

# Crypt-DAC Cryptographically Enforced Dynamic Access Control in the Cloud



## ABSTRACT:

- ❖ cryptographically enforced access controls for data hosted in untrusted cloud is attractive for many users and organizations. However, designing efficient cryptographically enforced dynamic access control system in the cloud is still challenging.
- ❖ In this paper, we propose Crypt-DAC, a system that provides practical cryptographic enforcement of dynamic access control.
- ❖ Crypt-DAC revokes access permissions by delegating the cloud to update encrypted data. In Crypt-DAC, a file is encrypted by a symmetric key list which records a file key and a sequence of revocation keys.
- ❖ In each revocation, a dedicated administrator uploads a new revocation key to the cloud and requests it to encrypt the file with a new layer of encryption and update the encrypted key list accordingly.

# INTRODUCTION

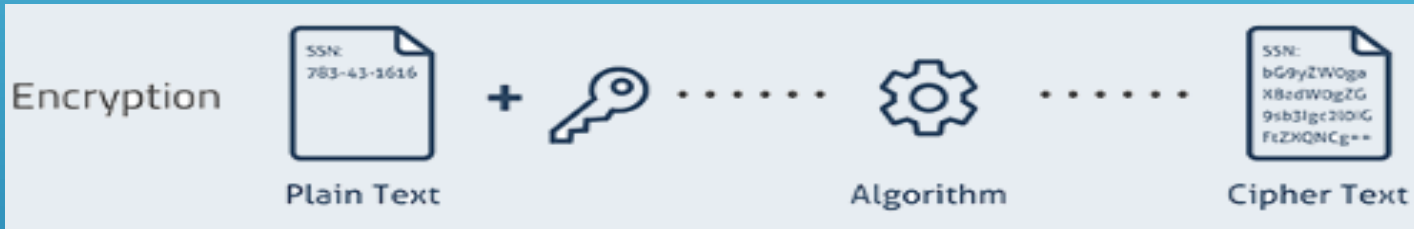
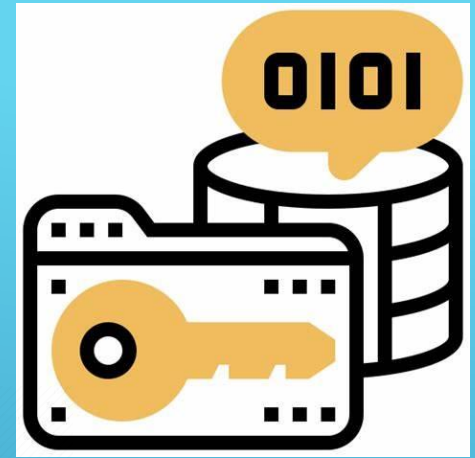
- ❑ With the considerable advancements in cloud computing, users and organizations are finding it increasingly appealing to store and share data through cloud services.
- ❑ Cloud service providers (such as Amazon, Microsoft, Apple, etc.) provide abundant cloud based services, ranging from small-scale personal services to large-scale industrial services. However, recent data breaches, such as releases of private photos , have raised concerns regarding the privacy of cloud-managed data. Actually, a cloud service provider is usually not secure due to design drawbacks of software and system vulnerability . As such, a critical issue is how to enforce data access control on the potentially untrusted cloud.

# Cryptography

- Cryptography derived its name from a Greek word called “krypto’s” which means “Hidden Secrets”.
- Cryptography is the practice and study of hiding information. It is the Art or Science of converting a plain intelligible data into an unintelligible data and again retransforming that message into its original form.
- It provides Confidentiality, Integrity, and Accuracy.

# ENCRYPTION:

Encryption is a process which transforms the original information into an unrecognizable form. This new form of the message is entirely different from the original message. That's why a hacker is not able to read the data as senders use an encryption algorithm.

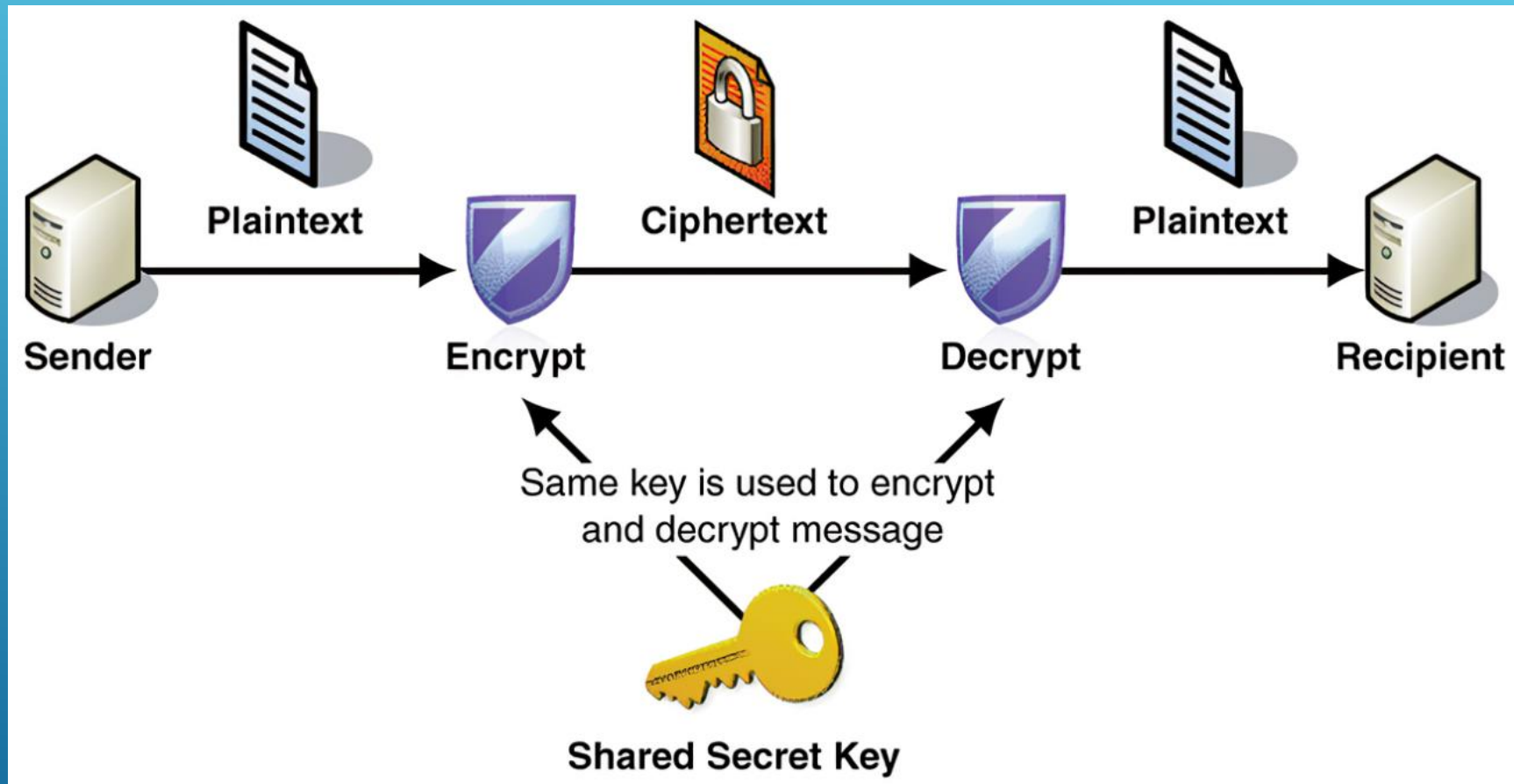


## Decryption

Decryption is a process of converting encoded/encrypted data in a form that is readable and understood by a human or a computer. This method is performed by un-encrypting the text manually or by using keys used to encrypt the original data.



## Encryption and Decryption DIAGRAM



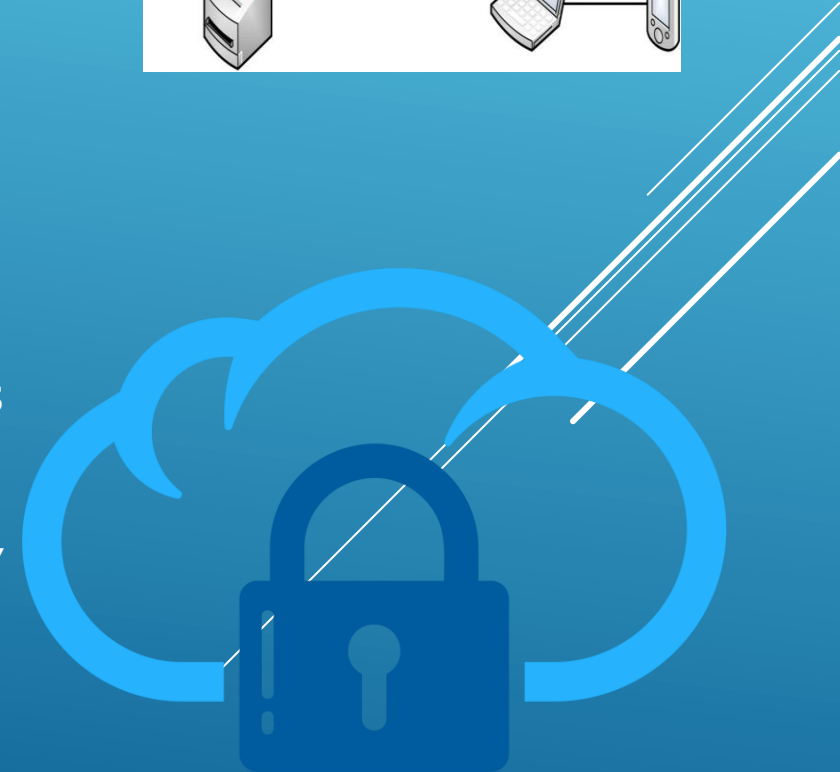
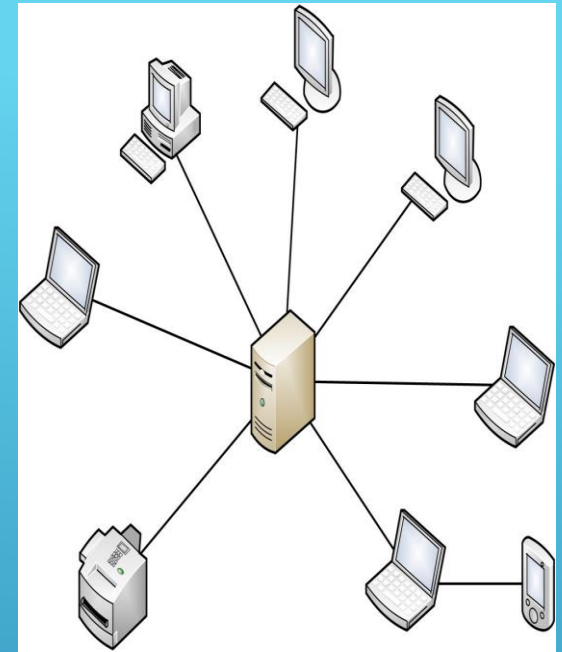


# CLIENT SERVER

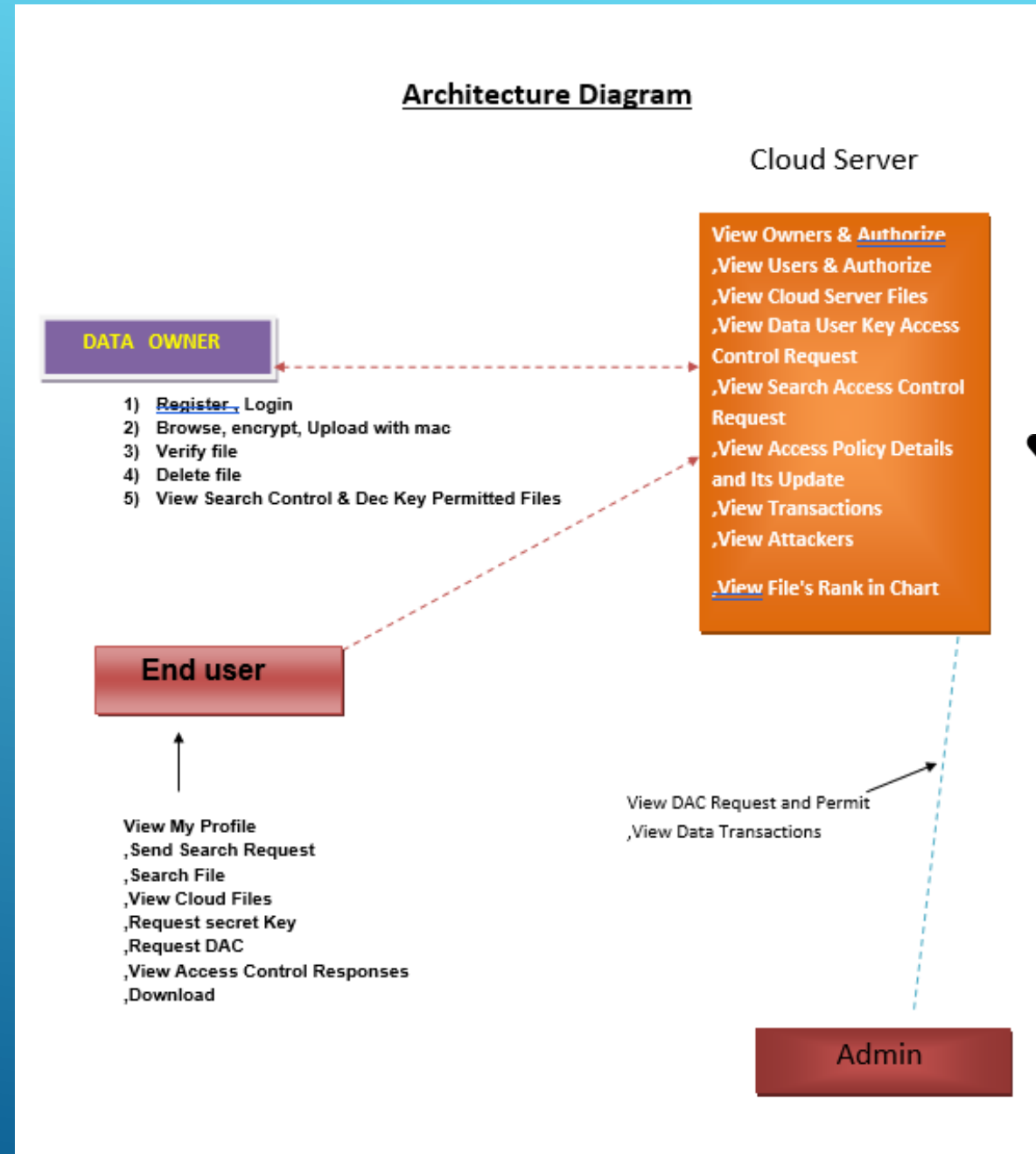
Client-server denotes a relationship between cooperating programs in an application, composed of clients initiating requests for services and servers providing that function or service.

## Security:

- Every time you download a “normal” program, you are risking a viral infection.
- Prior to Java, most users did not download executable programs frequently, and those who did scanned them for viruses prior to execution.
- This type of program can gather private information, such as credit card numbers, bank account balances, and passwords.
- Java answers both these concerns by providing a “firewall” between a network application and your computer.



# Architecture Diagram





# CRYPT-DAC

HOME PAGE

localhost:8090/Crypt%20DAC%20Cryptographically%20Enforced%20Dynamic%20Access%20Control%20in%20the%20Cloud/index.html

Apps Gmail YouTube Maps Translate

Organization

Admin

Users

Cloud

Access policy update

File data read/write

Policy data

File data

Policy data

File data

Search our site:

## Menu

- Home
- Cloud Data Server
- Data Owner
- End User
- Admin

## Crypt-DAC: Cryptographically Enforced Dynamic Access Control in the Cloud

Organization

Admin

Users

Cloud

Access policy update

File data read/write

Policy data

File data

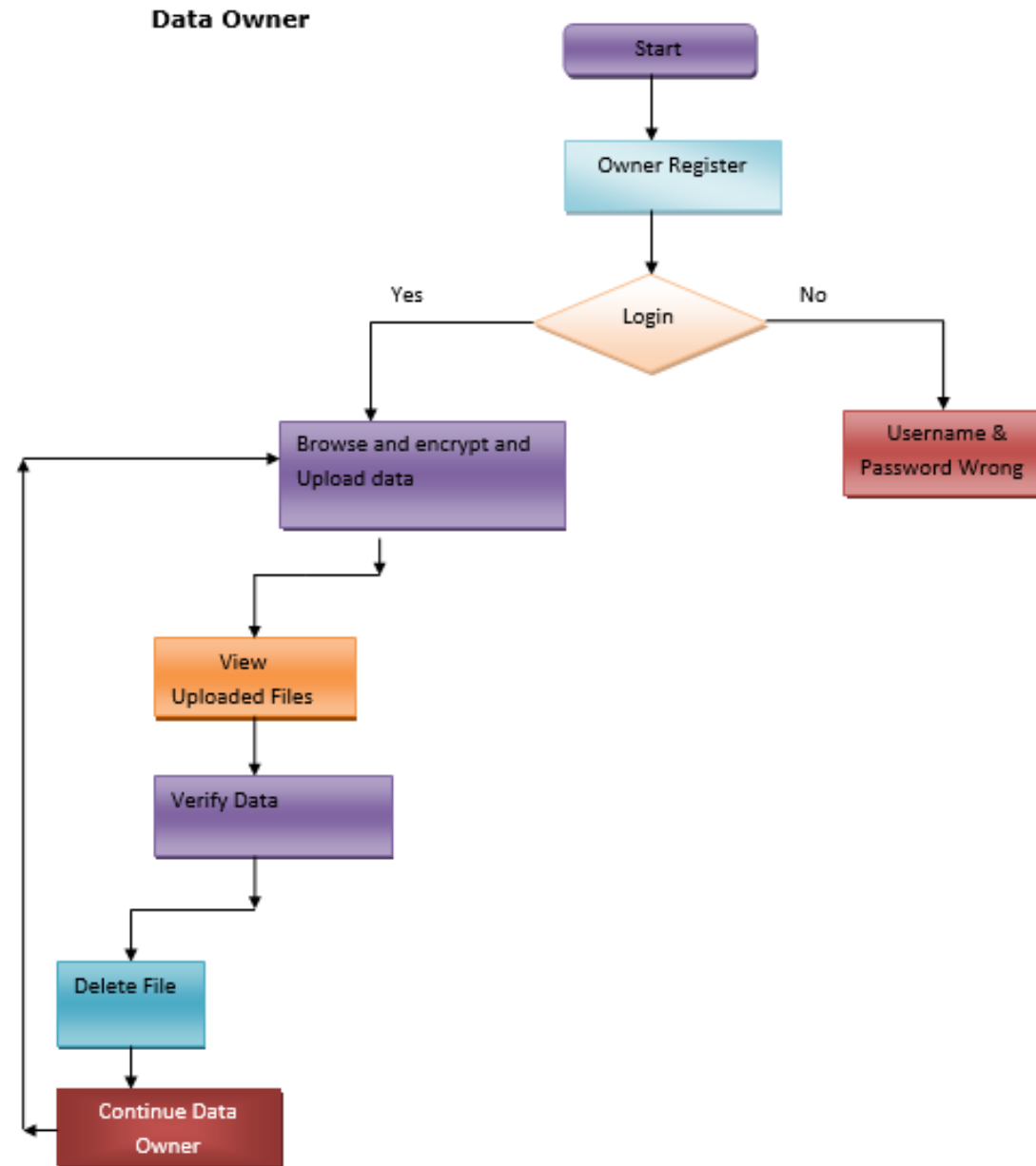
Policy data

File data

Cloud enabled data access control.



## Flow chart



# User Register InterFace

## Data User Register



Name (required)	<input type="text"/>
Password (required)	<input type="password"/>
Email Address (required)	<input type="text"/>
Mobile Number(required)	<input type="text"/>
Your Address	<input type="text"/>
DOB (required)	<input type="text"/>
Gender(required)	<input type="text" value="Male"/>
Pincode	<input type="text"/>
Location	<input type="text"/>
Select Profile Pic(required)	<input type="button" value="Choose File"/> No file chosen
<input type="button" value="Submit"/>	

Registered Successfully!!!!

[Back Home](#)

# User login Interface

Search our site:



## Menu

Home  
Cloud Data Server  
Data Owner  
End User

## End User Login



Name (required)

Password (required)

[Register](#)

Search our site:



# Welcome **hiuser** Data User Main

## End User Menu

View My Profile

Send Search Request

Search File

View Cloud Files

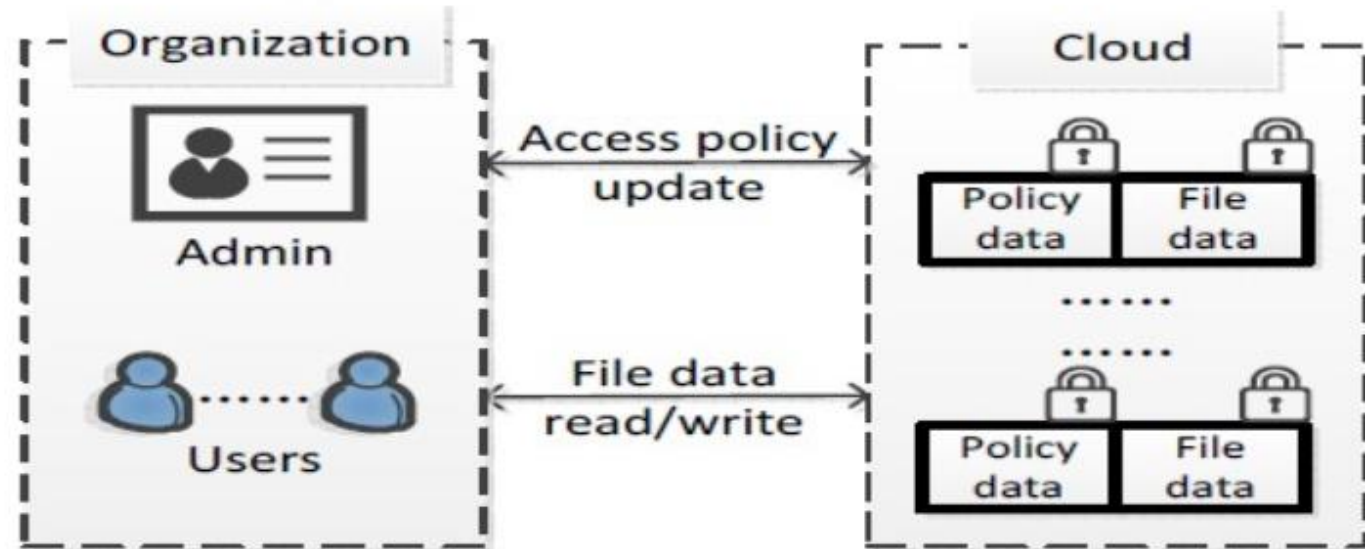
Request secret Key

Request DAC

View Access Control Responses

Download

Log Out




Cloud enabled data access control.

# View The Profile Details

## View My Profile

Owner Image	Owner Name	E-Mail	Mobile	Address	DOB	Location	Status
 Submit	hiuser	hiuser125@gmail.com	1234567890	bagaldesh	01012875	american	Authorized

# Data user can check the request search permission

Search our ste: 

Data User Menu

User Main

Log Out

Request Search Permission

Request Search and Dec Key Permission

- We sent to request to the cloud server for permission to read and write the Data

## Request Search Control & Permission

Hi Mr.hiuser ur request sent to Cloud Server

[Back](#)

### View User DAC Request & Permit

User Name	Request Date and Time	Read Data	Write Data
Kumar	18/07/2019 15:55:51	Permitted	Permitted
tmkemanju	19/07/2019 11:46:26	Permitted	Permitted
Vishnu	19/07/2019 11:52:53	Permitted	Permitted
datauser	06/05/2022 10:19:29	Permitted	Permitted
hiuser	13/05/2022 12:49:33	<u>No</u>	<u>No</u>

- We can check the status and view the Data in user interface

### View Access Control for File Secret Key

User Name	File Name	Owner Name	Req Date	Res Date	Permitted Trapdoor	Sk	Status
hiuser	panday	hi	13/05/2022 12:48:50	Waiting for Response	req	req	No

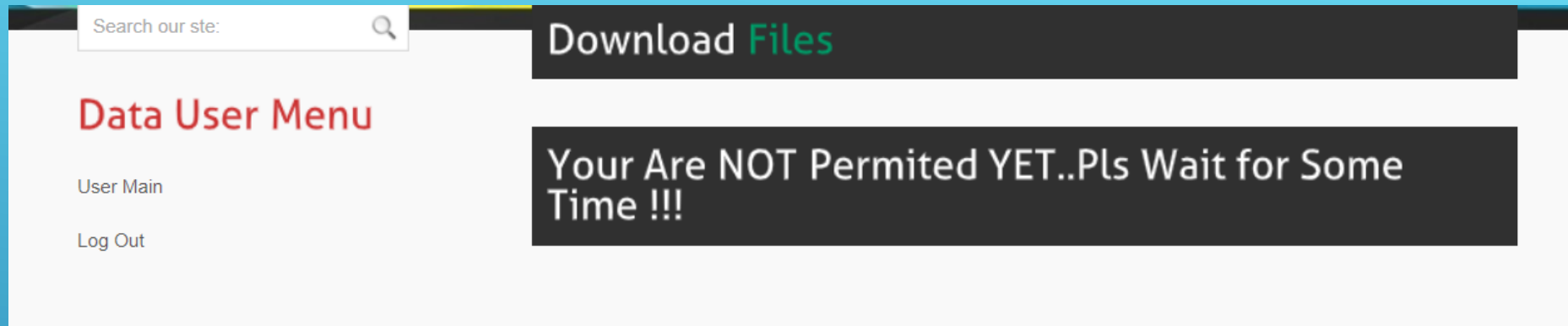
### View Access Control for Search

User Name	Date and Time	Status
hiuser	13/05/2022 12:46:13	No

[Go Back](#)



- If The User Need To Download the File.He should be send The request get permission.



- We Can check the user request and permission to Accept

View User Request & Permit							
User Name	File Name Req	Owner Name	Req Date	Res Date	Trapdoor	Sk	Status
Kumar	Android.txt	Harish	18/07/2019 18:00:43	18/07/2019 18:00:58	902d177dba1deedd45dc411b9653401a566ec36	[B@1ca5df9	<u>Yes</u>
tnksmanju	gst.txt	Manjunath	19/07/2019 11:45:55	19/07/2019 11:46:04	45e7d5d537dba0afa0b773cfd5af8032953ea369	[B@c24193	<u>Yes</u>
tnksmanju	Java.txt	Harish	19/07/2019 11:50:37	19/07/2019 11:50:49	-1ca6aa6f38a791173c2d718d2f4e70e4e053e5bd	[B@a6a21a	<u>Yes</u>
Vishnu	Android.txt	Harish	19/07/2019 11:54:18	19/07/2019 11:54:26	902d177dba1deedd45dc411b9653401a566ec36	[B@1ca5df9	<u>Yes</u>
datauser	DACA	dataowners	06/05/2022 10:18:39	06/05/2022 10:19:11	215e399eb4b1a24a8d395abb7ff4788a847c9294	[B@4e220453	<u>Yes</u>
hiuser	panday	hi	13/05/2022 12:48:50	Waiting for Response	req	req	No

# Uploading the file in cloud Server to secure the Data

Upload File

Main Menu

Data Owner Main

Log Out

Data Owner Menu

Upload

View My Files

View My Profile

Verify

Delete File

Select File :-

Choose File No file chosen

File Name :-

File will be in html documents types

Trapdoor :-

Encrypt

# Check The File was Secure or Not

## Main Menu

Data Owner Main

Log Out

## Data Owner Menu

Upload

View My Files

View My Profile

Verify

Delete File

## Verify File

File Name :-	<input type="text" value="panday"/>
	<input type="button" value="Verify"/>

## Main Menu

Data Owner Main

Log Out

## Data Owner Menu

Upload

View My Files

View My Profile

Verify

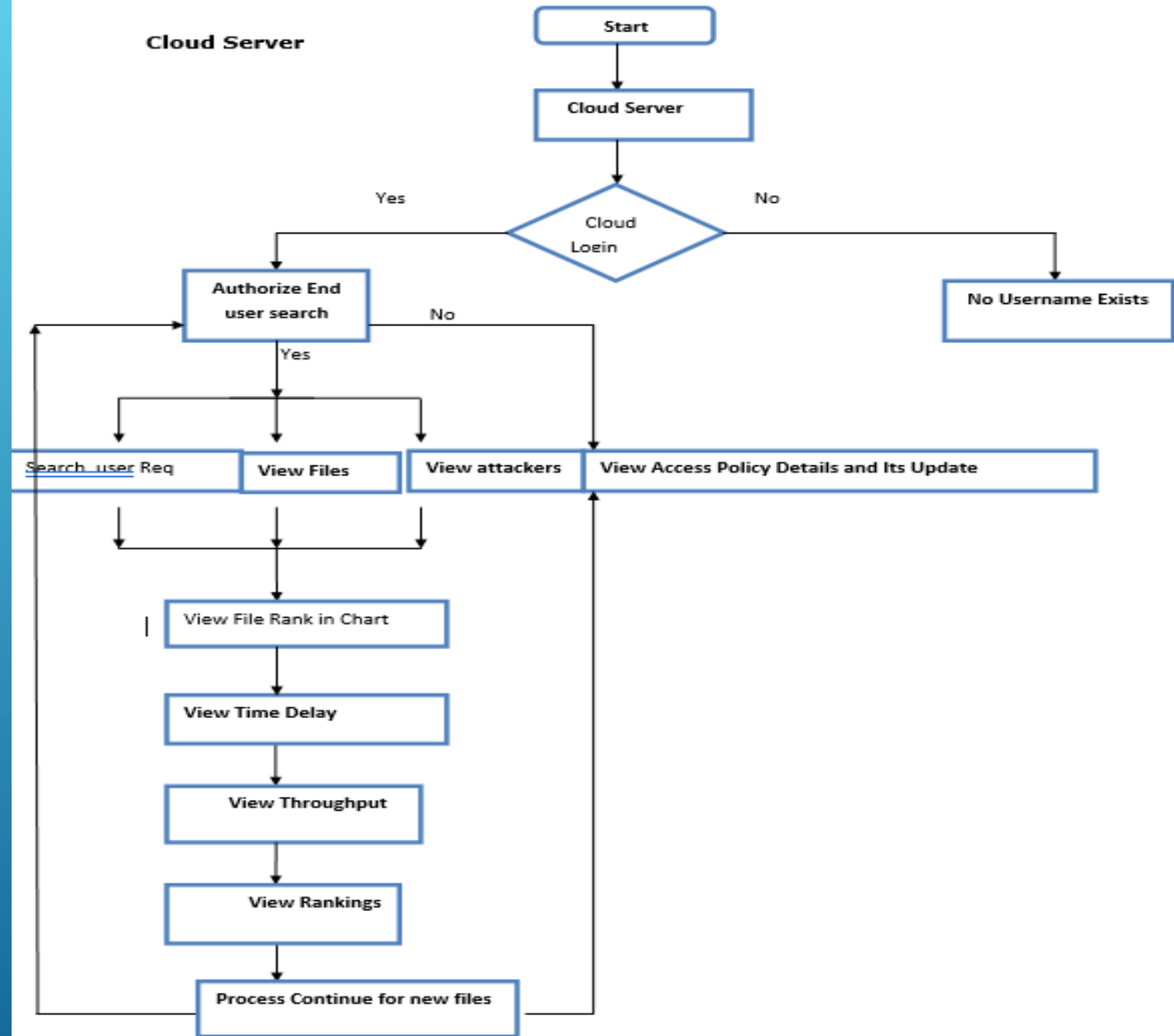
Delete File

## Verify File

panday is Secured !!!

[BACK](#)

# Flow chart



# Cloud Data Server interface

Welcome **Cloud Data Server** Main

## Cloud Server Menu

- View Owners & Authorize
- View Users & Authorize
- View Cloud Server Files
- View Data User Key Access Control Request
- View Search Access Control Request
- View Access Policy Details and Its Update
- View Transactions
- View Attackers
- View File's Rank in Chart
- View Time Delay in Chart
- View Throughput in Chart
- Log Out

The diagram illustrates the interaction between an Organization and a Cloud for data access control. The Organization side includes an Admin (represented by a person icon with a list) and Users (represented by two person icons connected by a dotted line). The Cloud side contains multiple blocks, each with Policy data and File data, both secured with padlock icons. Arrows indicate the flow of information: 'Access policy update' from the Cloud to the Admin, and 'File data read/write' from the Users to the Cloud.

```
graph LR; subgraph Organization; Admin[Admin]; Users[Users]; end; subgraph Cloud; subgraph Block1; P1[Policy data]; F1[File data]; end; subgraph Block2; P2[Policy data]; F2[File data]; end; end; Cloud -- "Access policy update" --> Admin; Users -- "File data read/write" --> Cloud;
```

Cloud enabled data access control.

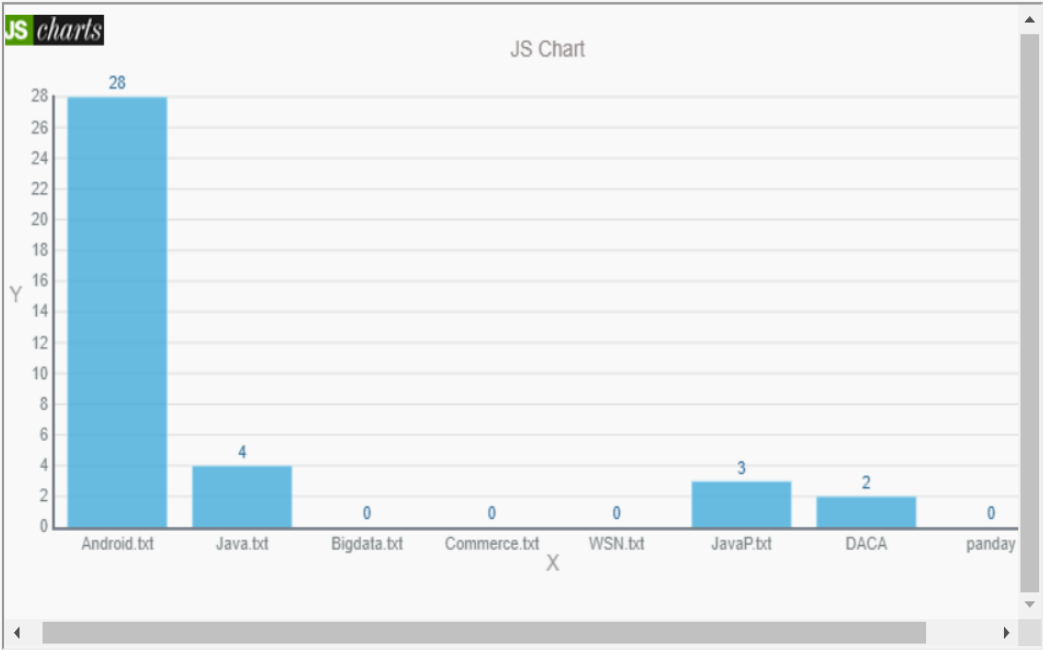
## We view the request Accept and permission to user

### View User Search Request & Permit

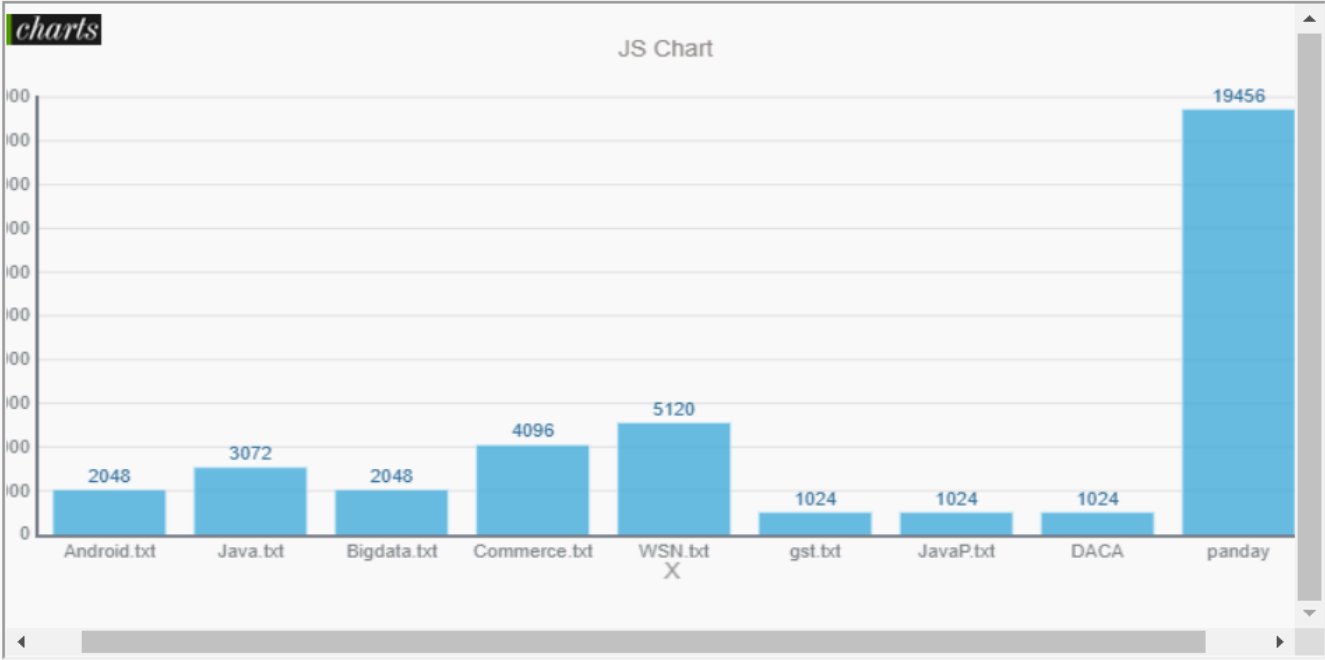
User Name	Date and Time	Status
Kumar	18/07/2019 18:02:01	<u>Yes</u>
tmksmanju	19/07/2019 11:44:36	<u>Yes</u>
Vishnu	19/07/2019 11:53:30	<u>Yes</u>
datauser	06/05/2022 10:15:03	<u>Yes</u>
hiuser	13/05/2022 12:46:13	<u>No</u>

# Chart interfacer

View All File Rank Details



View Throughput Details






## Get Request to Login the user inter face

### View User DAC Request & Permit

User Name	Request Date and Time	Read Data	Write Data
Kumar	18/07/2019 15:55:51	Permitted	Permitted
tmksmanju	19/07/2019 11:46:26	Permitted	Permitted
Vishnu	19/07/2019 11:52:53	Permitted	Permitted
datauser	06/05/2022 10:19:29	Permitted	Permitted
hiuser	13/05/2022 12:49:33	<u>No</u>	<u>No</u>

## CONCLUSION:-

We presented Crypt-DAC, a system that provides practical cryptographic enforcement of dynamic access control in the potentially untrusted cloud provider. Crypt-DAC meets its goals using three techniques. In particular, we propose to delegate the cloud to update the policy data in a privacy-preserving manner using a delegation-aware encryption strategy. We propose to avoid the expensive re-encryptions of file data at the administrator side using an adjustable onion encryption strategy. In addition, we propose a delayed de-onion encryption strategy to avoid the file reading overhead. The theoretical analysis and the performance evaluation show that Crypt-DAC achieves orders of magnitude higher efficiency in access revocations while ensuring the same security properties under the honest-but curious threat model compared with previous schemes.





**THANK YOU**

