

**Tribhuvan University**

# Institute of Science and Technology

**A Final Year Project Report**

**on**

# “Text File Sharing with Digital Signature System within Intranet”

**Submitted to:**

**Department of Computer Science and Information Technology**

## Samriddhi College

### In Partial Fulfillment of the Requirements

**For the Bachelor’s Degree in Computer Science and Information Technology**

**Submitted by:**

Aayush Bhattarai (TU Roll No.: 16133)

Deepesh Oli (TU Roll No.: 16147)

Parajanya Shrestha(TU Roll No.: 16152)

# ACKNOWLEDGEMENT

The completion of the project would not have been possible without the proper guidance and assistance of every individual supporting the project directly or indirectly. We would like to thank our respected supervisor Mr.Bikash Balami, for supervising us in this project and clearing our doubts and problems when we faced them. The problems were tackled by providing deeper knowledge on various aspects of the project.

We would like to express our sincere gratitude to our teacher Mr. Mohan Bhandari, for providing us with a valuable and necessary guideline to bring up with a good project. Also, we are glad to express our sincere gratitude to the Department of Computer Science, College Administration for motivating us to bring forth our project. Likewise, all the results that we care for now would not have been successful without the involvement of our respected sir, Mr.Lok Nath Regmi. All his experience and advice helped us to complete the project effectively on time. Moreover, we would like to thank all the friends and teachers who helped us by contributing their suggestions and ideas.

Lastly, we would like to express our sincere thanks to Samriddhi College for providing us the opportunities and space to work on our ideas into a valuable project. Also, we thank the department for teaching us the value of teamwork and co-operation.

**Abstract**

The security of information available to an organization was primarily provided through physical and administrative means. For example : Rugged file cabinets with a key lock were used for storing sensitive documents and personnel screening procedures were employed during the hiring process.

With the introduction of the computer, the need for automated tools for protecting files and other information stored on the computer became evident. This is especially the case for a shared system and the need is even more acute for a network. Computer networks were primarily used by university researches for sending e-mail, and by corporate employees for sharing printers. Under these conditions, security was not given much attention. Today, since the world is going global, and trillions of data are transferred daily across networks, security is looming on the horizon as a potentially massive problem. The generic name for the collection of tools designed to protect data and to thwart hackers is Computer Security .

**Keywords:***Java, Swing, Sockets, RSA, Hashing, etc.*

**CHAPTER 1**

# INTRODUCTION

## 1.1 Introduction

Digital signature is an electronic signature used to authenticate digitally transferred data and to ensure that the content of the message or the document sent has not been tempered. It validates the authenticity, and integrity of a message or document. It is the same as a handwritten signature, seal, or stamp. It is widely used to verify a digital message, financial documents, identity cards, etc.

The digital signature provides three main characters for the data sent namely authentication, integrity and non-repudiation. The identity of the sender is ensured by the private key of sender. So,the receiver can verify the identity of the sender. The author of the message is really who they claim to be. Digital signature ensures that the data is not tampered during the transfer. It ensures the message or a document cannot be altered while transmitting. As it ensures the authenticity of the sender.The falsely deny of the sender is also not possible. The author of the message can't later deny that they were the source.

## 1.2 Problem Statement

These days almost all organizations around the globe uses Intranet to transfer data among their employees. But the security provided is not of high standards. More and more unauthorized people are gaining access to confidential data.

The validity of sender is not known. The sender may deny sending a message that he/she has actually sent and similarly the receiver may deny the receipt that he/she has actually received. Unauthorized people can gain access to classified data. Intruders can modify the messages or the receiver himself may modify the message and claim that the sender has sent it.

## 1.3 Objectives

This project has been developed keeping in view the security features that need to be implemented in the Local Area Network following the fulfillment of these objectives:

* To deal with the security threats that arise in the network using Rivest–Shamir–Adleman (RSA)
* To enable the end-users come out with a safe file transfer without any threats from intruders or unauthorized people.

## 1.4 Scope and Limitations

### 1.4.1 Scope

Our system aim is broad in terms of other file sharer as it solves the problems of sending the file with authenticity, as RSA algorithm has better security triad. This system can be used as an solution for the intra-office file transfer. Easy transmission i.e. uploading, and downloading of documents or files can be performed. Users can send chunk of files at a time.

### 1.4.2 Limitation

The limitations of our System are as follows:

* It cannot be used to share file across network with different public IP address.
* The system is made for text files so it cannot allow the user to share multimedia files.
* An infrastructural barrier such as internet connectivity, electrical connection, etc. might interfere with how the system will work.
* The encryption facility is not yet provided by our system.

### 1.5 Report Organization

This report is divided into six chapters. Each chapter is further divided into different headings.

* **Chapter 1** gives introduction about Digital Signature System. The problem definition, objectives, scopes and limitation of this system are discussed here.
* **Chapter 2** focuses on the analysis part. It contains literature review section where the research works done in the field of the Document Signing and Verification System. This chapter also includes feasibility study, requirement analysis and diagram like DFD, Sequence.
* **Chapter 3** discusses in details about the design of the system. This chapter also discusses about interface design and flowchart of the system built.
* **Chapter 4** gives information about implementation and testing process. It discusses about how the system is implemented and what tools and software are used to implement this system. The testing process is also included in detail in this chapter.
* **Chapter 5** includes conclusion of whole project. This chapter shows major achievements in the system and also shows how it can be enhanced later in the future so that it can be made more productive and useful than prevailing system.

**CHAPTER 2**

**Background Study and Literature Review**

## 2.1 Literature Review

Pretty Good Privacy in short PGP, a freeware created by Phil Zimmerman, providing encryption and authentication for e-mail and file storage applications across multiple platforms uses RSA algorithm for its key transportation. If client machines are slow then DSA is used and if slower server then RSA is used for verification, yet again difference of speed is at the start of the process.

Digital signature schemes based on public-key cryptosystem are vulnerable to existential forgery attack which can be prevented by use of one-way hash function and message redundancy. In this paper the authors have proposed an forgery attack over the digital signature scheme proposed by Chang and Chang in 2004. The authors have additionally proven stepped forward scheme the use of new key agreement protocol over the Chang and Chang model which honestly lacks the usage of one way hash function and redundancy padding [2]

The problem of fair exchange is one of the major threats in the field of secure electronic transactions. In this paper the authors have presented a multi signature scheme based on DSA which describes a method of constructing efficient fair-exchange protocols based on improved DSA signatures.[3]

In this paper, the authors compare the computational times of RSA and DSA with some bits and choose which bits are better used. Then combine both RSA and DSA algorithms to improve data security. From the simulation results, the authors chose RSA 1024 for the encryption process and added digital signatures using DSA 512, so the messages sent are not only encrypted but also have digital signatures for the data authentication process.[4]

This paper proposed an implementation of a complete and practical RSA encrypt/decrypt solution based on the study of RSA public key algorithm. In addition, the encrypt procedure and code implementation is provided in details.[5]

The authors have proposed an enhance algorithm for the RSA cryptosystem. This new proposed cryptosystem uses a third prime number in calculating the value of n. This additional third prime number increases the factor complexity of n. It will provide more security to the RSA. [6]

The authors have proposed a scheme of digital signature in electronic government to settle some specific problems such as spilling out secret, forging or denial and so on. Apart from this, a brief analysis regarding security issues of digital signature is also mentioned in this paper. [7]

The public key cryptosystem RSA is the first and most popular cryptosystem for performing encryption and decryption of data, to keep data secret, to transfer data from one location to another. Also it is known that the security of RSA depends on large factorization. If the factorization is possible then the whole algorithm can become breakable.Authors proposed a new methodology to change the original modulus with the fake modulus. Therefore if the hacker factorizes this new modulus value then he will not be able to locate the original decryption key.[8]

**CHAPTER 3**

**System Analysis**

## 3.1 System Analysis

### 3.1.1 Requirement Analysis

The functional and non-functional requirements are necessary to analyze the system requirements before developing and implementing. As the system design evolves, requirements analysis activities support allocation and derivation of requirements down to the system elements representing the lowest level of the design. The functional requirements specify the documentation of the system and activities that a system must be able to perform.

Two types of requirements are analyzed in two different ways:

1. Functional requirements and
2. Non-functional requirements
3. **Functional Requirement**

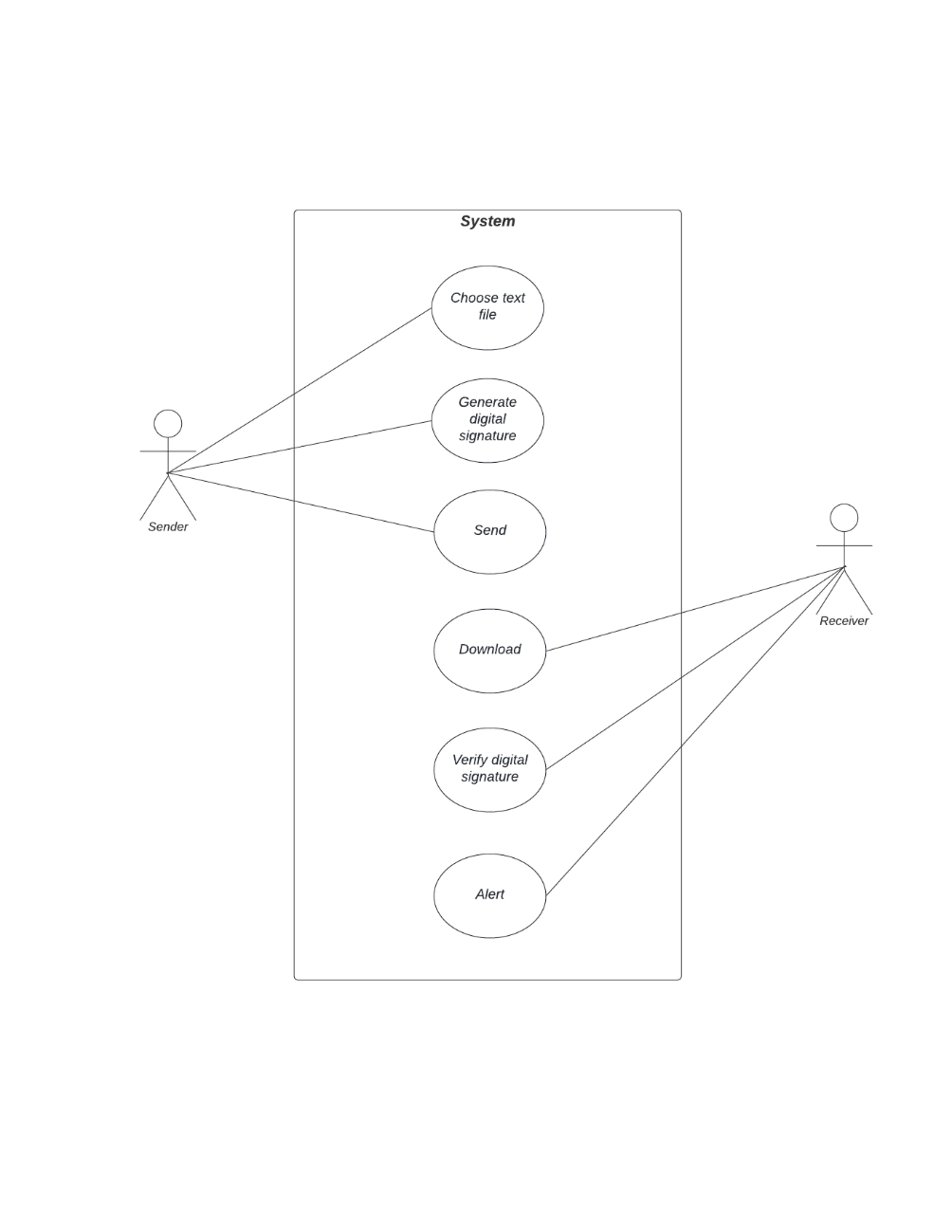
****

Figure 1:Use Case Diagram

* The system should allow user to send file to another user.
* The system should have the functionality of verifying the digital signature.
* The system should alert the user if file has been tempered.

1. **Non-functional Requirement**

In addition to the obvious features and functions that are provided in this system, there are other requirements that don’t actually do anything, but are important characteristics nevertheless which are called as non-functional requirement or sometimes Quality Attributes. They are those type of requirements which is not directly concerned with the system functionality but in absence of it reduces the quality of the system process.

For example : attributes such as performance, security, usability, compatibility.

### 3.1.2 Feasibility Analysis

1. **Technical**

It is desktop based application that uses Java Swing as front-end and core Java as backend. It will be based on client-server architecture provided by Java socket programming. Java is rich language making it technically feasible to develop proposed system.

1. **Operational**

The operational feasibility of project deals with user’s requirements and usefulness. The project is confined to the intranet in an organization. This application makes sure that security services such as authentication, integrity and non-repudiation are provided to the communicating parties. It can be easily operated once properly guided.

1. **Economic**

The purpose of economic feasibility is to analyze a project’s costs and revenues in an effort to determine whether or not it is logical and possible to complete. Economically it has very little cost for development of our software. Anyone with core Java knowledge can understand the ins and outs. For the deployment, it requires at least 2 PC with Internet connection. The operational and training cost is minimal because in most cases the user will be IT professional or the people with knowledge of how to operate computer programs.

1. **Schedule**

It is the most important for the completion of the project on time. The project that we are proposing will too be completed within time constraints.



Figure 2: Gantt Chart

### 3.1.3 Analysis

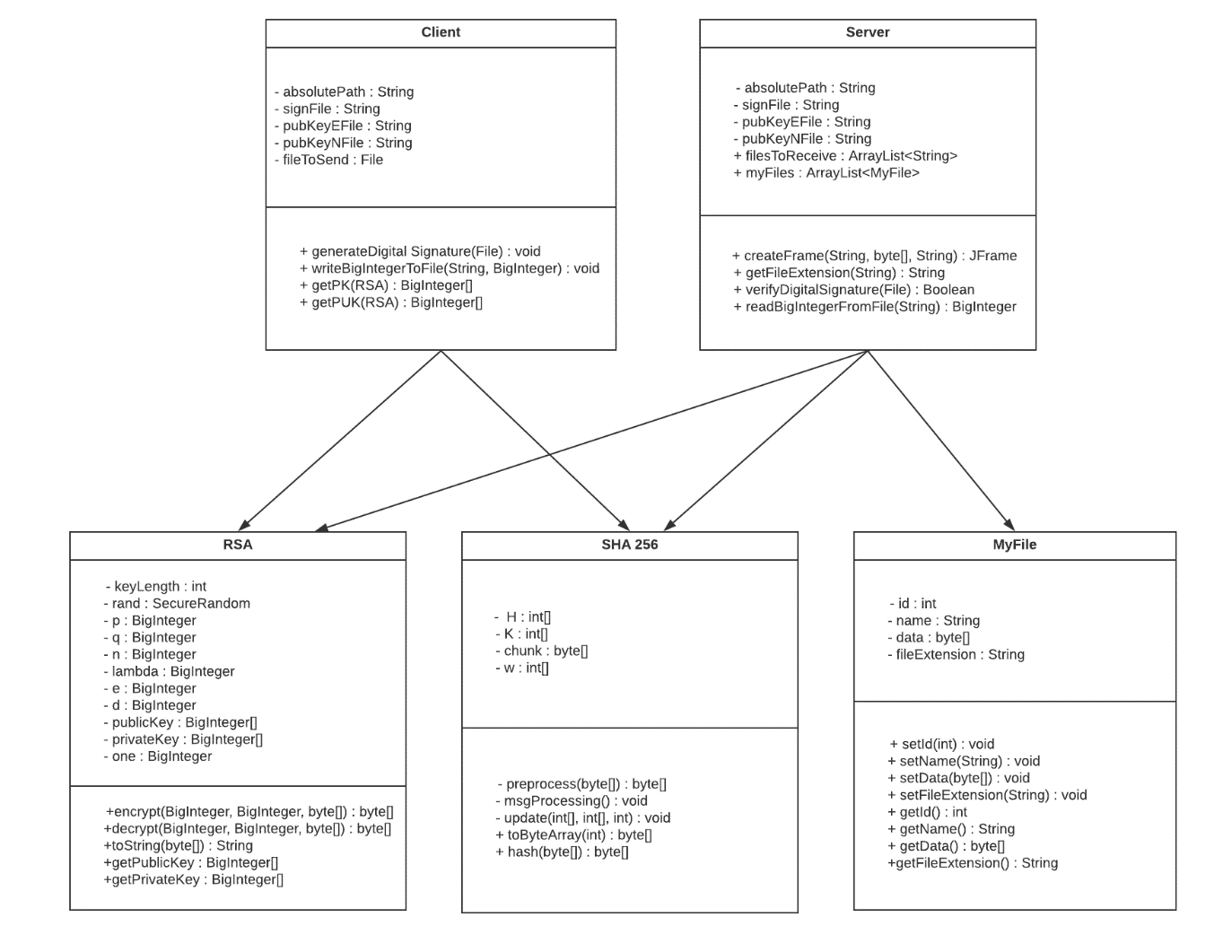


Figure 3: Class Diagram

**CHAPTER 4**

**System Design**

**4.1 Design**

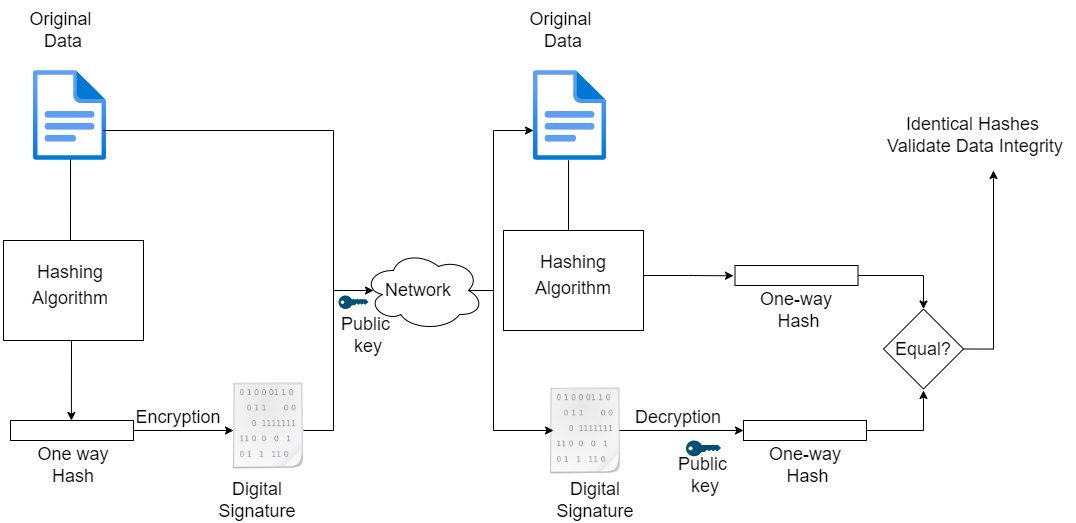
****

Figure 4: Activity Diagram

**4.2 Algorithm Details**

**RSA Algorithm**

Currently, there are three Federal Information Processing Standard (FIPS) approved digital

signature algorithms: Digital Signature Algorithm (DSA), RSA and Elliptic Curve Digital

Signature Algorithm (ECDSA). RSA is used in this project.

The RSA algorithm is a public-key signature algorithm developed by Ron Rivest, Adi

Shamir, and Leonard Adleman. Their paper was first published in 1977, and the algo-

rithm uses logarithmic functions to keep the working complex enough to withstand brute

force and streamlined enough to be fast post-deployment

**Key Generation** The required public and private keys are generated using following

steps :

• Two large prime numbers (p and q) are chosen.

• Compute n = p\*q and φ(n) = (p-1)(q-1)

• Public key(e) is chosen where 1 < e < φ(n)

• Private key(d) is calculated as de ≡ 1 modφ(n)

• Private key pair is (n,d)

• Public key pair is (n,e)

**1. Signing Algorithm**

S = Md mod n

**2. Verification Algorithm**

M = Se mod n

**3. Euclidean Algorithm**

It is used for determining GCD of two large positive integers.

while(b != 0)

gcd(a,b) = gcd(b, mod b)

gcd(a,0) = a

**4. Extended Euclidean Algorithm**

ri = si ∗ r0 +ti ∗ r1

where si = si−2 −qi−1 ∗ si−1

ti = ti−2 −qi−1 ∗ti−1

i >= 2,s0 = 1,s1 = 0,t0 = 0,t1 = 1

s and t are Bezout’s identity. r1 and r2 are two integer

**SHA Algortihm**

The secure hash algorithm are a family of cryptographic hash functions that are published by the National Institute of Standards and Technology(NIST) along with the NSA. It was passed as a federal information processing standard also known as FIPS. SHA-256 uses 32-bit words

To be considered cryptographically secure the hash function should meet two requirements,

1. It is impossible for an attacker to generate a message that matches a specific hash value
2. It should be impossible for an attacker to create two messages producing the exactly same hash value. Even a slight change in the plaintext should trigger a drastic difference in the two digest.

A message is processed by blocks of 512 = 16 × 32 bits, each block requiring 64 rounds.

**Basic operations**

• Boolean operations AND, XOR and OR, denoted by ∧, ⊕ and ∨, respectively.

• Bitwise complement, denoted by ̄

• Integer addition modulo 232, denoted by A + B.

Each of them operates on 32-bit words. For the last operation, binary words are interpreted as

integers written in base 2.

• Rot R(A, n) denotes the circular right shift of n bits of the binary word A.

• Sh R(A, n) denotes the right shift of n bits of the binary word A.

• A||B denotes the concatenation of the binary words A and B.

**Functions and constants**

The algorithm uses the functions:

Ch(X, Y, Z) = (X ∧ Y ) ⊕ ( ∧ Z),

M aj(X, Y, Z) = (X ∧ Y ) ⊕ (X ∧ Z) ⊕ (Y ∧ Z),

Σ0(X) = RotR(X, 2) ⊕ RotR(X, 13) ⊕ RotR(X, 22),

Σ1(X) = RotR(X, 6) ⊕ RotR(X, 11) ⊕ RotR(X, 25),

σ0(X) = RotR(X, 7) ⊕ RotR(X, 18) ⊕ ShR(X, 3),

σ1(X) = RotR(X, 17) ⊕ RotR(X, 19) ⊕ ShR(X, 10),

and the 64 binary words Ki given by the 32 first bits of the fractional parts of the cube roots of the first 64 prime numbers:

0x428a2f98 0x71374491 0xb5c0fbcf 0xe9b5dba5

0x3956c25b 0x59f111f1 0x923f82a4 0xab1c5ed5

0xd807aa98 0x12835b01 0x243185be 0x550c7dc3

0x72be5d74 0x80deb1fe 0x9bdc06a7 0xc19bf174

0xe49b69c1 0xefbe4786 0x0fc19dc6 0x240ca1cc

0x2de92c6f 0x4a7484aa 0x5cb0a9dc 0x76f988da

0x983e5152 0xa831c66d 0xb00327c8 0xbf597fc7

0xc6e00bf3 0xd5a79147 0x06ca6351 0x14292967

0x27b70a85 0x2e1b2138 0x4d2c6dfc 0x53380d13

0x650a7354 0x766a0abb 0x81c2c92e 0x92722c85

0xa2bfe8a1 0xa81a664b 0xc24b8b70 0xc76c51a3

0xd192e819 0xd6990624 0xf40e3585 0x106aa070

0x19a4c116 0x1e376c08 0x2748774c 0x34b0bcb5

0x391c0cb3 0x4ed8aa4a 0x5b9cca4f 0x682e6ff3

0x748f82ee 0x78a5636f 0x84c87814 0x8cc70208

0x90befffa 0xa4506ceb 0xbef9a3f7 0xc67178f2

**Padding**

To ensure that the message1 has length multiple of 512 bits:

• first, a bit 1 is appended,

• next, k bits 0 are appended, with k being the smallest positive integer such that l + 1 + k ≡ 448 mod 512, where l is the length in bits of the initial message,

• finally, the length l < 264 of the initial message is represented with exactly 64 bits, and these bits are added at the end of the message.

The message shall always be padded, even if the initial length is already a multiple of 512.

**Block decomposition**

For each block M ∈ {0, 1}512, 64 words of 32 bits each are constructed as follows:

• the first 16 are obtained by splitting M in 32-bit blocks

M = W1||W2|| · · · ||W15||W16

• the remaining 48 are obtained with the formula:

Wi = σ1(Wi−2) + Wi−7 + σ0(Wi−15) + Wi−16, 17 ≤ i ≤ 64.

**Hash computation**

• First, eight variables are set to their initial values, given by the first 32 bits of the fractional part of the square roots of the first 8 prime numbers:

H1(0) = 0x6a09e667

H2(0) = 0xbb67ae85

H3(0) = 0x3c6ef372

H4(0) = 0xa54ff53a

H5(0) = 0x510e527f

H6(0) = 0x9b05688c

H7(0) = 0x1f83d9ab

H8(0) = 0x5be0cd19

• Next, the blocks M(1), M(2), . . . , M(N) are processed one at a time:

For t = 1 to N

– construct the 64 blocks Wi from M(t), as explained above

– set

(a, b, c, d, e, f, g, h) = (H1(t−1), H2(t−1), H3(t−1), H4(t−1), H5(t−1), H6(t−1), H7(t−1), H8(t−1))

– do 64 rounds consisting of:

**CHAPTER 4**

**Conclusions and Future Recommendations**

**6.1 Conclusions**

The system is designed to provide the simple text file sharing with digital signature functionality. But at the end of the day it is upto the Network Administrator to make sure that his network is out of danger.

**6.2 Future Recommendations**

The algorithm used in the system can be extended to sign the files other than .txt extension. The encryption and decryption facility can be incoporated to make the sharing of information more confidential.