1. Introduction

1.1 Problem Statement

Reversible Data Hiding in Encrypted Images by Reversible Image Transformation is used for increasing security of images while transferring it from a source to destination. The major challenge lies in encrypting an image so as to improve its security. Besides, to fit different image resolutions, the image must be properly merged with another image on top of it. This document proposes the implementation of an algorithm for transformation and anti-transformation of an image.

1.2 Purpose

This software requirement specification (SRS) document describes the functional and non-functional requirements of the software application of Reversible Data Hiding in Encrypted Images by Reversible Image Transformation. The working and objectives is briefly summarized followed by detailed description of the system's scope, vision, use case, features and other related requirement issues. In the project's later phases, such as system design, database design, implementation and testing, this document should be referred as functional model of the system.

1.3 Document Conventions

All system development activities should follow the final version of this document. Any discrepancy that found during in later phases should be modified subject to SRS. However, this document may be subjected to change depending on the decision of the group members.

The typographical conventions used in writing this SRS are:

- SRS main headings: Font=Times New Roman, Bold, Size=18.
- SRS headings: Font=Times New Roman, Bold, Size=18.
- SRS sub headings: Font=Times New Roman, Bold, and Size=14.
- SRS Body text: Font=Times New Roman, Size=11.
- Header & Footer – Font Size: 10, Bold & Italics, Times New Roman. The document contains header on all pages. The header is the name of the project on top left end and page number on the top right end of the page.
- Bullets are used to denote main points in the section.

1.4 Intended Audience and Reading Suggestions

The document is intended for different types of readers such as developers, administrator and the authorized users. The rest of this SRS contains an overall description, external interface requirements, system features and other non-functional requirements.

Developers and testers can go through the details mentioned from topic 2 to 5. Tester can rely on the document section 4, where each system feature is listed. Database designers will be interested on sections 2.5 and 3.

1.5 Product Scope

The main aim of this project, **'Reversible Data Hiding in Encrypted Images by Reversible Image Transformation'** is to improve the security of images. The system first accepts an image from client and then it will be encrypted with another image so as to obtain a merged image by keeping the new image over the original image. For this encryption we need to make resolution of both images same and then divide the original into four pieces of equal resolution and then rotate the four pieces in different directions using a transformation algorithm and merge new image over it. Also create a key and set access rights to users so that others can't access it. A new client will access this image using his login and authentication so that others can't access. If user has access right and if key is correct then provide the original image to client else produce a fake image.

1.6 References

[1] [IEEE Standard 181-1998]: The standard followed by the current SRS.

[2] Roger S. Pressman, "Software Engineering: A Practitioner's Approach", 7th Edition, McGraw-Hill, Singapore, 2011.

[3] Reversible Data Hiding in Encrypted Images by Reversible Image Transformation, Weiming Zhang, Hui Wang, Dongdong Hou, Nenghai Yu. 1520-9210 (c) 2016 IEEE.

2. Overall Description

2.1 Product Perspective

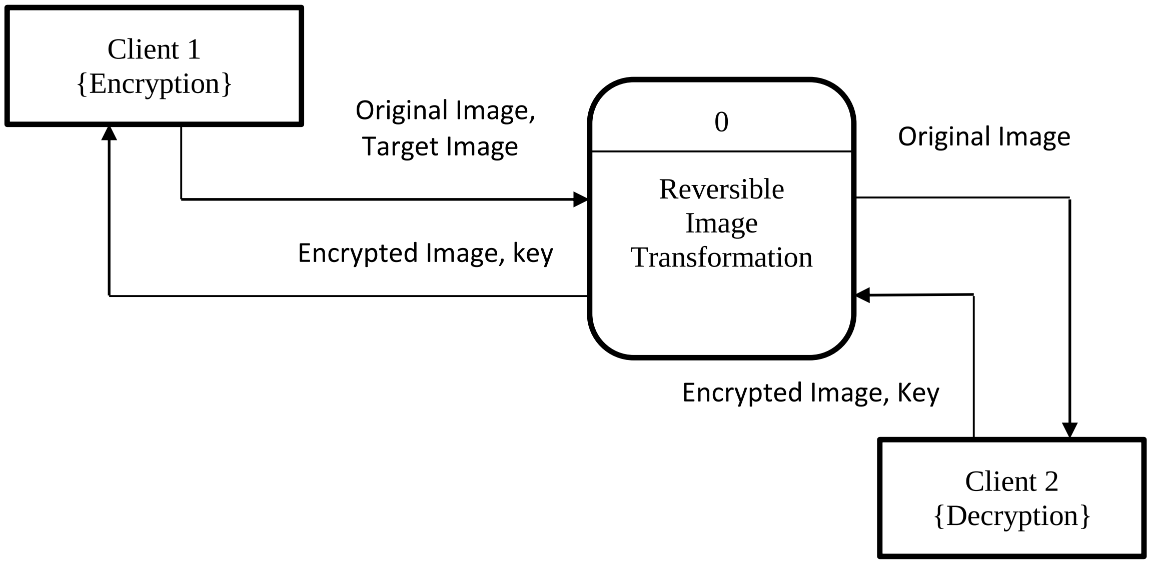
In this project, the system takes two inputs: original image and a target image. Using an image transformation algorithm, we will create the target image and a key at the server. A new client login to the server in order to access the original image. And server check his authenticity and if true then using anti transformation algorithm return the original image.

2.2 Product Functions

In this project, we are using functions such as:

- Image Acquisition
- Encryption
- Decryption
- Image Retrieval

Level 0 DFD:



2.3 User Classes and Characteristics

Generally, the users are classified into two:

- **User for Encryption**

  They can use the system to generate Encrypted image and key from input images.

- **User for Decryption**

  They can use system to generate Output image from encrypted image and key.

2.4 Operating Environment

2.4.1 Hardware Requirements:

- Processor: Pentium IV or above.
- RAM: Minimum 64 MB.
- Standard Keyboard and Mouse.

2.4.2 Software Requirements:

- Operating System: Windows XP and above.
- NetBeans IDE 8.2
- MySQL

2.5 Design and Implementation Constraints

2.5.1 Output Design: Output design involves:

- The original image send can be viewed in another client login.

2.5.2 Input Design: This involves:

- Input to the system include the original image to be transmitted.

2.5.3 Control Design: Controls provide ways:

- To encrypt the image using a transformation algorithm and retrieve using anti transformation algorithm.

2.6 Assumptions and Dependencies

There are many dependencies and assumptions associated with the software. They are:

- Encrypting the original image with a new image is the main task so that we use an encryption algorithm. It produces a key and gives access right to users who can access the original image.
- Image retrieval can be done using anti transformation algorithm and by matching the access right and key produced by a new client original image is produced else a fake image is produced.

3. External Interface Requirements

3.1 User Interfaces

The interface between user and the system include many provisions from where they can access the whole system. It contains the following options for data entry from the user:

- Original image selection from any folder
- Target image selection from any folder

3.2 Hardware Interfaces

The entire software requires a completely equipped computer system including monitor, keyboard, and other input output devices.

3.3 Software Interfaces

- The system can use Microsoft, Linux as the operating system platform. System also makes use of certain GUI tools. The system uses NetBeans IDE 8.2, MySQL.

4. System Features

The system takes original image as input also a target image chosen from user. Both images must have same resolution. First, original is divided into 4 equal parts and then rotated in 0° , 90° , 180° , 270° and target image is merged over the original image using transformation algorithm and a key is generated. Target image is saved to a database at server and privileges are given to users to access it. A new client who wants to access the image has to perform authentication and must produce the original key. And after authentication using anti transformation algorithm produce the original image to client if authentication fails produce a fake image.

4.1 Image Acquisition

4.1.1 Description and Priority

The original image must be encrypted with a key to a target image and send.

4.1.2 Functional Requirements

FREQ-1: System should accept user login.
FREQ-2: System should accept the original image.
FREQ-3: System should accept target image.

4.2. Encryption

4.2.1 Description and Priority

In order to encrypt the image, we will accept the original image from client and then using transformation algorithm we encrypt the original image to a target image and produce a key for decoding.

4.2.2 Functional Requirements

FREQ-1: System should use encryption algorithm by dividing original image and rotating it, finally merge target image over the original image.
FREQ-2: System should generate a key for decryption and assign access rights to users who can decrypt.
FREQ-3: System should assign access rights for users to decrypt and obtain original image.

4.3. Decryption

4.3.1 Description and Priority

For decryption of target image to produce original image user must provide his authorisation and then should insert target image and key along with it. After that decryption algorithm is executed.

4.3.2 Functional Requirements

FREQ-1: System should ask user for authorisation.

FREQ-2: System should accept target image and key form user for decryption.

FREQ-3: System should implement anti transformation algorithm to obtain original image.

4.4. Image Retrieval

4.4.1 Description and Priority

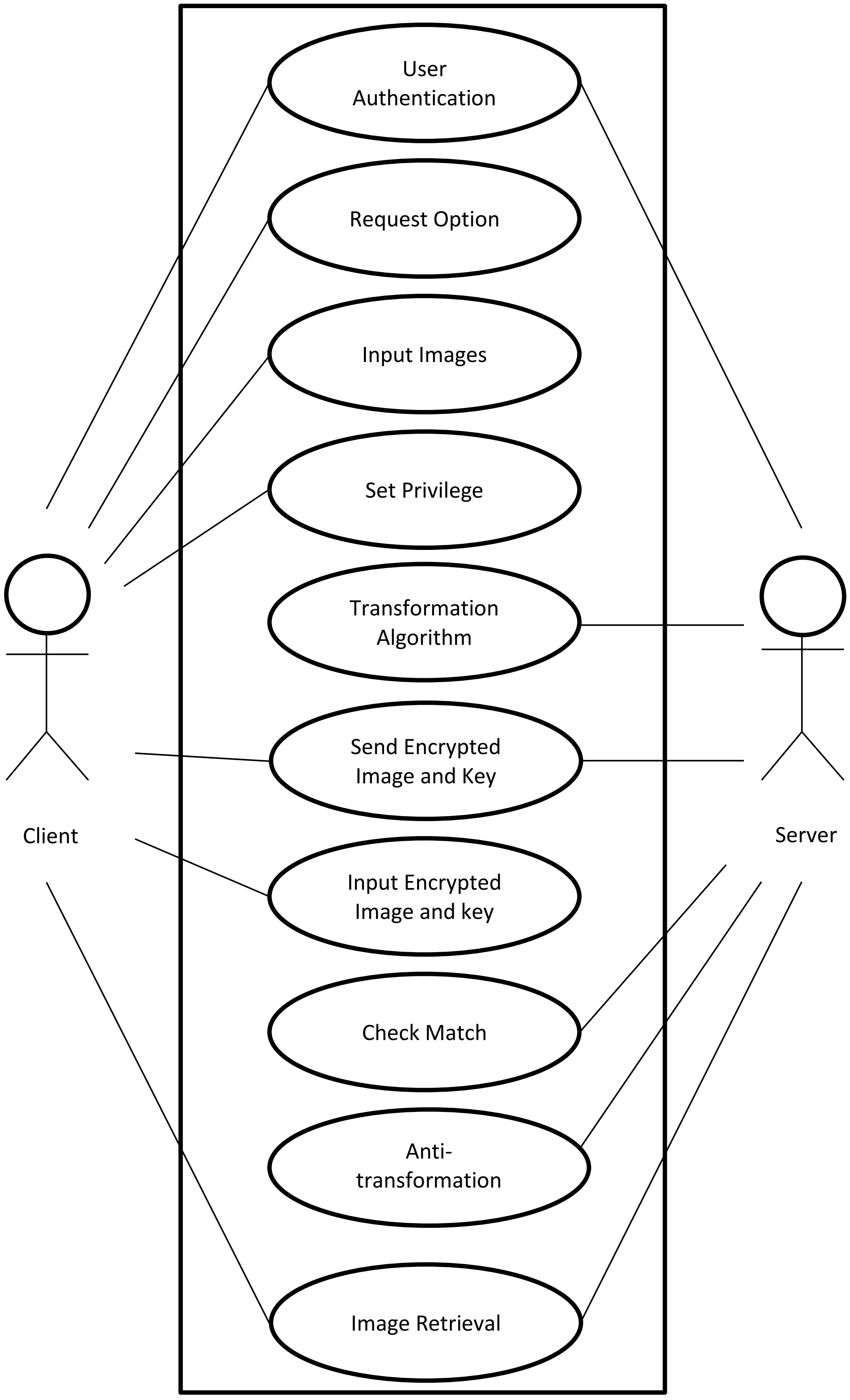
For retrieval of original image after decryption target image is taken out and then rotate original image.

4.4.2 Functional Requirements

FREQ-1: System should remove the image over the original image.

FREQ-2: System should rotate the original image so that original position is retained.

Use Case Diagram



\

5. Other Non-functional Requirements

5.1 Performance Requirements

The performance of the system lies in the way it is handled. Every user must be given proper guidance regarding how to use the system. The other factor which affects the performance is the absence of any of the suggested requirements.

5.2 Safety Requirements

To ensure the safety of the system, perform regular monitoring of the system so as to trace the proper working of the system. An administrator should be there to ensure the safety of the system. He has to be trained to handle extreme error cases.

5.3 Software Quality Attributes

1. Planned approach towards working: - The working in the system will be well planned and organized. The image will be stored properly and will help in implementing the secrecy.

2. Accuracy: - The level of accuracy in the proposed system will be higher. All operation would be done.

3. Reliability: - The reliability of the proposed system will be high due to the above stated reasons. The reason for the increased reliability of the system is that now there would be proper security in transmission of image.

4. No Redundancy: - In the proposed system utmost care would be that no information is repeated anywhere, in storage or otherwise. This would assure economic use of storage space and consistency in the data stored.

5. Easy to Operate: - The system should be easy to operate and should be such that it can be developed within a short period of time and fit in the limited budget of the user.

5.4 Business Rules

This data hiding in images finds wide application in the field of military and communication systems, e.g. military contracts, secret documents, and online image transfer.

6. Conclusion

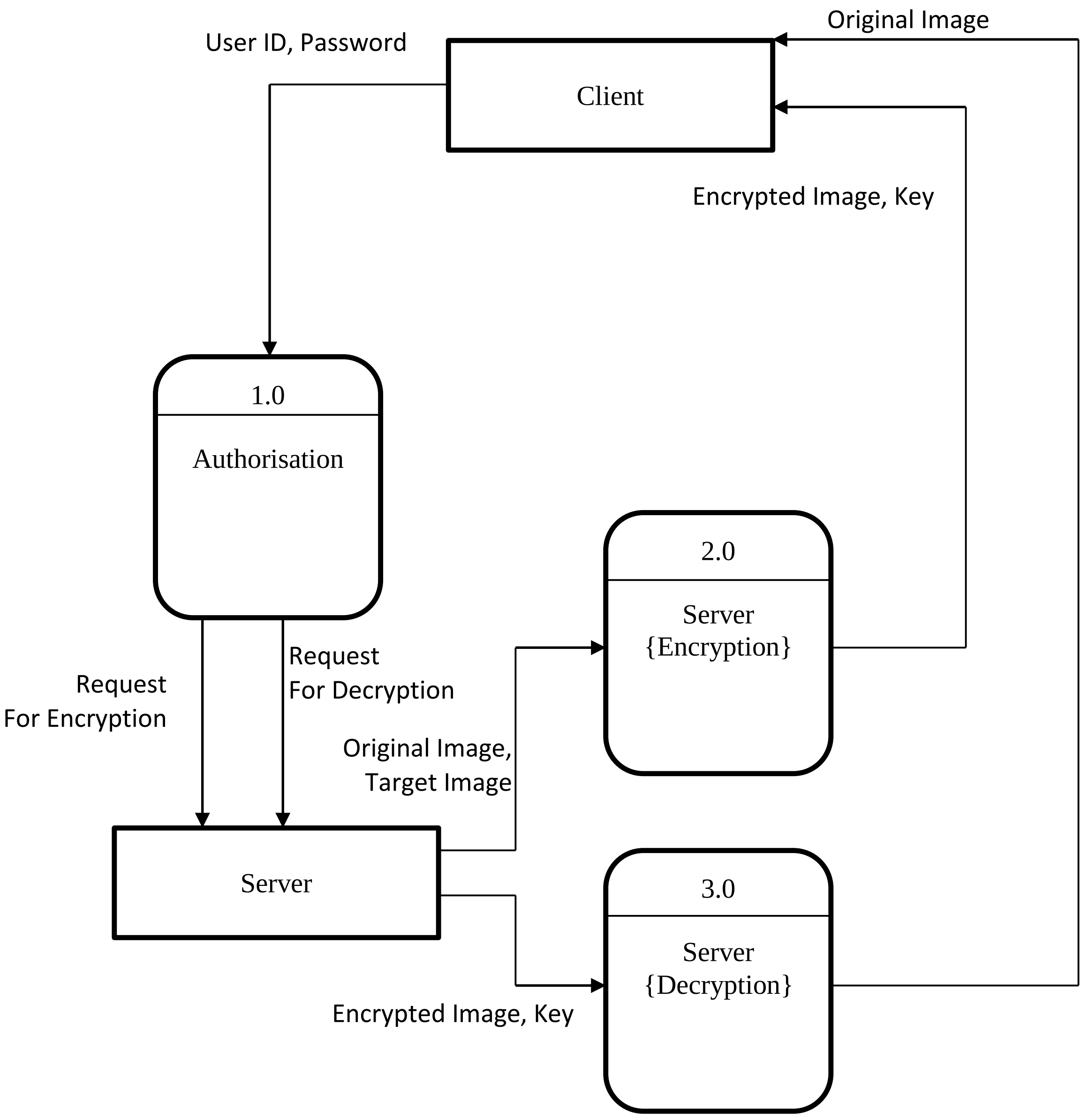
The document proposes the implementation of a model to transfer images from one user to another through a server. It generates a target image along with a key so as to improve security. Using encryption algorithm, we create a target image from the original image. Finally, we transmit the target image and the new user will have to authenticate himself. After authentication when target image and key is produced original image is retrieved using anti transformation algorithm. Hence image is safely transferred.

Appendix A: Glossary

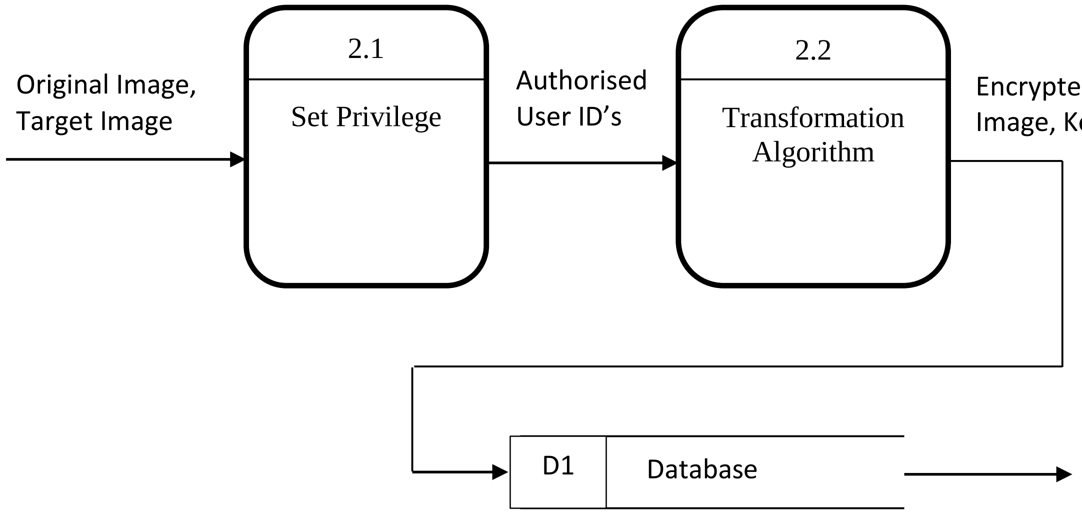
- SRS: Software Requirement Specification
- GUI: Graphical User Interface
- FREQ: Functional Requirement
- NFREQ: Non Functional Requirement

Appendix B: Analysis Models

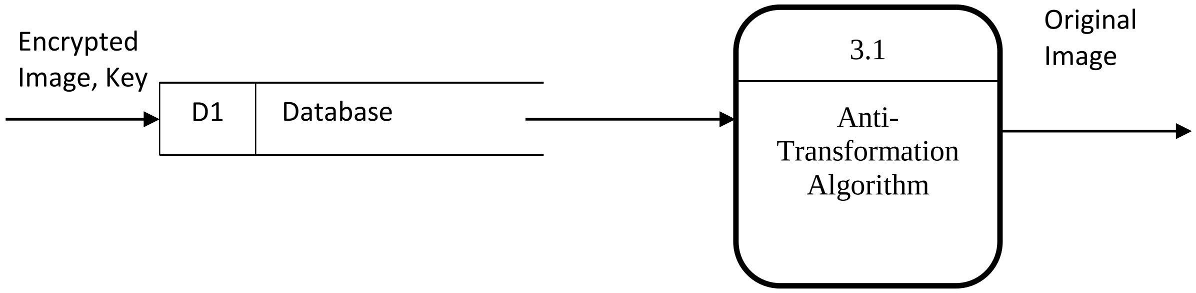
Level 1 DFD:



Level 2 DFD for Encryption:



Level 2 DFD for Decryption:


Encrypted
Image, Key
D1
Database
3.1
Anti-
Transformation
Algorithm
Original
Image