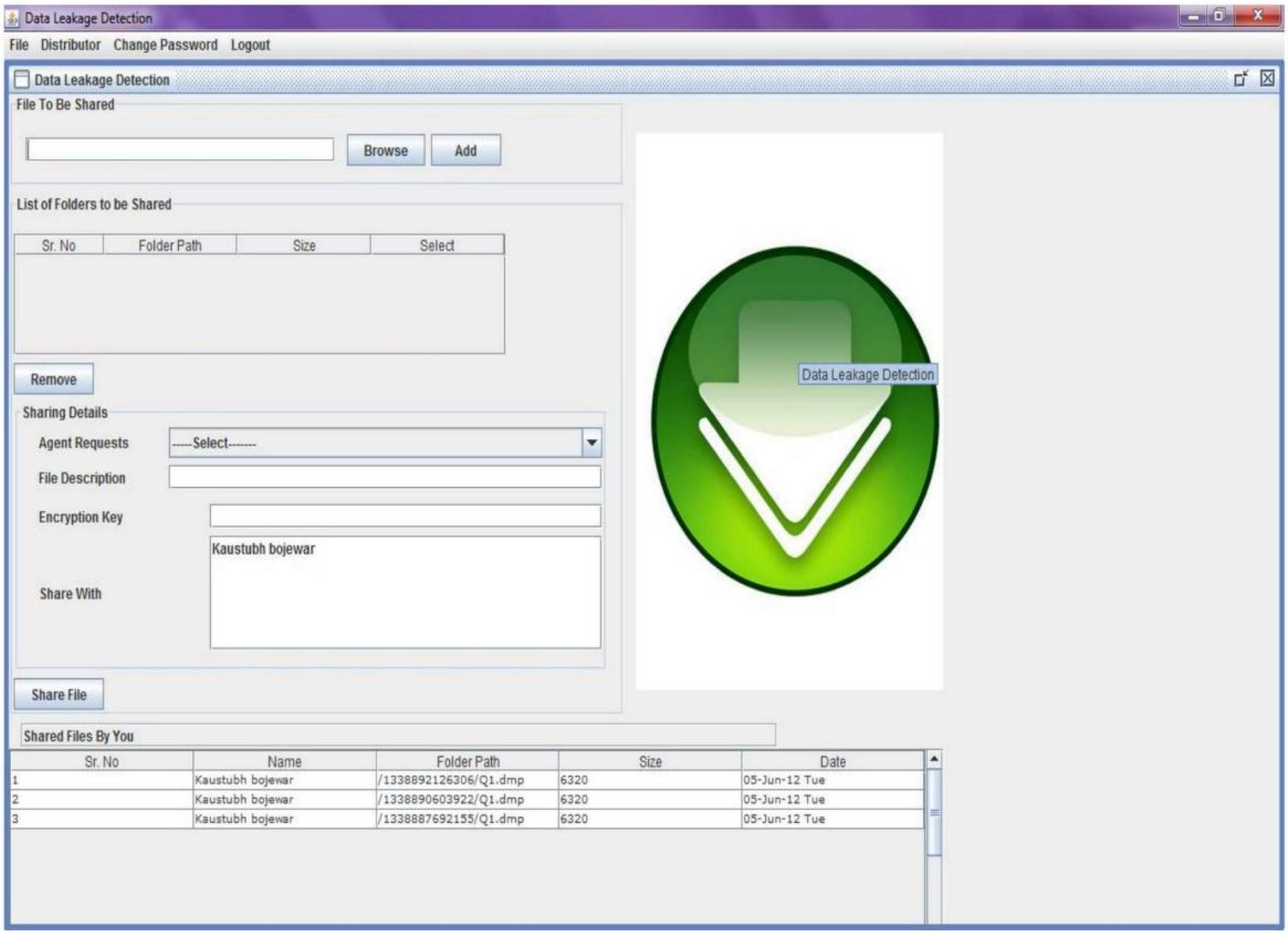


List of Folders to be Shared

Sr. No	Name	Folder Path	Size	Date
1	Kaustubh bojewar	/133889212630...	6320	05-Jun-12 Tue
2	Kaustubh bojewar	/133889060392...	6320	05-Jun-12 Tue
3	Kaustubh bojewar	/133888769215...	6320	05-Jun-12 Tue

Scanned by TapScanner

5.Distributor(Upload Files)



Sr. No	Folder Path	Size	Select

Agent Requests: -----Select-------

File Description:

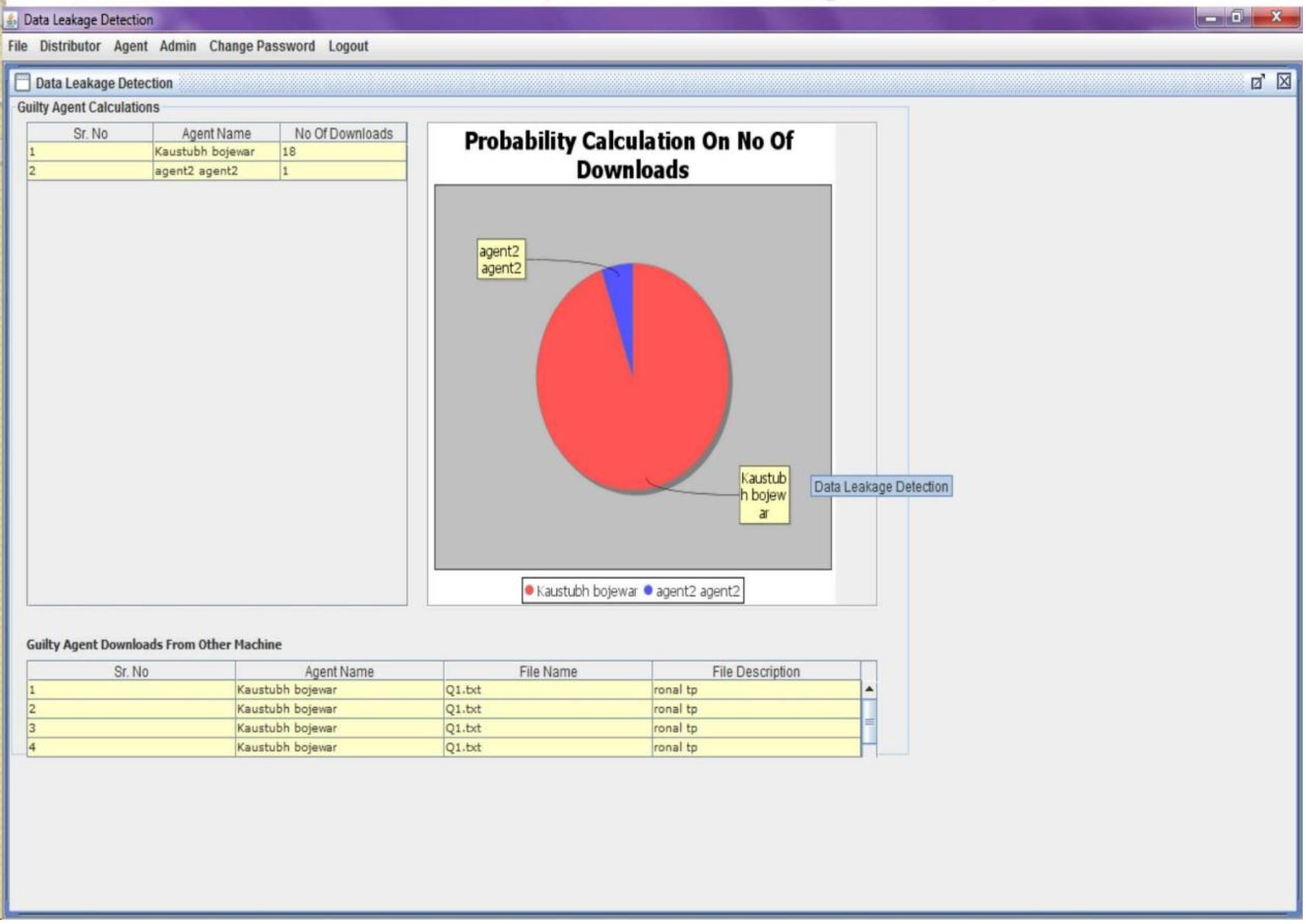
Encryption Key:

Share With: Kaustubh bojewar

Shared Files By You

Sr. No	Name	Folder Path	Size	Date
1	Kaustubh bojewar	/1338892126306/Q1.dmp	6320	05-Jun-12 Tue
2	Kaustubh bojewar	/1338890603922/Q1.dmp	6320	05-Jun-12 Tue
3	Kaustubh bojewar	/1338887692155/Q1.dmp	6320	05-Jun-12 Tue

6. Administrator (Probability Calc)



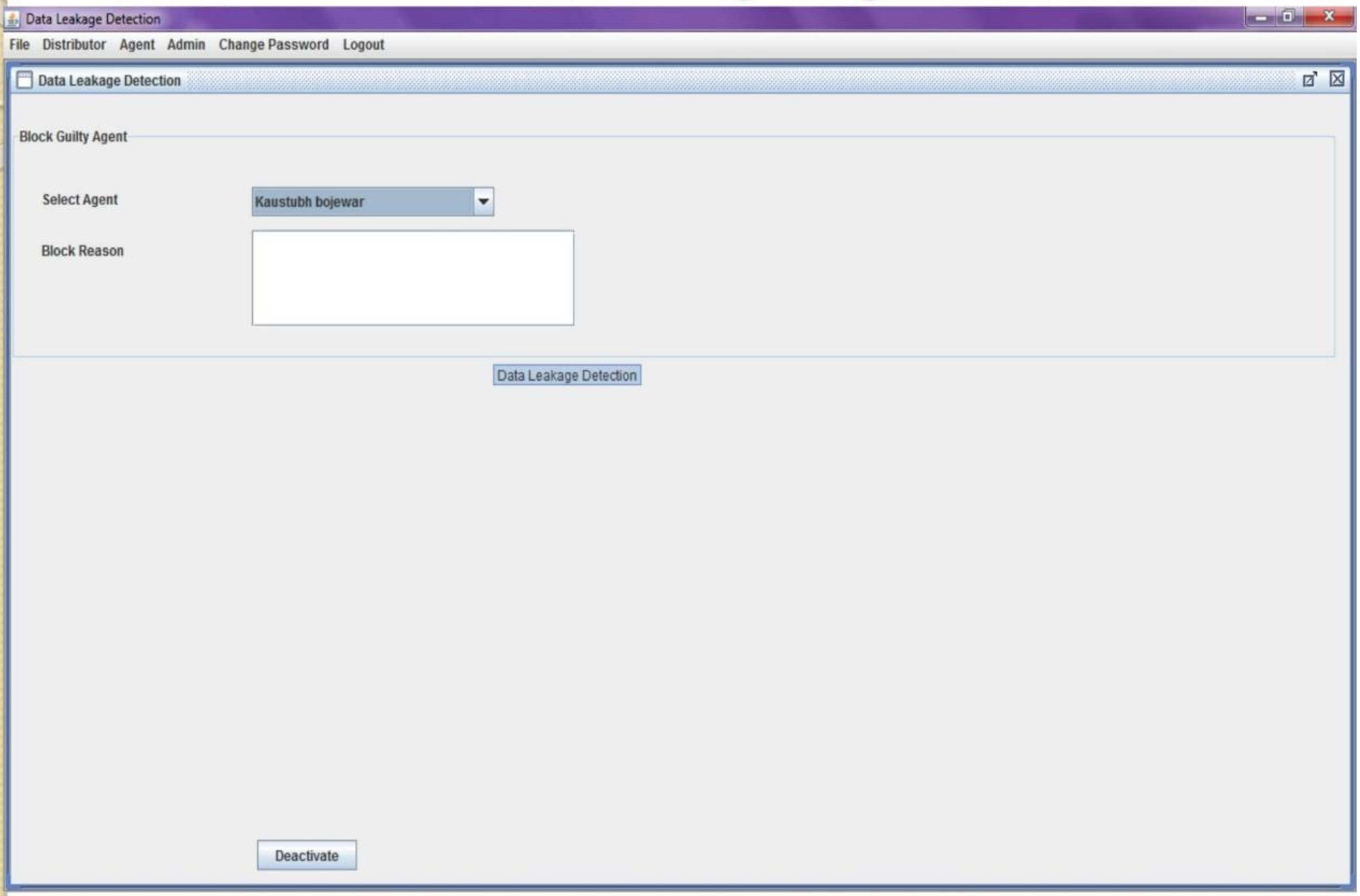
Guilty Agent Calculations

Sr. No	Agent Name	No Of Downloads
1	Kaustubh bojewar	18
2	agent2 agent2	1

Guilty Agent Downloads From Other Machine

Sr. No	Agent Name	File Name	File Description
1	Kaustubh bojewar	Q1.txt	ronal tp
2	Kaustubh bojewar	Q1.txt	ronal tp
3	Kaustubh bojewar	Q1.txt	ronal tp
4	Kaustubh bojewar	Q1.txt	ronal tp

7. Administrator (Manage Agents)

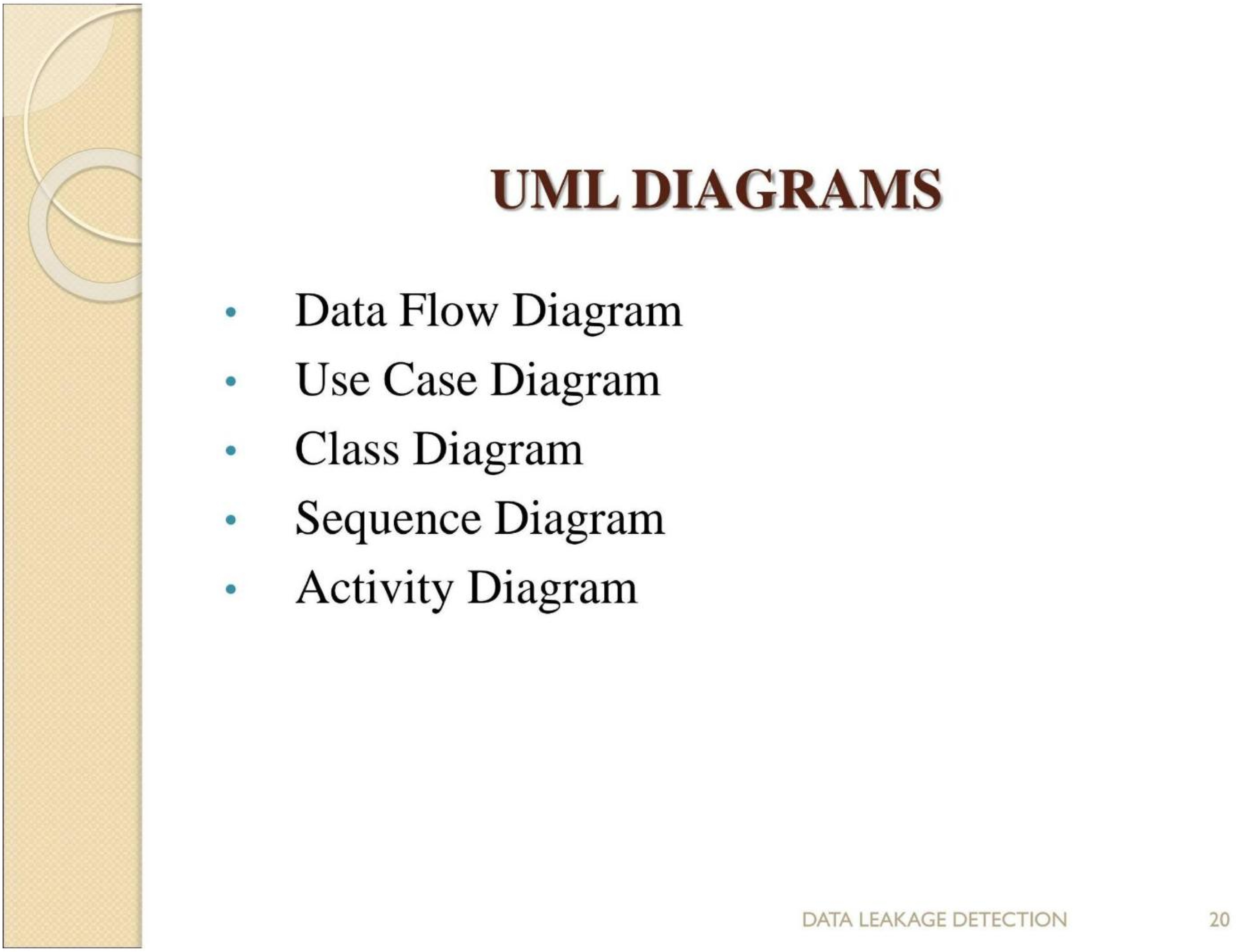


Block Guilty Agent

Select Agent: Kaustubh bojewar

Block Reason

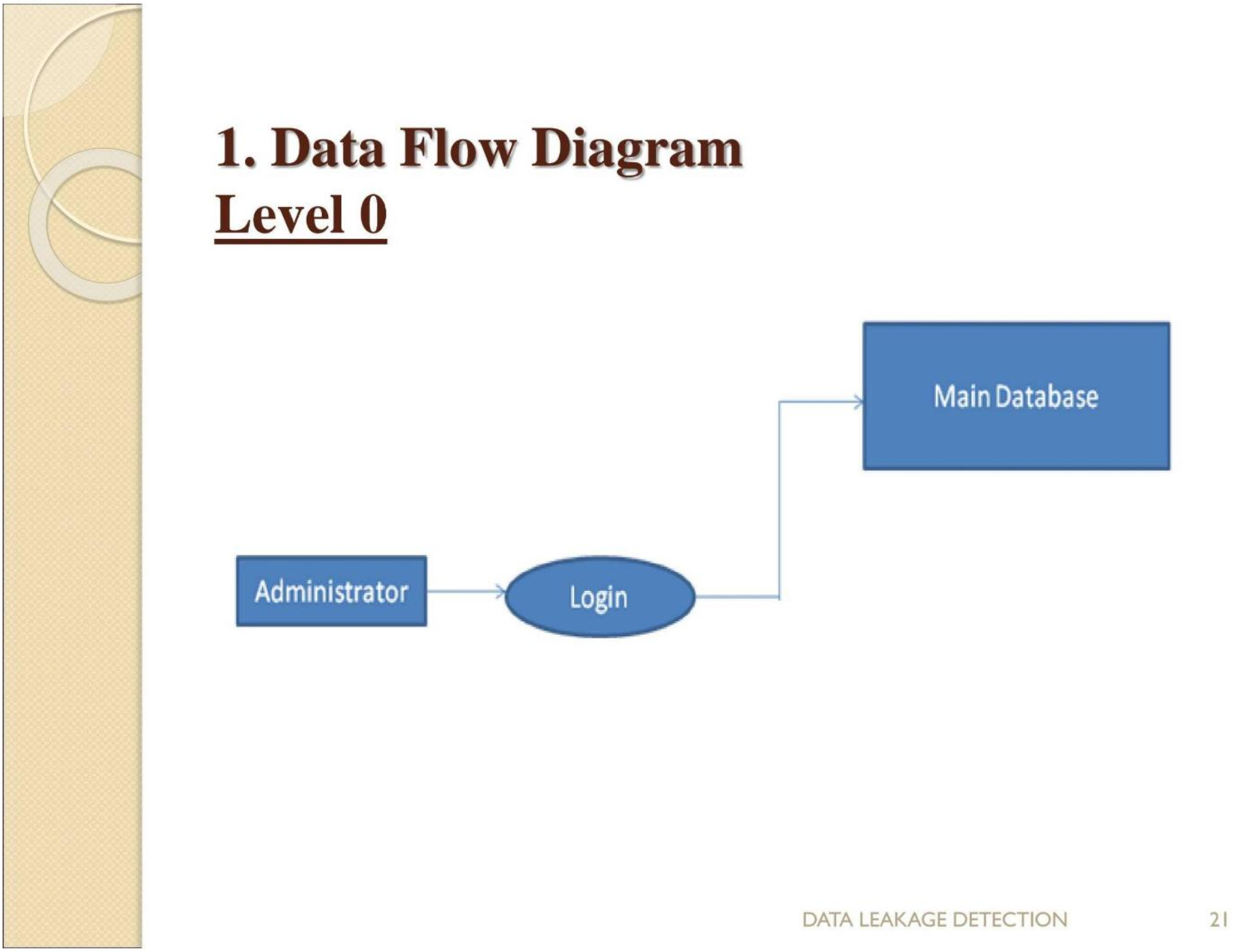
Scanned by TapScanner



UML DIAGRAMS

- Data Flow Diagram
- Use Case Diagram
- Class Diagram
- Sequence Diagram
- Activity Diagram

Scanned by TapScanner



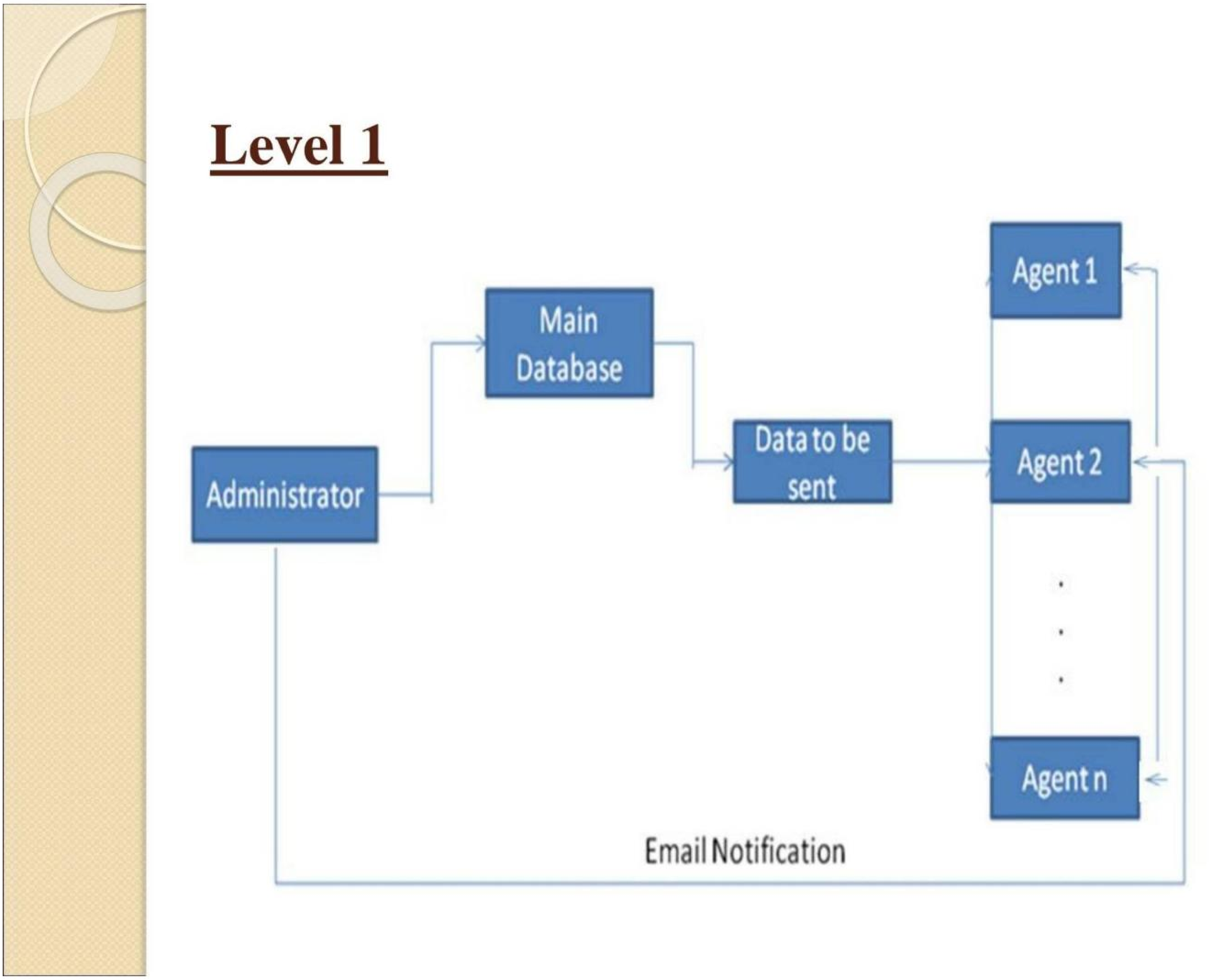
1. Data Flow Diagram

Level 0

Main Database

Administrator

Login



Level 1

Administrator

Main Database

Data to be sent

Agent 1

Agent 2

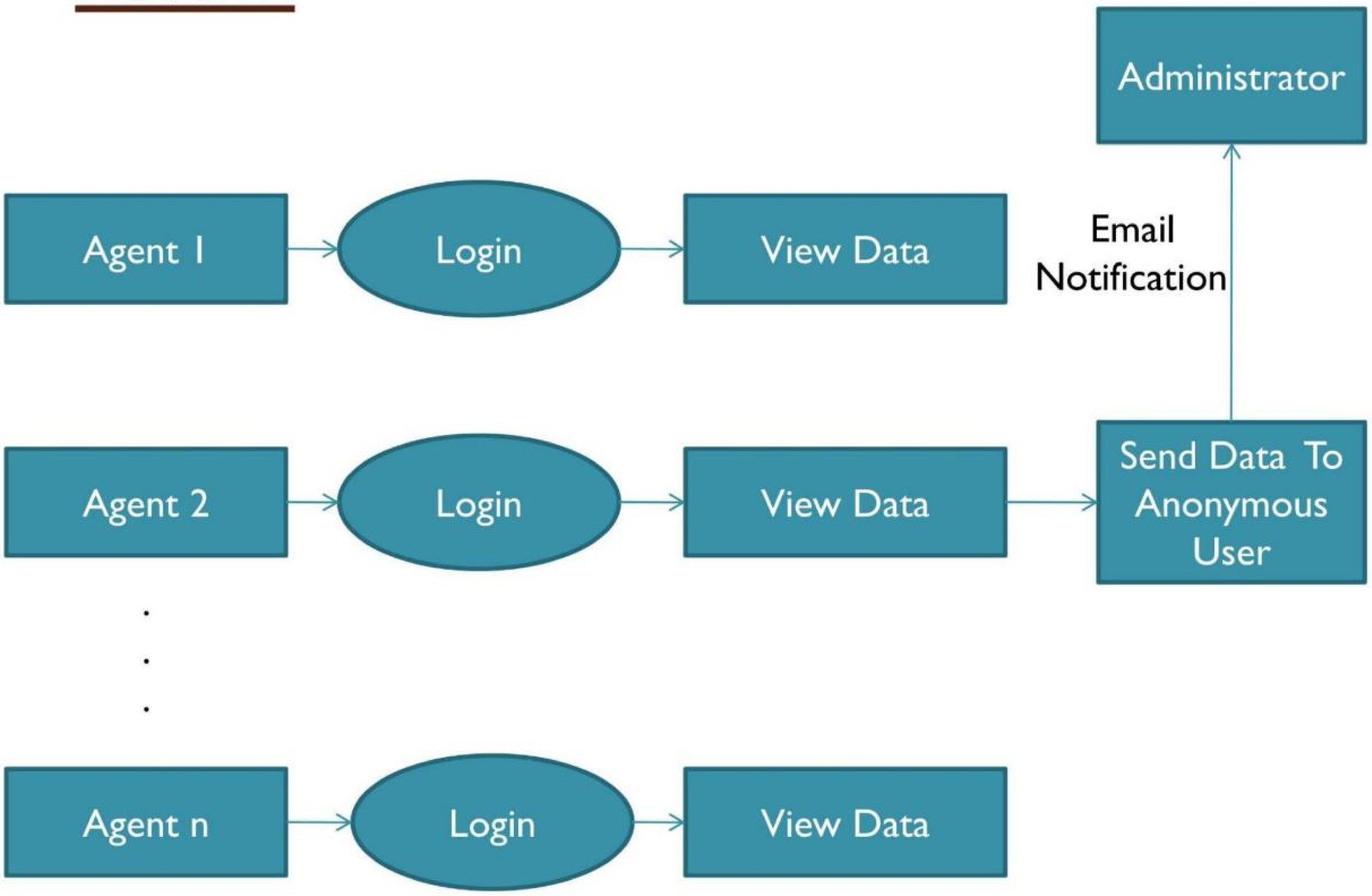
.
.
.

Agent n

Email Notification

Scanned by TapScanner

Level 2



Administrator

Agent 1

Login

View Data

Email Notification

Agent 2

Login

View Data

Send Data To Anonymous User

.
.
.

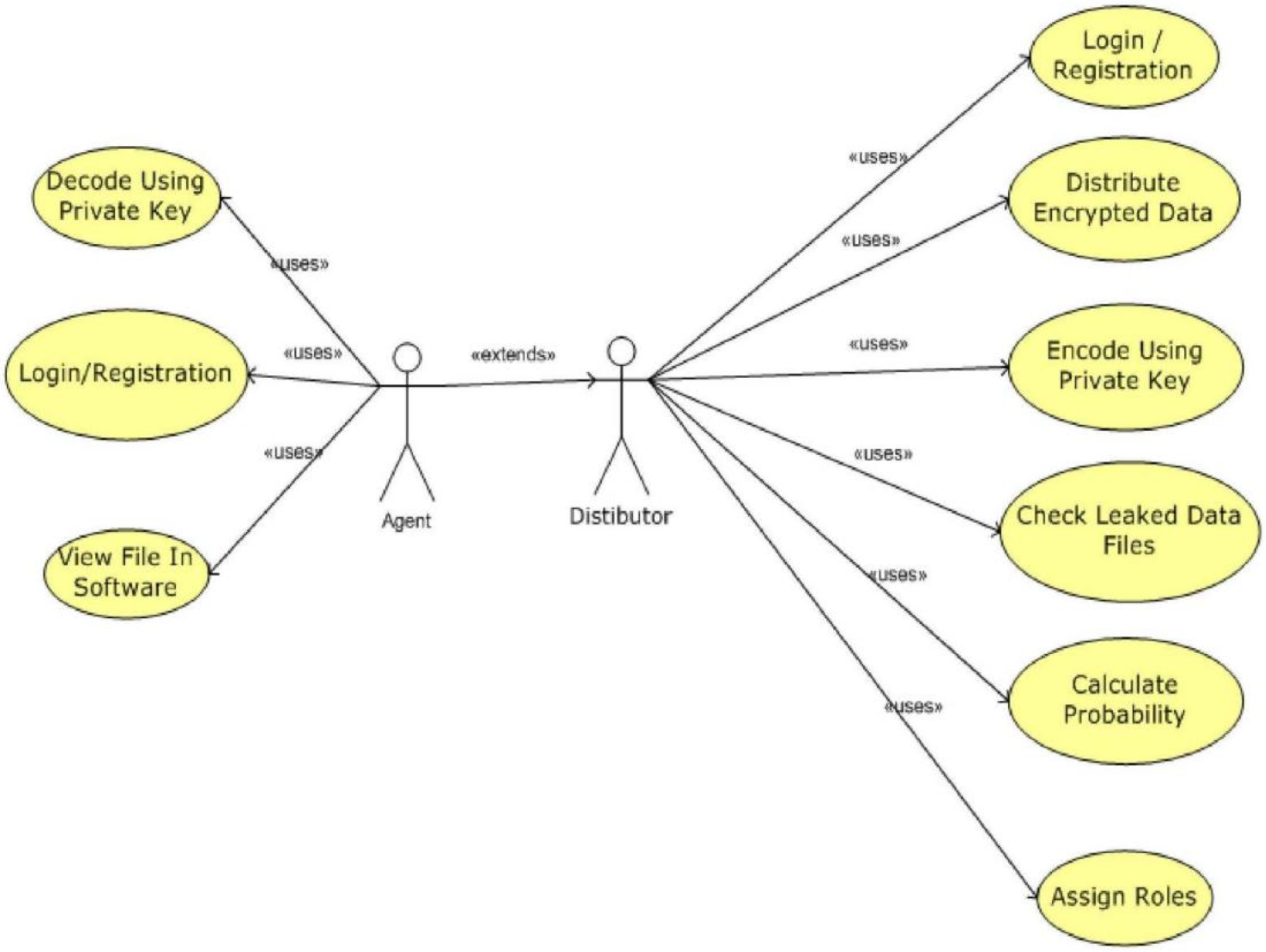
Agent n

Login

View Data

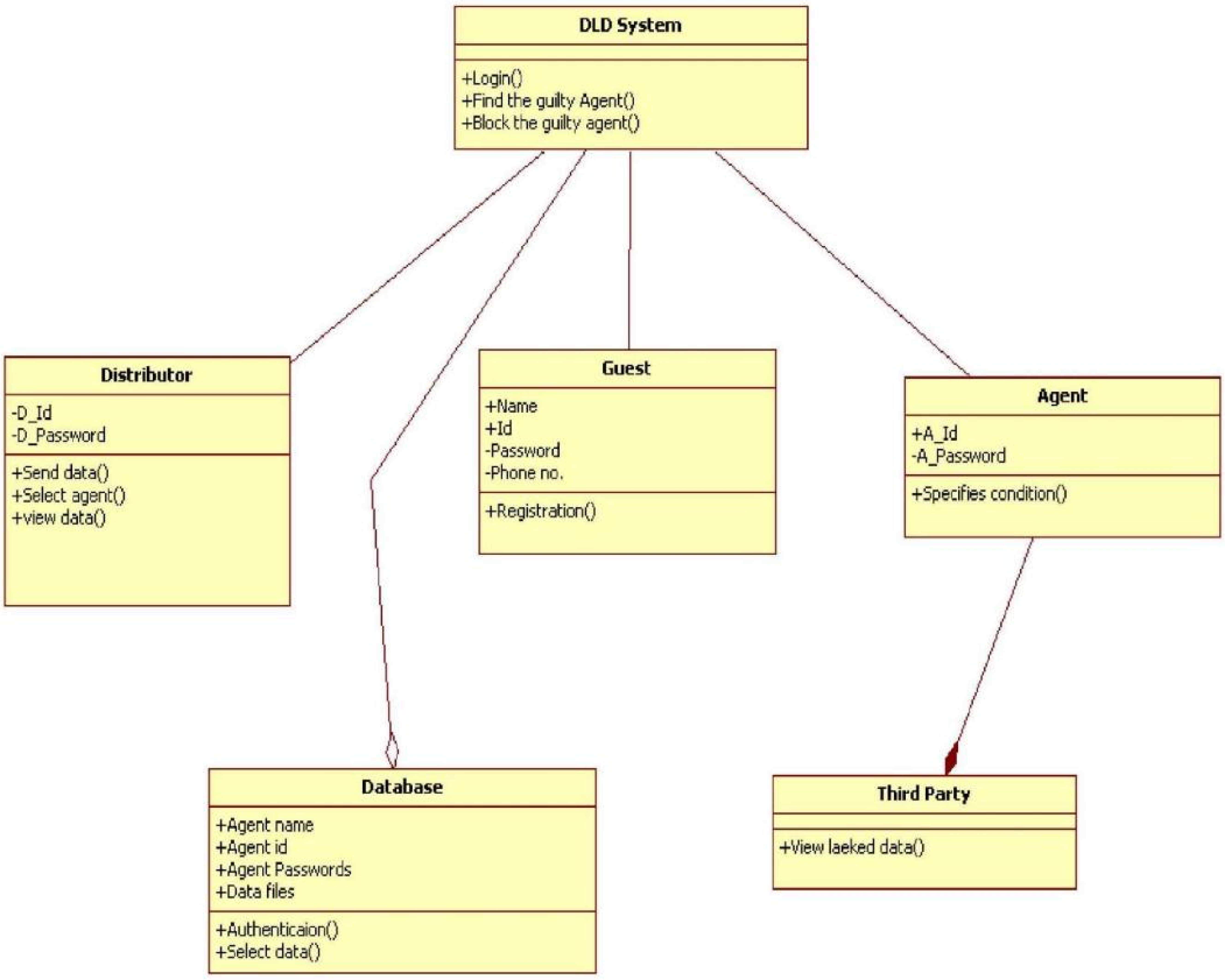
Scanned by TapScanner

2. Use Case Diagram



Scanned by TapScanner

3.Class Diagram



DLD System

+Login()
+Find the guilty Agent()
+Block the guilty agent()

Distributor

-D_Id
-D_Password

+Send data()
+Select agent()
+view data()

Guest

+Name
+Id
-Password
-Phone no.

+Registration()

Agent

+A_Id
-A_Password

+Specifies condition()

Database

+Agent name
+Agent id
+Agent Passwords
+Data files

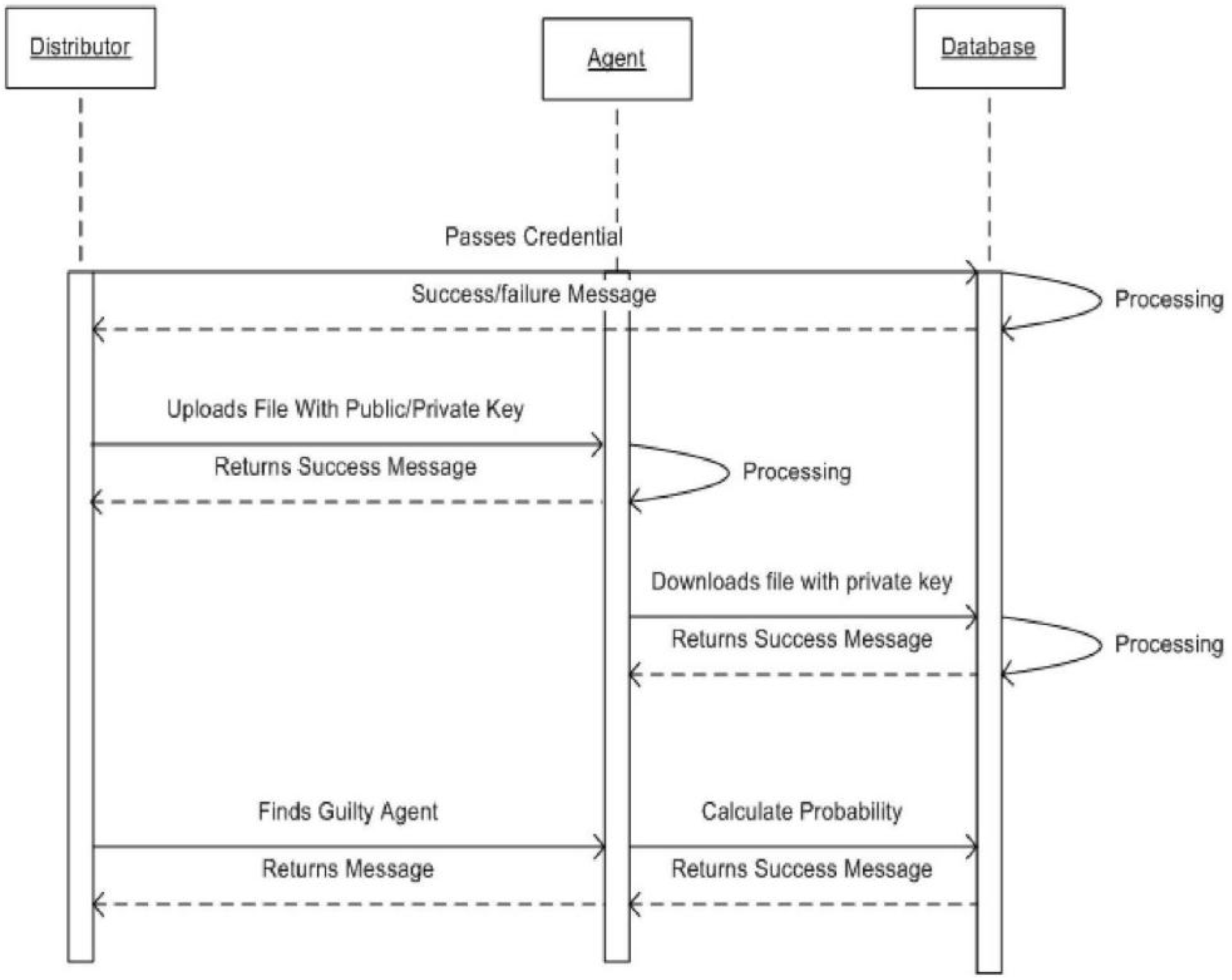
+Authenticaion()
+Select data()

Third Party

+View laeked data()

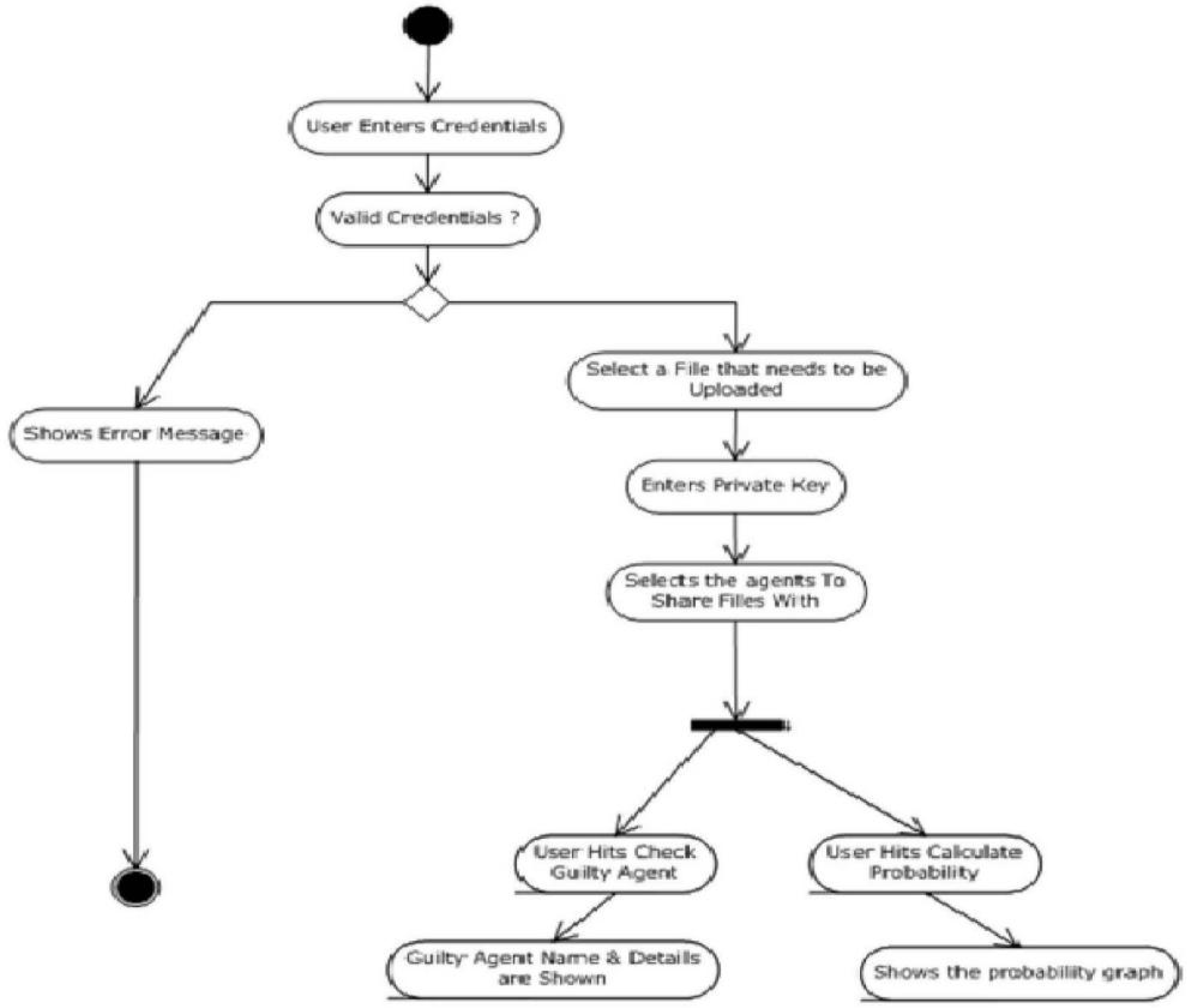
Scanned by TapScanner

4.Sequence Diagram



Scanned by TapScanner

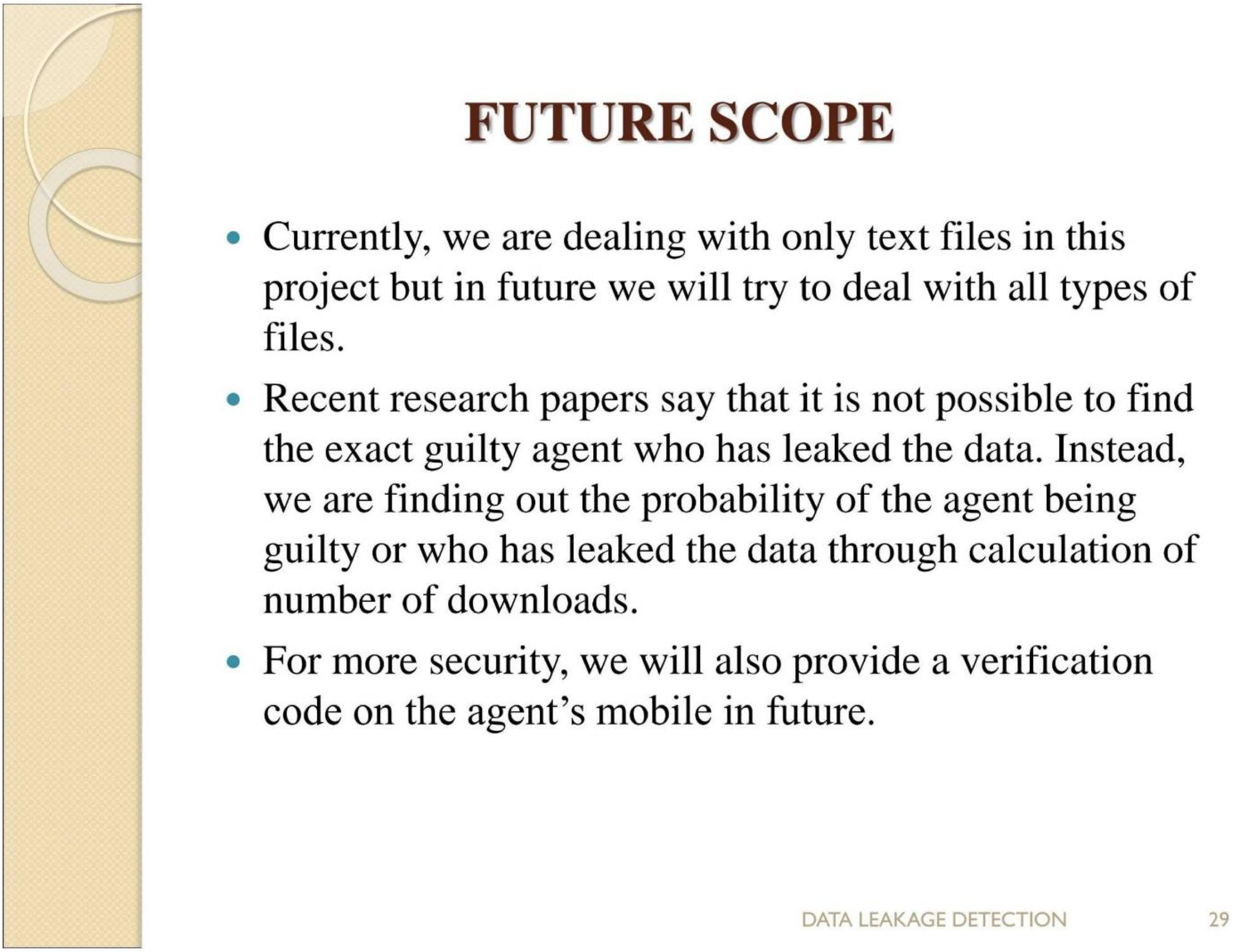
5. Activity Diagram


User Enters Credentials
Valid Credentials ?
Shows Error Message
Select a File that needs to be Uploaded
Enters Private Key
Selects the agents To Share Files With
User Hits Check Guilty Agent
User Hits Calculate Probability
Guilty Agent Name & Details are Shown
Shows the probability graph

Scanned by TapScanner

ADVANTAGES

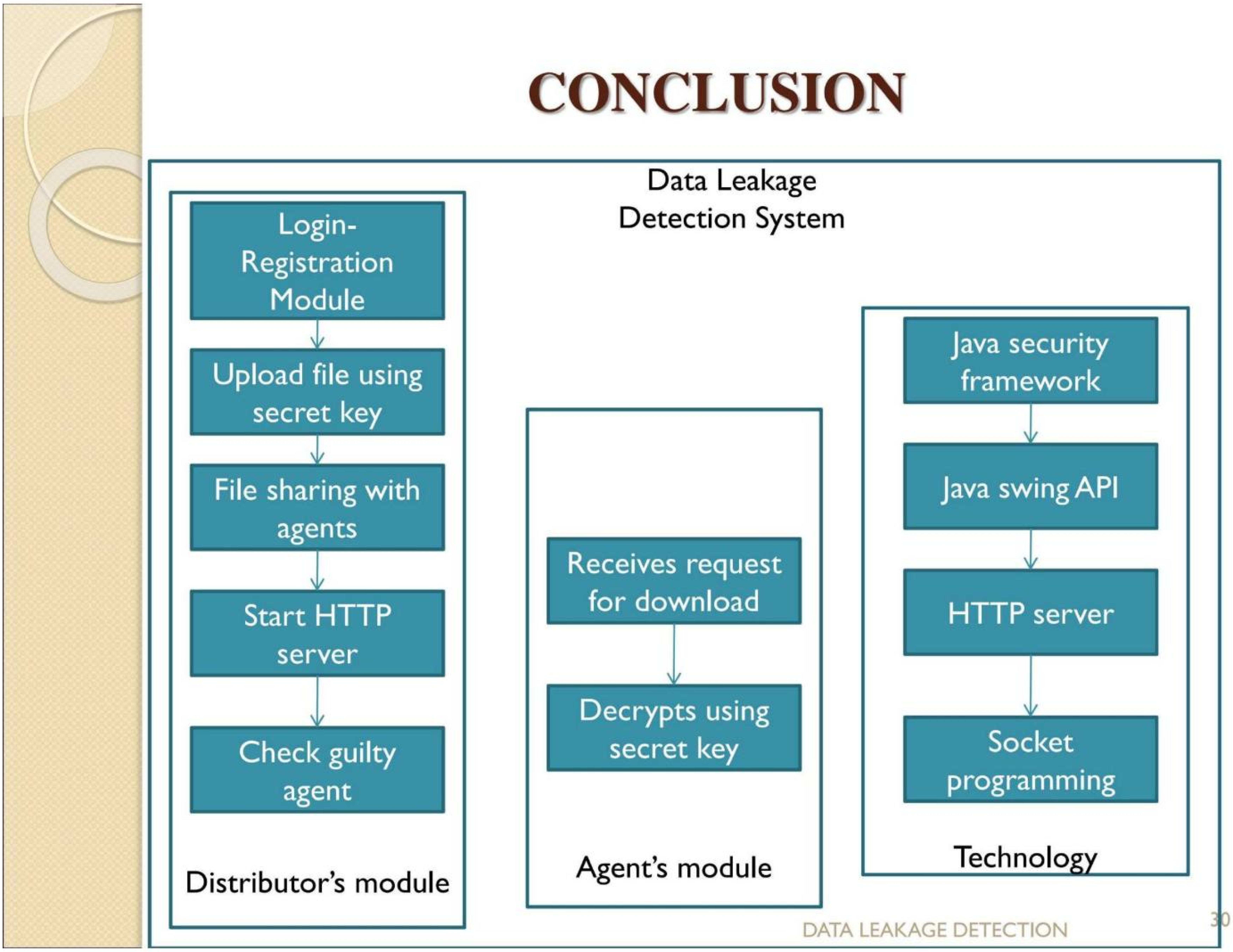
- This system includes the *data hiding* along with the provisional software with which only the data can be accessed.
- This system gives *privileged access to the administrator (data distributor) as well as the agents* registered by the distributors. Only registered agents can access the system. The user accounts can be activated as well as cancelled.
- The exported file will be accessed only by the system. The agent has given only the permission to access the software and view the data. If the data is leaked by the agent' system the path and agent information will be sent to the distributor thereby the identity of the leaked user can be traced.



FUTURE SCOPE

- Currently, we are dealing with only text files in this project but in future we will try to deal with all types of files.
- Recent research papers say that it is not possible to find the exact guilty agent who has leaked the data. Instead, we are finding out the probability of the agent being guilty or who has leaked the data through calculation of number of downloads.
- For more security, we will also provide a verification code on the agent’s mobile in future.

Scanned by TapScanner



CONCLUSION

Data Leakage Detection System

Distributor's module

- Login-Registration Module
- Upload file using secret key
- File sharing with agents
- Start HTTP server
- Check guilty agent

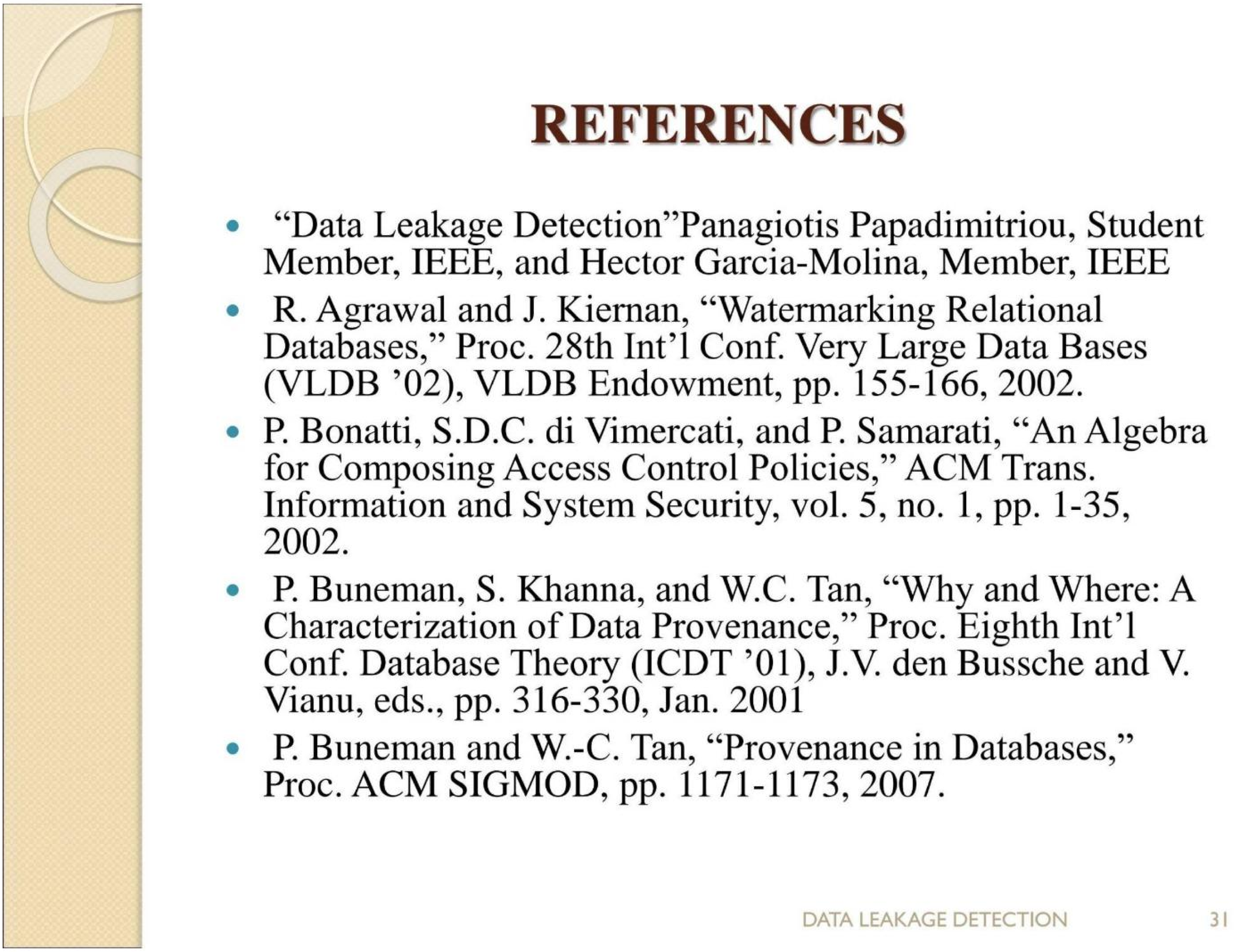
Agent's module

- Receives request for download
- Decrypts using secret key

Technology

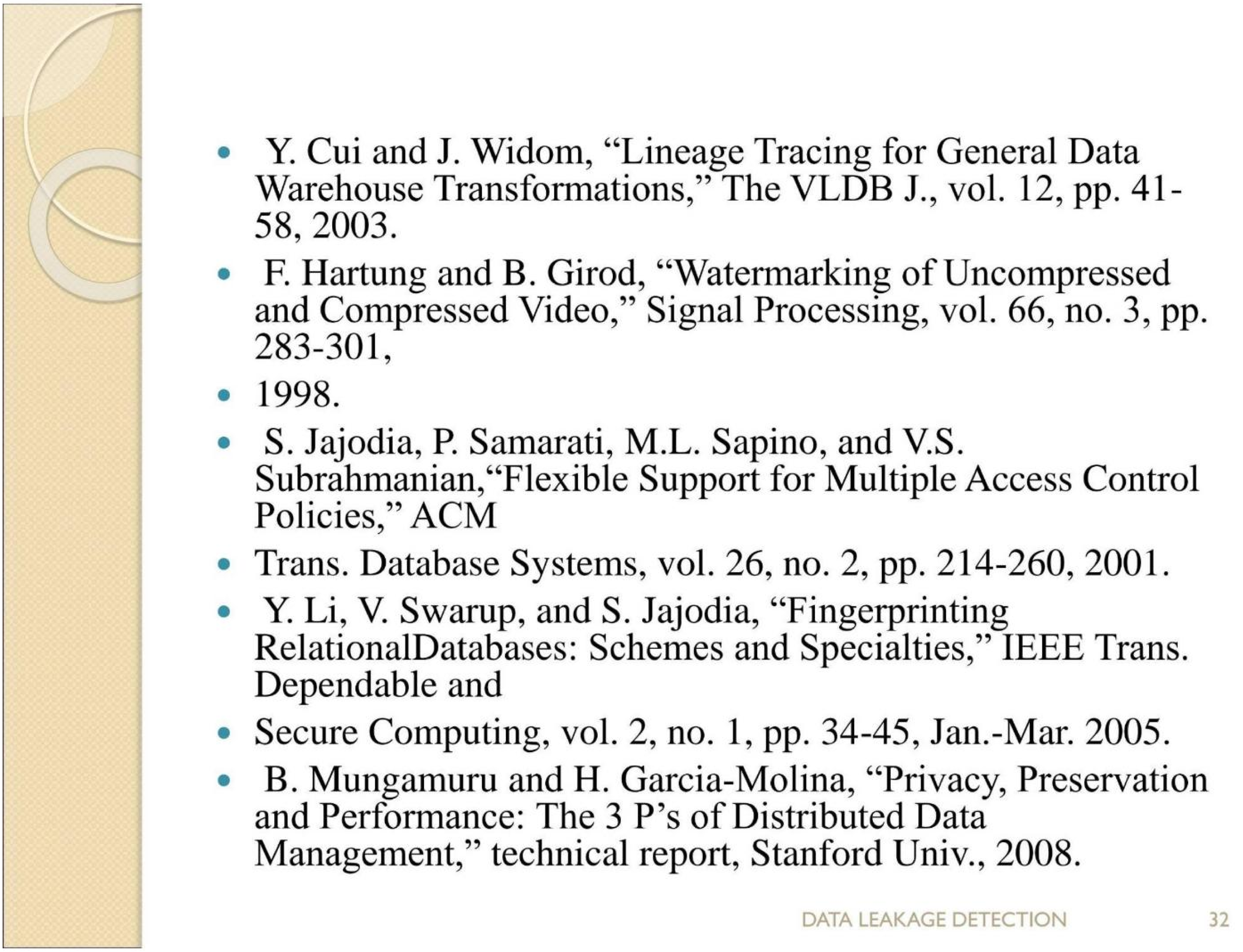
- Java security framework
- Java swing API
- HTTP server
- Socket programming

Scanned by TapScanner



REFERENCES

- “Data Leakage Detection”Panagiotis Papadimitriou, Student Member, IEEE, and Hector Garcia-Molina, Member, IEEE
- R. Agrawal and J. Kiernan, “Watermarking Relational Databases,” Proc. 28th Int’l Conf. Very Large Data Bases (VLDB ’02), VLDB Endowment, pp. 155-166, 2002.
- P. Bonatti, S.D.C. di Vimercati, and P. Samarati, “An Algebra for Composing Access Control Policies,” ACM Trans. Information and System Security, vol. 5, no. 1, pp. 1-35, 2002.
- P. Buneman, S. Khanna, and W.C. Tan, “Why and Where: A Characterization of Data Provenance,” Proc. Eighth Int’l Conf. Database Theory (ICDT ’01), J.V. den Bussche and V. Vianu, eds., pp. 316-330, Jan. 2001
- P. Buneman and W.-C. Tan, “Provenance in Databases,” Proc. ACM SIGMOD, pp. 1171-1173, 2007.



- Y. Cui and J. Widom, “Lineage Tracing for General Data Warehouse Transformations,” The VLDB J., vol. 12, pp. 41-58, 2003.
- F. Hartung and B. Girod, “Watermarking of Uncompressed and Compressed Video,” Signal Processing, vol. 66, no. 3, pp. 283-301,
- 1998.
- S. Jajodia, P. Samarati, M.L. Sapino, and V.S. Subrahmanian,“Flexible Support for Multiple Access Control Policies,” ACM
- Trans. Database Systems, vol. 26, no. 2, pp. 214-260, 2001.
- Y. Li, V. Swarup, and S. Jajodia, “Fingerprinting RelationalDatabases: Schemes and Specialties,” IEEE Trans. Dependable and
- Secure Computing, vol. 2, no. 1, pp. 34-45, Jan.-Mar. 2005.
- B. Mungamuru and H. Garcia-Molina, “Privacy, Preservation and Performance: The 3 P’s of Distributed Data Management,” technical report, Stanford Univ., 2008.

Scanned by TapScanner

THANK YOU...

Scanned by TapScanner