CRYPTOGRAPHY AND NETWORK SECURITY

Project report submitted
in partial fulfillment of the requirement for the Diploma
of

Network Security

By

Surbhi Anand
(2K18/CO/365)

Tanyam Singhal
(2K18/CO/371)

ABSTRACT

Electronic computers have evolved from exiguous experimental enterprises in the 1940s to prolific practical data processing systems in the 1980s. As we have come to rely on these systems to process and store data, we have also come to wonder about their ability to protect valuable data.

Data security is the science and study of methods of protecting data in computer and communication systems from unauthorized disclosure and modification. The goal of this seminar is to introduce the mathematical principles of data security and to show how these principles apply to ATM, Smart cards, e-commerce and other purposes.

Data security has evolved rapidly since 1975. Exciting developments in cryptography: public-key encryption, digital signatures, the Data Encryption Standard (DES), key safeguarding schemes, and key distribution protocols. We have developed techniques for verifying that programs do not leak confidential data or transmit classified data to users with lower security clearances. We have come to a better understanding of the theoretical and practical limitations to security.

TABLE OF CONTENTS

Table of Contents

CHAPTER 1

1.1 INTRODUCTION

Cryptography, art and science of preparing coded or protected communications intended to be intelligible only to the person possessing a key. Cryptography (Greek *kryptos*, “secret”; *graphos*, “writing”) refers both to the process or skill of communicating in or deciphering secret writings (codes, or ciphers) and to the use of codes to convert computerized data so that only a specific recipient will be able to read it using a key.

Cryptographers call an original communication the cleartext or plaintext. Once the original communication has been scrambled or enciphered, the result is known as the ciphertext or cryptogram. The enciphering process usually involves an algorithm and a key.

An encryption algorithm is a method of scrambling—a computer program or a written set of instructions. The key specifies the actual scrambling process. The original communication maybe a written or broadcast message or a set of digital data.

In its broadest sense, cryptography includes the use of concealed messages, ciphers and codes. Concealed messages, such as those hidden in otherwise innocent text and those written in invisible ink, depend for their success on being unsuspected. Once they are discovered, they frequently are easy to decipher.

Codes, in which predetermined words, numbers, or symbols represent words and phrases, are usually impossible to read without the key codebook. Cryptography also includes the use of computerized encryption to protect transmissions of data and messages.

Today most communication leaves some kind of recorded trail. For example, communications over telephone lines, including faxes and e-mail messages, produce a record of the telephone number called and the time it was called. Financial transactions, medical histories, choices of rental movies, and even food choices may be tracked by credit card receipts or insurance records.

Every time a person uses the telephone or a credit card, the telephone company or financial institution keeps a record of the number called or the transaction amount, location, and date. In the future, as telephone networks become digital, even the actual conversations may be recorded and stored.

All of this amounts to a great privacy. The ability to encrypt data, communications, and other information gives individuals the power to restore personal privacy. Cryptography is important for more than just privacy, however. Cryptography protects the world's banking systems as well. Many banks and other financial institutions conduct their business over open networks, such as the Internet. Without the ability to protect bank transactions and communications, criminals could interfere with the transactions and steal.

1.1.1 Objective

Security measures must be incorporated into computer systems whenever they are potential targets for malicious or mischievous attacks. This is especially for systems which handle financial transactions or confidential, classified or other information whose secrecy and integrity are critical. With the need to protect the integrity and privacy of information belonging to individuals and organizations, we have developed this system.
The objective of encryption algorithms is to allow the encryption of a message by one person and the decryption of the message by another person In this project, the private key is being divided into 2n+1 parts ,that is odd number of parts so that even if the intruder is able to detect some of the parts of the private key, it is computationally infeasible to detect all the 2n+1 parts That is the encrypted message can be decrypted only if all the 2n+1 parts of the private key is known. Here our objective is to unable the intruder identifies the message even if he succeeds in calculating one or some of the parts of the private key because the calculation of all 2n+1 parts of the private key is a difficult task

1.1.2 Scope

This project aims at converting the plaintext into a form unreadable by unauthorized people and hence can be readily transferred across the web and decrypted at the recipient side only by authorized people. It provides an interactive environment to encrypt, decrypt or transfer encrypted files without compromising with the integrity and privacy of critical information. In the era of wide area , open distributed systems , this system will help resolve various security issues.

1.1.3 Types of cryptography

There are many types of cryptography, including codes, steganography (hidden or secret writing), and ciphers. Codes rely on codebooks. Steganography relies on different ways to hide or disguise writing. Ciphers include both computer-generated ciphers and those created by encryption methods. The different types of ciphers depend on alphabetical, numerical, computer based or other scrambling methods.

1.1.4 Codes and Codebooks

A well-constructed code can represent phrases and entire sentences with symbols, such as five-letter groups and is often used more for economy than for secrecy. A properly constructed code can give a high degree of security, but the difficulty of printing and distributing codebooks—books of known codes—under conditions of absolute secrecy limits their use top places in which the books can be effectively guarded. In addition, the more a codebook is used, the less secure it becomes.

1.1.5 Steganography:

Steganography is a method of hiding the existence of a message using tools such as invisible ink, microscopic writing, or hiding code words within sentences of a message (such as making every fifth word in a text part of the message). Cryptographers may apply steganography to electronic communications. This application is called transmission security.

Steganography, or secret writing, seems to have originated almost as early as writing itself did. Even in ancient Egypt, where writing itself was a mystery to the average person, two distinct forms of writing were used. Hieratic or sacred writing was used for secret communication by the priests, and demotic writing was used by other literate people. The ancient Greeks and Romans, as well as other civilizations that flourished at around the same time, used forms of steganography. The invention of the first shorthand system was presumably intended as a form of secret writing.

1.1.6 Ciphers

Ease of use makes ciphers popular. There are two general types of ciphers. Substitution ciphers require a cipher alphabet to replace plaintext with other letters or symbols. Transposition ciphers use the shuffling of letters in a word to make the word incomprehensible.

Ciphers are the secret codes used to encrypt plaintext messages. Ciphers of various types have been devised, but all of them are either substitution or transposition ciphers. Computer ciphers are ciphers that are used for digital messages. Computer ciphers differ from ordinary substitution and transposition ciphers in that a computer application performs the encryption of data. The term *cryptography* is sometimes restricted to the use of ciphers or to methods involving the substitution of other letters or symbols for the original letters of a
message.

1.1.7 Computer Ciphers & Encryption

Government agencies, banks, and many corporations now routinely send a great deal of confidential information from one computer to another. Such data are usually transmitted via telephone lines or other nonprivate channels, such as the Internet.

In 1978 three American computer scientists, Ronald L. Rivest, Adi Shamir, and Leonard Adleman, who later founded the company RSA Data Security, created the Rivest-Shamir-Adleman (RSA) system. The RSA system uses two large prime numbers, p and q, multiplied to form a composite, n. The formula $n = pq$, capitalizes on the very difficult problem of factoring prime numbers.

CHAPTER – 2

2.1 REVIEW OF LITERATURE

Cryptography is probably the most important aspect of communication's security and is becoming increasingly important as a basic building block for computer security. The increased use of computer and communications system by industry has increased the risk of theft of proprietary information although these threats may require a variety of counter measures. Encryption is a primary method of protecting valuable electronic information. Encryption is the process of encoding a message in such a way as to hide it's contents. Modern cryptography includes several secure algorithms for encrypting and decrypting messages. They are all based on the use of secrets called key. A cryptography key is a parameter used in an encryption algorithm in such a way that the encryption can not be reversed without the knowledge of the key. Terms used in cryptography are as follows:

- **Plain text**: original message is known as plain text.
- **Cipher text**: coded message is known as cipher text.
- **Encryption**: the process of converting the plain text to cipher text is known as encryption.
- **Decryption**: the process of restoring the plain text from the cipher text is known as decryption.

Cryptographic systems are characterised along three independent dimensions -

The type of operation used for transforming plain text to cipher text:

 Substitution Technique: A substitution technique is one in which letters of plain text are replaced by other letters or by numbers or by symbols.

Transposition Technique: In this we perform some sort of permutation on the plaintext letters.

➢ **The number of keys used:**

 Symmetric encryption: When both the sender and receiver use the same key for encrypting the plaintext and decrypting the ciphertext to plaintext, it is called symmetric encryption.

 Asymmetric encryption: When the sender use the public key of the receiver that has already been published by the recipient to encrypt the plaintext to cipher text and the receiver uses its private key to decrypt the cipher text to plaintext, it is called asymmetric encryption.

- **The way in which plain text is proccessed :**

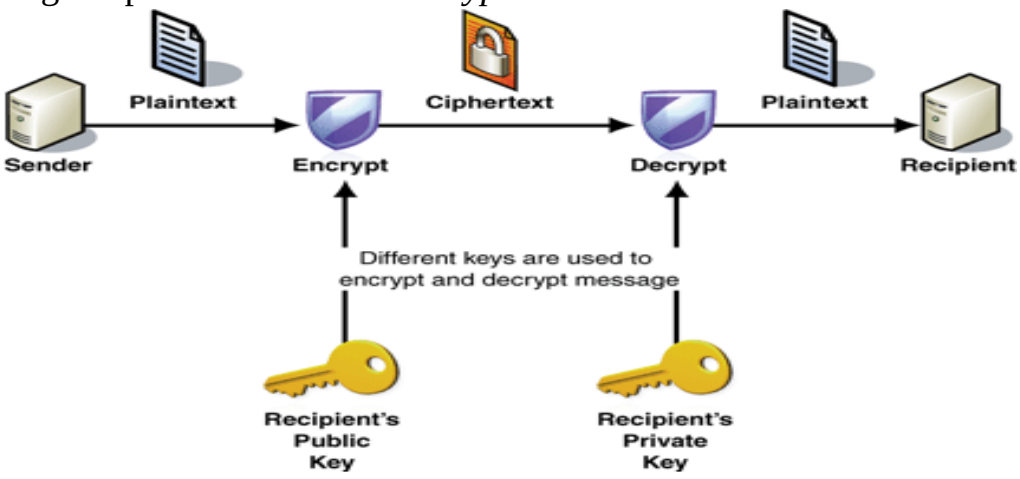
 Block Cipher : It is one in which a block of plaintext is treated as a whole and is used to produce a cipher block of equal length.

 Stream Cipher : It is one in that encrypts a digital data stream a bit or one byte at a time.

Three basic building blocks are used:
Encryption is used to provide confidentiality, can provide authentication and integrity protection Digital signatures are used to provide authentication, integrity protection, and nonrepudiation. Checksums/hash algorithms are used to provide integrity protection, can provide authentication One or more security mechanisms are combined to provide a security service.

Encryption and decryption

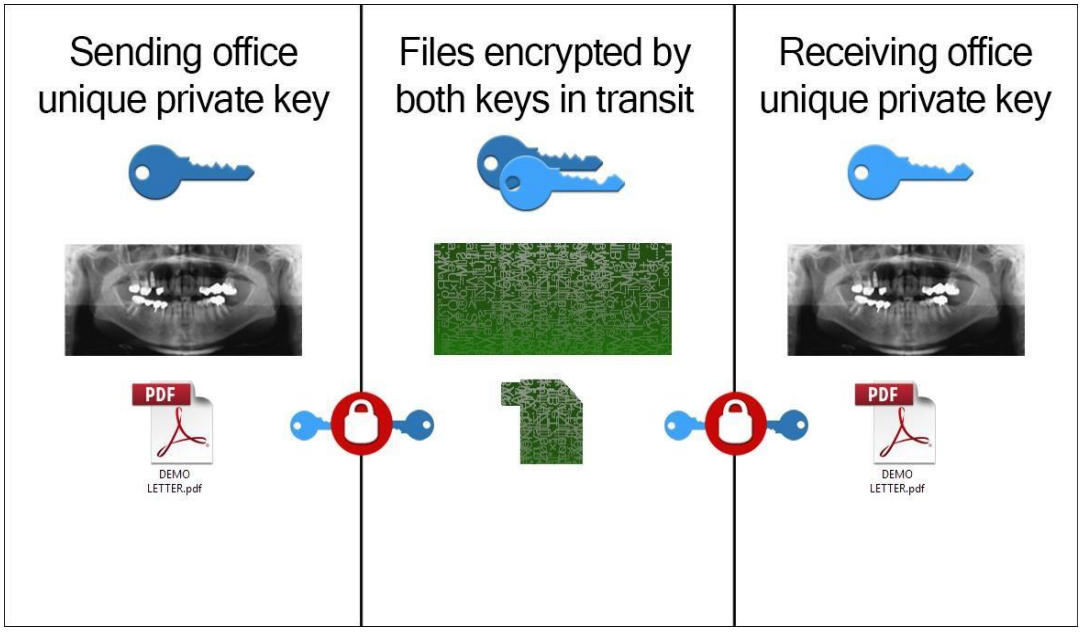
Data that can be read and understood without any special measures is called *plaintext* or *cleartext*. The method of disguising plaintext in such a way as to hide its substance is called *encryption.* Encrypting plaintext results in unreadable gibberish called *ciphertext*. You use encryption to ensure that information is hidden from anyone for whom it is not intended, even those who can see the encrypted data. The process of reverting ciphertext to its original plaintext is called *decryption*.



Public key cryptography is an asymmetric scheme that uses a *pair* of keys for encryption: a *public key*, which encrypts data, and a corresponding *private,* or *secret key* for decryption. You publish your public key to the world while keeping your private key secret. Anyone with a copy of your public key can then encrypt information that only you can read. Even people you have never met.

It is computationally infeasible to deduce the private key from the public key.Anyone who has a public key can encrypt information but cannot decrypt it. Only the person who has the corresponding private key can decrypt the Information.

Private key cryptography



CHAPTER 3

Existing System and Problem Statement

At present, the Elgamal encryption algorithm works by sending data to the receiver who has just one private key to decrypt the data .The entire process is as follows :

Key generation : The receiver who wishes to get message, chooses a large prime number p, a random number g which is also prime and less than the prime number initially chosen and a random integer x from 0 to (p-1). He then calculates

$$y=gx \bmod p$$

The public key of the sender is (p , g , y) and his private key is x.

Encryption by the sender: The sender generates an integer k lying between 0 to (p-1). He then calculates

$$r = g^k \bmod p \text{ and}$$
$$t = (y^k M) \bmod p$$

and transmits (r, t) as the encrypted message.

Decryption of the cipher text: The receiver with his private key calculates $t.r^{-x}$ which gives the plain text But in this algorithm as there is just one private key, it can be guessed by any intruder and is thus not reliable.

Proposed System

In this project we are modifying the existing Elgamal encryption algorithm by dividing the private key and assigning them to 2n+1 authorized receivers individually. The persons will be able to decrypt the message received from the sender only if they are together, separately this operation being impossible for them. It has the following operations:

Key generation: A large prime number p and a random number g which is prime and less than the initially chosen prime number is chosen. Then after from {0,…,p-1} there are chosen the elements x1,x2,…,x2n+1, preferably distinct , then there are being calculated y1=gx1 mod p,y2=gx2 mod p,…, y2n+1=gx2n+1 mod p .The public key is {p,g,y1,y2,…….,y2n+1} and the private key consists of {x1,x2,…,x2n+1}.

Encryption of a message: The sender encrypts message m knowing the public key as follows:

He chooses a random element k from {0 …, p-1} and calculates c1=gk mod p,c21=m.x1k mod p, c22=m.x2k mod p,… , c22n+1=m.x2n+1k mod p, c2= c21. c23. c25. c27…./c22. c24. c26…. then sends the encrypted message (c1, c2) to the recipient.

Decryption of the message: In order to decrypt the message (c1, c2),the receiver use p and the private keys {x1}, {x2},…..,{x2n+1} respectively, computing together

c2. c1x2. c1x4. c1x6 …/ c1x1. c1x3. c1x5. c1x7…= (c21. c23. c25. c27…/ c22. c24. c26…)(c1x2. c1x4. c1x6…/ c1x1. c1x3.c1x5. c1x7…) = (m .y1k. m . y3k. m . y5k. m . y7k…/ m .y2k. m . y4k. m . y6k…)(c1x2. c1x4. c1x6…/ c1x1. c1x3. c1x5. c1x7…) = m.

ADVANTAGE OF PROPOSED SYSTEM

The encrypted message can be decrypted only if all the 2n+1 parts of the private key is known. Thus, the intruder is unable to identify the message even if he succeeds in calculating one or some of the parts of the private key because the calculation of all 2n+1 parts of the private key is a difficult task and thus it is more secure.

CHAPTER - 4

SYSTEM ANALYSIS & PLANNING

Systems analysis and design refers to the process of examining a business situation with the intent of improving it through better procedures and methods. Systems development can generally be thought of as having two major components: Systems Analysis and Systems Design. Systems design is the process of planning a new system or replace or complement an existing system. But before this planning can be done, we must thoroughly understand the existing system and determine how computers can best be used to make its operation more effective. Systems analysis, then, is the process of gathering and interpreting facts, diagnosing problems and using the information to recommend improvement to the system.

Requirement Analysis

Requirements analysis in systems engineering and software engineering, encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product, taking account of the possibly conflicting requirements of the various stakeholders, such as beneficiaries or users. Requirements analysis is critical to the success of a development project. Requirements must be documented, actionable, measurable, testable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design.
Requirements are a description of how a system should behave or a description of system properties or attributes. It can alternatively be a statement of what an application is expected to do. The Software Requirements Analysis Process covers the complex task of eliciting and documenting the requirements of all these users, modeling and analyzing these requirements and documenting them as a basis for system design.

Steps in Requirement Analysis Process

- Fix system boundaries
- Identify the customer
- Requirements elicitation
- Requirements Analysis Process
- Requirements Specification
- Requirements Management

Requirement Analysis Process

The following are the steps for requirement analysis process are:

Brainstorming Session: Brainstorming is a group creativity technique designed to generate a large number of ideas for the solution of a problem Although brainstorming has become a popular group technique, when applied in a traditional group setting, researchers have not found evidence of its effectiveness for enhancing either quantity or quality of ideas generated. Because of such problems as distraction, social loafing, evaluation apprehension, and production blocking, conventional brainstorming groups are little more effective than other types of groups, and they are actually less effective than individuals working independently. There are four basic rules in brainstorming. These are intended to reduce social inhibitions among groups‟ members, stimulate idea generation, and increase overall creativity of the group.

- **Focus on quantity**: This rule is a means of enhancing divergent production, aiming to facilitate problem solving through the maxim, quantity breeds quality. The assumption is that the greater the number of ideas generated, the greater the chance of producing a radical and effective solution.
- **Withhold criticism**: In brainstorming, criticism of ideas generated should be put 'on hold'. Instead, participants should focus on extending or adding to ideas, reserving criticism for a later 'critical stage' of the process. By suspending judgment, participants will feel free to generate unusual ideas.

Welcome unusual ideas: To get a good and long list of ideas, unusual ideas are welcomed. They can be generated by looking from new perspectives and suspending assumptions. These new ways of thinking may provide better solutions.

 Combine and improve ideas: Good ideas may be combined to form a single better good idea, as suggested by the slogan "1+1=3". It is believed to stimulate the building of ideas by a process of association.

SRS Document: A Software Requirements Specification (SRS) is a complete description of the behaviour of the system to be developed. It includes a set of use cases that describe all the interactions the users will have with the software. In addition to use cases, the SRS also contains non-functional (or supplementary) requirements. Non-functional requirements are requirements which impose constraints on the design or implementation (such as performance requirements, quality standards, or design constraints).Goals of SRS are:

 It provides feedback to the customer. An SRS is the customer's assurance that the development organization understands the issues or problems to be solved and the software behaviour necessary to address those problems.

 It decomposes the problem into component parts. The simple act of writing down software requirements in a well-designed format organizes information, places borders around the problem, solidifies ideas, and helps break down the problem into its component parts in an orderly fashion.

 It serves as an input to the design specification. Therefore, the SRS must contain sufficient detail in the functional system requirements so that a design solution can be devised.

Feasibility Study

Eight steps are involved in the feasibility analysis. They are:

 Form a project team and appoint a project leader.

 Prepare system flowcharts.

 Enumerate potential proposed systems

- Define and identify characteristics of proposed system.

 Determine and evaluate performance and cost effectiveness of each proposed system.

 Weight system performance and cost data.

 Select the best proposed system.

 Prepare and report final project directive to management.

Economic Feasibility

Economic analysis is the most frequently used technique for evaluating the effectiveness of a proposed system. More commonly known as cost / benefit analysis; in this procedure we determine the benefits and savings that are expected from a proposed system and compare them with costs. We found the benefits outweigh the costs; we take a decision to design and implement the new proposed system.

Technical Feasibility

This is concerned with specifying equipment and software that will successfully satisfy the user requirement. The technical needs of the system may vary considerably but might include the facility to produce outputs in a given time, response time under certain conditions, ability to process a certain volume of transaction at a speed, facility to communicate data to distant location.

Behavioural Feasibility

People are inherently resistant to change, and computers have been known to facilitate change. An estimate should be made of how strong a reaction the user staff is likely to have toward the development of a computerized system. It is common knowledge that computer installations have something to do with turnover, transfers, retraining, and change in employee job status. Therefore, it is understandable that the introduction of a candidate system requires special effort to educate, sell, and train the staff on new ways of conducting business.

Time feasibility

Time feasibility is a determination of whether a proposed project can be implemented fully within a stipulated time frame. If a project takes too much time it is likely to be rejected.

System Planning

Before any project development system planning is the crucial task in order to perform systematic development of the project. System planning is generally taken to collect the information from the existing system so that the system being implemented can work in favor of the user. Interview, questionnaire and observation collect facts.These techniques are not mutually exclusive indeed in practice more than one technique will be employed to establish the facts.

PERT Chart

A PERT chart is a graphic representation of a project's schedule, showing the sequence of tasks, which tasks can be performed simultaneously, and the critical path of tasks that must be completed on time for the project to meet its completion deadline.

The chart can be constructed with a variety of attributes, such as earliest and latest start dates for each task, earliest and latest finish dates for each task, and slack time between tasks.

A PERT chart can document an entire project or a key phase of a project.

Tasks can also branch out and travel their own paths re-joining the main path at some later point. Any milestones such as points of review or completion is indicated as well.

CHAPTER – 5

SYSTEM DESIGN

Key Generation:
This module generates keys. The keys can be generated randomly or it can be specified by the user. If the keys are being generated randomly,the user does not need to check if the number is prime or not. But if the keys are entered manually , the user needs to be sure of the number to be prime otherwise an error message will be displayed every time the user enters a wrong value. When the keys are entered manually , the test for prime number is done by **Modules and Their Description**
There are following five modules in our project –

- GUI(Graphical User Interface)
- Key generation
- Encryption
- Decryption
- Data transfer

Now we describe each of the modules:

1.GUI (Graphical User Interface):
GUI provides an efficient way through which user can interact with the system and works accordingly to fulfill the task .Through GUI user can select the file that is to be encrypted or decrypted.User can generate keys randomly or enter manually.At every window a “back” button is provide that lets the user to go back to the previous window. Thus it provides a user friendly environment Millar Rabin algorithm.

3.Encryption:
The sender chooses a random element k from {0,…,p-1} and calculates c1=gk mod p, c21=m.x1k mod p, c22=m.x2k mod p,…, c22n+1=m.x2n+1k mod p, c2= c21. c23. c25. c27…./ c22. c24. c26…. then
sends the encrypted message (c1, c2) to the recipient.Thus the size of the encrypted message is double of the plaintext as each charater is represented by two values.

4. Decryption:

In order to decrypt the message (c1, c2), receiver1 , receiver2 ,…, receiver2n+1 are using q and the private keys {x1}, {x2},…..,{x2n+1} respectively, computing together c2. c1x2. c1x4. c1x6 …/ c1x1. c1x3. c1x5. c1x7…

5. Data Transfer:

This module will be used to transfer encrypted files from one system to the other. The other system should keep waiting for the file if it wants to receive a file or else a connection failure occurs. The IP address of the destination needs to be mentioned correctly.

DESIGN

Software design is a process of problem solving and planning for a software solution. After the purpose and specifications of software are determined, software developers will design or employ designers to develop a plan for a solution. It includes low-level component and algorithm implementation issues as well as the architectural view. Software design can be considered as putting solution to the problem(s) in hand using the available capabilities. Hence the main difference between Software analysis and design is that the output of the analysis of a software problem will be smaller problems to solve and it should not deviate so much even if it is conducted by different team members or even by entirely different groups.

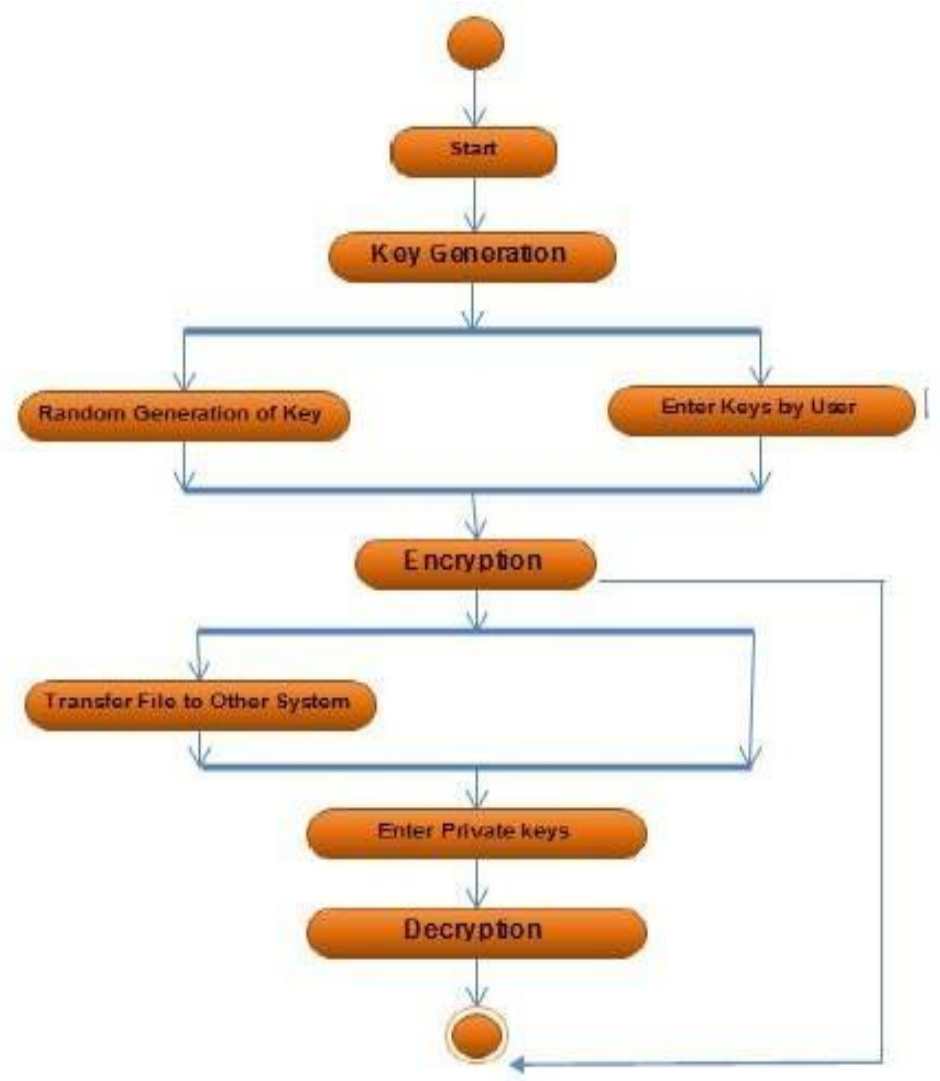
But since design depends on the capabilities, we can have different designs for the same problem depending on the capabilities of the environment that will host the solution (whether it is some OS, web, mobile or even the new cloud computing paradigm). The solution will depend also on the used development environment (Whether you build a solution from scratch or using reliable frameworks or at least implement some suitable design patterns).

ACTIVITY DIAGRAM

Activity diagrams are a loosely defined diagram technique for showing workflows of stepwise activities and actions, with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step

workflows of components in a system. An activity diagram shows the overall flow of control. In SysML the activity diagram has been extended to indicate flows among steps that convey physical element (e.g., gasoline) or

energy (e.g., torque, pressure). In UML 1.x, an activity diagram is a variation of the UML State diagram in which the "states" represent activities, and the transitions represent the completion of those activities. Activity diagrams are typically used for business process modeling. They consist of initial node, activity final node, activities. The starting point of the diagram is the initial node, and the activity final node is the ending.



Activity Diagram

CHAPTER 6

Methodology Adopted: Iterative Model

An iterative lifecycle model does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which can then be reviewed in order to identify further requirements. This process is then repeated, producing a new version of the software for each cycle of the model. Consider an iterative lifecycle model which consists of repeating the following four phases in sequence:

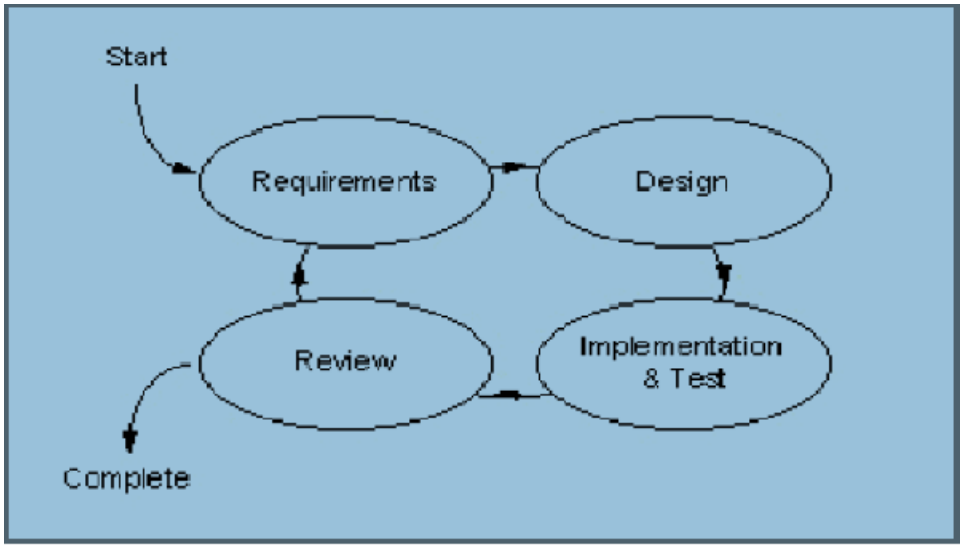


Figure 5.1: Iterative Process

A Requirements phase in which the requirements for the software are gathered and analyzed. Iteration should eventually result in a requirements phase that produces a complete and final specification of requirements.A Design phase, in which a software solution to meet the requirements is designed. This may be a new design or an extension of earlier design.

An Implementation and Test phase, when the software is coded, integrated and tested.A Review phase, in which the software is evaluated, the current requirements are reviewed, and changes and addition to the system proposed. For each cycle of the model, a decision has to be made as to whether the software produced by the cycle will be discarded, or kept as a starting point for the next cycle (sometimes referred to as incremental prototyping).

Eventually a point will be reached where the requirements are complete and the software can be delivered, or it becomes impossible to enhance the software as required, and a fresh start has to be made.

The iterative lifecycle model can be likened to producing software by successive approximation. Drawing an analogy with mathematical methods that use successive approximation to arrive at a final solution, the benefit of such methods depends on how rapidly they converge on a solution.

The key to successful use of an iterative software development lifecycle is rigorous validation of requirements, and verification (including testing) of each version of the software against those requirements within each cycle of the model. The first three phases of the example iterative model is in fact an abbreviated form of a sequential V or waterfall lifecycle model.

Each cycle of the model produces software that requires testing at the unit level, for software integration, for system integration and for acceptance. As the software evolves through successive cycles, tests have to be repeated and extended to verify each version of the software.

System Implementation

The purpose of System Implementation can be summarized as follows: making the new system available to a prepared set of users (the deployment) and positioning on-going support and maintenance of the system within the Performing Organization (the transition).

At a finer level of detail, deploying the system consists of executing all steps necessary to educate the consumers on the use of the new system, placing the newly developed system into production, confirming that all data required at the start of operations is available and accurate, and validating that business functions that interact with the system are functioning properly.

Transitioning the system support responsibilities involves changing from a system *development* to a system support and maintenance mode of operation, with ownership of the new system moving from the Project Team to the Performing Organization.

A key difference between System Implementation and all other phases of the lifecycle is that all project activities up to this point have been performed in safe, protected, and secure environments, where project issues that arise have little or no impact on day-to-day business operations. Once the system goes live, however, this is no longer the case. Any miscues at this point will almost certainly translate into direct operational and/or financial impacts on the Performing Organization. It is through the careful planning, execution, and management of System Implementation active ties that Project Team can minimize the likelihood of these occurrences, and determine appropriate contingency plans in the event of a problem.

HARDWARE AND SOFTWARE REQUIREMENTS

We are making a project on security for which we have used Java Swings and the system requirement are as follows.

Hardware Requirements:

- Intel Pentium IV processor
- RAM size 256 MB
- Hard disk 1 GB

Software Requirements:

- Windows XP
- JDK 1.6.0

CHAPTER – 7

C code on encryption and decryption:

```
#include <stdio.h>
#include<string.h>
void encrpyt(char password[],int key)
{
   unsigned int i;
   for(i=0;i<strlen(password);i++)
   {
      password[i]=password[i]-key;
   }
}
void decrypt(char password[],int key)
{
   unsigned int i;
   for(i=0;i<strlen(password);++i)
   {
   password[i]=password[i]+key;
   }
}
int main()
{
   char password[20];
   printf("enter password");
   scanf("%s",password);
   printf("password=%sn",password);
   encrypt(password,0XFACA);
   printf("encrypted value=%s",password);
   decrypt(password,0XFACA);
   printf("decrypted value=%s",password);
   return 0;
}
```

SUMMARY & CONCLUSIONS

Cryptology presents a difficulty not found in normal academic disciplines: the need for the proper interaction of cryptography and cryptanalysis. This arises out of the fact that in the absence of real communications requirements, it is easy to propose a system that appears unbreakable. Many academic designs are so complex that the would–be cryptanalyst doesn't know where to start; exposing flaws in these designs is far harder than designing them in the first place. The result is that the competitive process, which is one strong motivation in academic research, cannot take hold.

Many applications are useful in real-time and daily life that are implemented by cryptography through implicit or explicit concept of it. For example banking system, ATM cards, Smart cards, Magnetic strip technology, National Security Agency(NSA) to trace information through RADAR and with well-equipped material, E-commerce, Economics, business information, operating systems, databases and finally in System Protection.

In this way Cryptography has many roles and many applications.

Network Security is a branch of computer science that involves in securing a computer network and network infrastructure devices to prevent unauthorized access, data theft, network misuse, device and data modification. Another function of Network Security is in preventing DoS (Denial of Service) attacks and assuring continuous service for legitimate network users. Network Security involves proactive defence methods and mechanisms to protect data, network and network devices from external and internal threats.

REFERENCES

1.Microsoft Encarta encyclopaedia

2.Cryptography Encyclopaedia

3.Cryptography and Network security by William Stallings.

4.Computer-security and Cryptography by Alan konheim, ACM portal.

5.Cryptography & Data security by Denning, Amazon.com

6. Bloom, Jefrey A., Ingemar J. Cox, Ton Kalker, Jean-Paul M.G. Linnartz, Matthew L. Miller C., and Brendan S. Traw (1999). "Copy protection for DVD video." Proceedings of the IEEE,87(7), 1267–1276.

7.Applied Cryptography and Data security by Prof. Christof Paar