A NOVEL GROUP SIGNATURE SCHEME WITHOUT ONE WAY HASH

A THESIS SUBMITTED IN PARTIAL FULLFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELORS OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING

BY

BISWAJIT JENA (107CS006)
GOURI SANKAR MISHRA (107CS053)

Under the Guidance of
Prof. SUJATA MOHANTY

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Orissa, India

National Institute of Technology

Rourkela

CERTIFICATE

This is to certify that the thesis entitled, A NOVEL GROUP SIGNATURE SCHEME WITHOUT ONE WAY HASH submitted by Biswajit Jena, Roll No: 107CS006 and Gouri Sankar Mishra, Roll No: 107CS053 in partial fulfillment of the requirements for the award of Bachelor of Technology Degree in Computer Science and Engineering at the National Institute of Technology, Rourkela is an authentic work carried out by them under my supervision and guidance. To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university / institute for the award of any Degree or Diploma

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university / institute for the award of any Degree or Diploma

Prof. Sujata Mohanty

Dept. of Computer Science and Engineering

National Institute of Technology Rourkela - 769008

ACKNOWLEDGEMENT

We avail this opportunity to extend our hearty indebtedness to our guide Prof. Sujata Mohanty, Computer Science Engineering Department, for their valuable guidance, constant encouragement and kind help at different stages for the execution of this dissertation work.

We also express our sincere gratitude to Prof. A.K. Turuk, Head of the Department, Computer Science Engineering, for providing valuable departmental facilities.

Submitted by :

Gouri Sankar Mishra
Roll no: 107CS053

Biswajit Jena
Roll no:107CS006

Abstract

The group signatures scheme was introduced by Chaum and van Heijst which allow members of a group to sign messages anonymously on behalf of the whole group. Only a designated Group Manager is able to trace the identify of the group member who issued a valid signature. The group members sign a message with their secret key gsk and produce a signature that cannot be linked to the identities of the signers without the secret key of the manager. The group manager can open the signature to recover the identities of the signers in case of any legal dispute. Group signatures have been widely used in Electronic markets where the sellers are the group members, the buyers are the verifiers and the market administrator is the group manager.

We aim to propose a group signature scheme that is devoid of any one-way hash function and is based upon the Integer Factorization Problem (IFP). The scheme uses the concept of safe primes to further enhance the security of the scheme. The scheme supports message recovery and hence the overload of sending the message is avoided. The scheme satisfies security properties such as Anonymity (The verifier cannot link a signature to the identity of the signer), Traceability (The Group Manager can trace the identity of the signer of any valid signature), Unforgeability (A valid signature cannot be produced without the group secret keys), Exculpability (Neither the GM nor any member can produce a signature on behalf of a group member).

Contents

List of Figures

Chapter 1

Introduction

1 Introduction

Digital signatures play a major role in the modern electronic society because of the properties they possess, i.e, integrity and authentication. According to the integrity property it ensures that the received messages are not modified during the transmission of message from sender to receiver, and the authentication property ensures that the sender is not impersonated. In well-known conventional digital signatures, such as RSA and DSA, a single signer is sufficient to produce a valid signature, and anyone can verify the validity of any given signature using some keys. Because of its importance, many variations of digital signature scheme were proposed, such as blind signature, group signature, undeniable signature etc, which can be used in different application situations.

Group Signatures allow members of a group to sign messages on behalf of the group. The signatures can be verified using a single group public key, but they do not reveal the identity of the signer. Furthemore, it is still difficult to predict that two froup signatures have been signed by the same group member or not. However, there exists a designated Group Manager who can, in case of a legal dispute, open signatures, i.e., reveal the identity of the signer.[1] Members of a company can sign contracts with the customers such that the customer does not know the identity of the signer.If some kind of problem arrises with that contract the company can trace the member responsible for the problem.

Group signatures can be used to conceal organizational Structures. For example, an employee of a large organization/company can use group signatures to sign documents on behalf of the company. In this situation, it is sufficient for a verifier (who maybe a customer of the company) to know that some employee of the company has signed the document. Moreover, in contrast to when an ordinary signature scheme would be used, the verifier does not need to check whether a particular employee is allowed to sign contracts on behalf of the company, i.e., he needs only to know a single companys public key which helps the verifier in the process of verification.

Our scheme is devoid of any one-way hash functions thereby reducing the complexity in the signature generation phase. The use of safe primes further enhances the security of the scheme and thereby reduces the probability of an outsider recovering the secret parameters from the signature. Our scheme also supports message recovery. The overload of sending the message is

avoided which considerably reduces the complexity and improves the security of the scheme.

1.1 Motivation

Group Signatures play a very important role in every e-commerce applications. There has always been an increase in demand for a more secure and a less complex Group Signature scheme and our proposed scheme provides these features. The use of safe prime concept increases the security of our proposed scheme.The complexity of our scheme is less because it is devoid of any one-way hash functios.

Chapter 2

Literature Survey

2 Literature Survey

2.1 What is Cryptography?

Cryptography is the modern technique by which ordinary text is converted to unintelligible text. The ordinary text is otherwise known as plain text and unintelligible text is called cipher text. This technique is also known as encryption. In reverse decryption is the technique of converting the cipher text back to the original text. In the past, cryptography meant only encryption and decryption of message by the use of a common secret key. Due to the advancement of the technology, now-a-days three different standard mechanisms have been proposed. They are [11, 12]

- Symmetric Key cryptography
- Asymmetric Key cryptography
- Hashing

2.1.1 Symmetric Key Cryptography:

In this technique, the message is encrypted by the sender using some encryption algorithm and a secret key which is known only to the sender and the receiver. On receiving the message, the receiver decrypts the message using some decryption algorithm and the same secret key. Thus, in symmetric key Cryptography, only one key is used both for encryption and decryption which has to be passed between the sender and the receiver in a secure channel.

2.1.2 Asymmetric Key Cryptography:

This is also known as Public key Cryptography. It is similar to symmetric key cryptography in the sense that it involves encryption and decryption of message. The difference lies in thee number of keys used. This technique uses two keys i.e public key and private key. The sender uses the public key of the receiver to encrypt the message and sends it to the receiver. The receiver decrypts the message using its private key and receives the messageas shown in figure 1.[11, 12]

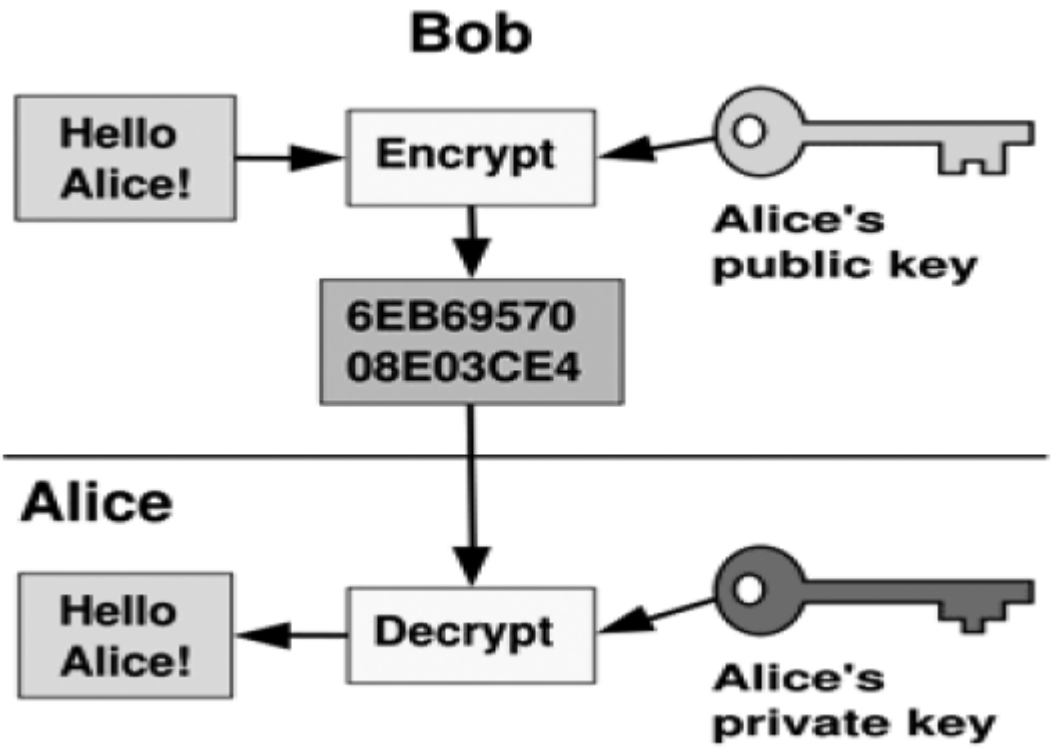

Figure 1: Asymmetric Key Cryptography

2.2 Cryptanalysis:

Cryptanalysis is the method of obtaining the meaning of encrypted information without the information of the secret parameters that are normally required to obtain the meaning.This typically involves the knowing of the system, how it works and finding the secret key. In non-technical language, this is the practice of code breaking or cracking the code, although these phrases have a specialized technical meaning. "Cryptanalysis" is used also to refer to any attempt to circumvent the security of some other types of cryptographic algorithms and protocols in general, and not just encryption. However, cryptanalysis usually excludes methods of attack that do not primarily target weaknesses in the actual cryptography, although these types of attack are an important concern and are often more effective than traditional cryptanalysis. [11, 12]
The International Telecommunication union-Telecommunication standardization Sector (ITU-T) provides some security services and some mechanism to implement those services.

2.3 Security Services:

The security services include:
- Data Confidentiality
- Data Integrity
- Authentication
- Non repudiation
- Access Control

2.3.1 Data Confidentiality:

Confidentiality has been defined by the International Organization for Standardization (ISO) in ISO-17799 as "ensuring that information is accessible only to those authorized to have access" and is one of the cornerstones of information security. Confidentiality is one of the design goals for many cryptosystems, made possible in practice by the techniques of modern cryptography.
It is designed to protect data from disclosure attack. The service as defined by X.800 is very

broad and encompasses confidentiality of whole message or part of a message and also protection against traffic analysis. That is it is designed to prevent snooping and traffic analysis [11, 12]

2.3.2 Data Integrity:

Data Integrity is designed for the protection of data from unauthorized modification, insertion, deletion and replaying by an advisory. It can protect the whole message or the part of message.

2.3.3 Authentication:

This service provides the authentication of the party at the other end of the line. In the connection oriented communication, it provides the authentication of the sender or receiver during the connection establishment (peer entity authentication). In connectionless communication, it authenticates the source of data (also called data origin authentication).

2.3.4 Non-repudiation:

Non-repudiation service protects against repudiation by either the sender or the receiver of the data. In this with the proof of origin, the receiver of the data can later prove the identity of the sender. If denied. In non-repudiation with the real proof of delivery the sender of the data can later prove the data were delivered to the intended recipient [11, 12].
Non-repudiation is the concept of ensuring that a party in a dispute cannot repudiate, or refute the validity of a statement or contract. Although this concept can be applied to any transmission, including television and radio, by far the most common application is in the verification and trust of signatures.

2.3.5 Access Control:

Access control is a system which enables an authority to control access to areas and resources in a given physical facility or computer-based information system. An access control system, within the field of physical security, is generally seen as the second layer in the security of a physical structure. It provides security against unauthorized access against data. The term access in this

definition is very broad and can involve reading, writing, modifying, executing programs. [11, 12]

2.4 Digital Signature

A digital signature is an electronic security mark that can be added to files. It can be also defined as an electronic signature that is used for demonstrating the authenticity of a digital message or document. A digital signature functions for electronic documents like a handwritten signature does for printed documents. Digital signatures are equivalent to the traditional handwritten signature in many aspects. But the digital signature provides a high level of security than that of a handwritten signature because the digital signatures are very difficuilt to forge . Digital signature uses public key encryption to verify digital information such as documents and emails. Digital signature scheme consists of mainly three types of algorithms
A Key generation algorithm: It is used to generate the public key and the private key.
A Signing algorithm: It takes the private key and the message as input and produces the signature as the output.
A signature verifying algorithm: A message, public key and a signature is given. Accepts the message if the verification phase returns true, rejects if the authentication fails.

Digital signature utilizes asymmetric encryption, where one key (private key) is used to create the signature code and a different but related key (public key) is used to verify it.
. A digital signature works by creating a message digest of size ranges between 128-bit to 256-bit number which is generated by running the entire message through a hash algorithm.The number generated from the hash algorithm is then encrypted with the sender's private key and added to the end of the message. When the receiver receives the message then he runs the message through the same hash algorithm and generates the message digest number. Then decrypts the signature using the senders public key and checks if the two numbers match then he is able know that the message is from the sender and it is has not been modified and if the match fails then the message received is not correct that means either it is modified or the sender is different.

Digital signature is used in many organizations for communication due to some of its properties like non-repudiation, authentication and integrity. Non-repudiation is the property by

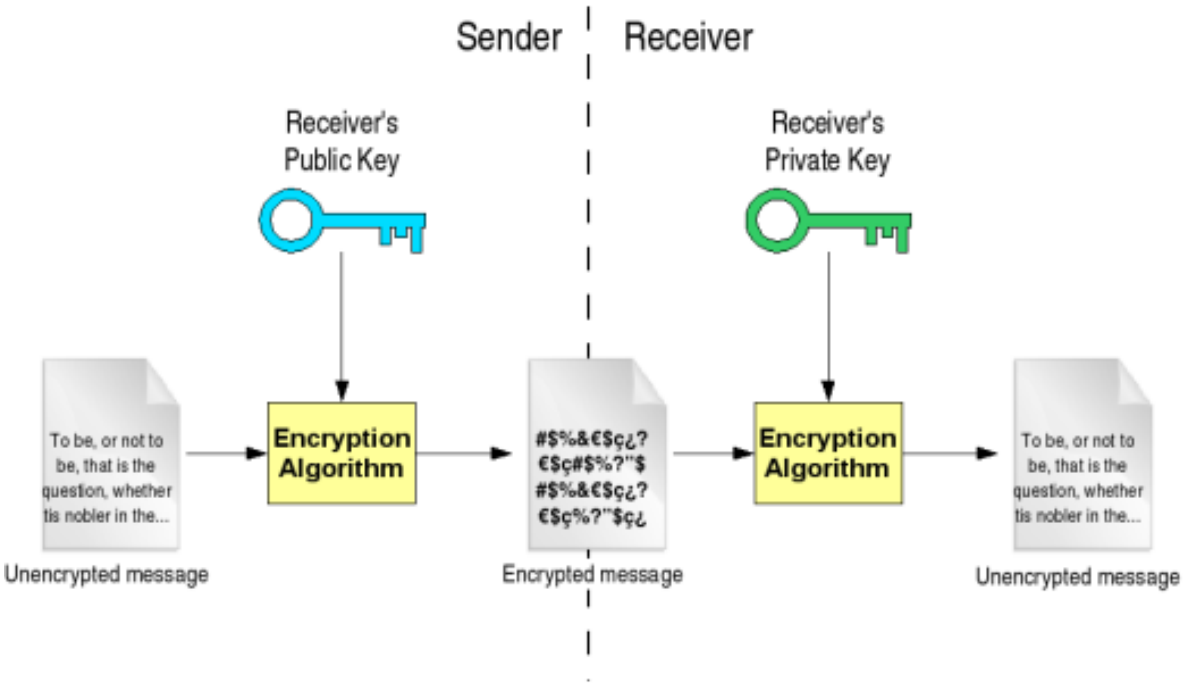

Figure 2: Digital Signature Model

which an entity that has signed some information cannot at a later time deny having signed it. Authentication is the property that describes about the information inside the message is accurate. Basically authentic means that you know who created the document and you know that it has not been altered or modified in any way since that person created it. We can also say that the digital signature is used to authenticate the source message. The authentication property ensures that the sender is not impersonated. Integrity is the property that describes the security of the message during transmission from source to destination. This means the message send at the sender side and the message received at the receiver side remains the same during its transmission.

Digital signatures offer many applications including signing messages such as e-mail. Digital signature can be created for any kind of files. The digital signature can be used as proof that the file was not modified after the digital signature was created. Using digital signature a Web browser and server can communicate with each other in a secure way. In Electronic money/digital cash digital signature is used to make the file unique it means appending a serial number to the file and signing it. Digital signature is also used to authenticate software applications. The manufacturer of a computer program can generate a digital signature for the users. When a user

downloads the program, he can verify that the digital signature is correct. If the user trusts the manufacture then he can safely install the application.

2.5 Group Signature

2.5.1 Definition

Group signature scheme was introduced by Chaum and van Heijst.[2] It allows any member of a group to digitally sign a document on behalf of the group in a manner such that a verifier can confirm that it came from the group, but does not know about the member who signed the document. For example the group signature scheme can be applied to a large company where numbers of employees are more. In this case any employee can sign a message on behalf of the company and during the verification the verifier is able know that the message comes from the company but he is unable to know the identity of the employee who signed the message.The essential part of the group signature scheme is the group manager who can add new members, delete members and also has the ability to reveal the identity of the signer.

Group signature is a useful cryptographical tool, which is widely discussed in the literature and also has many potential applications, such as network meeting, online business, and software trading. The similar requirement of these applications is to allow a member to sign a message on behalf of the group, and still remain anonymous within the group. Group signature schemes meet this requirement by providing anonymity and traceability at the same time, that is, a group signature can be related with its signers identity only by a party who possesses an open authority. In such environment, there exists a group manager to distribute certificates, open authority and other group settings. If one group member generates a group signature, anyone can only verify the signature by using group public parameters. When some dissention happens, an opener finds out the real signers identity. In this way, group members could protect their privacy.

2.5.2 Properties

Anonymity:Given A valid signature it is hard for anyone to determine the identity of the signer. Though the constant varies each time, the same member can generate different signatures each time he signs the message. Only the group manager is able to determine the identity of the signer

using his secret key. For an outsider it is almost impossible to determine the secret parameters of the signer because it requires the knowledge of the secret key of the group manager and hence without the secret key of the group manager it is difficult to determine the secret parameters of the signer and hence the identity of the signer cannot be determined.. Here we conclude that according to this property if both group managers secret key and group members secret key are not exposed then it is infeasible to find the signer of a particular valid signature.

Unforgeability:Only the member of a group can produce a valid signature that means only a valid member can produce a signature on behlf of the group.

Unlinkability:According to this property, deciding whether two valid signatures were computed by the same group member is hard.According to this property given two signatures one cannot come to the conclusion that they both are from the same member or not.

Traceability:Given any valid signature, only the group manager can trace the identity of the signer by using the open algorithm and the group manager's secret key.Hence the identity of any signer can be traced only by the group manager in case of any legal dispute or other emergencies. But an outsider cannot trace the identity of the signer because open algorithm, used to trace the member, requires the knowledge of the group managers secret key.

Exculpability:The group members as well as the group manager cannot able to sign a document for another group member. Generating a valid signature requires the knowledge of the secret parameters of the signer. Each signer has their distinct secret keys which are used during the signature. Even a group manager cannot sign in place a group member because the group manager is not having the secret keys of the members.

2.5.3 Mechanisms

Group signature allows a group member to sign anonymously a message on behalf of the group. Anyone can verify the group signature with the group's public key. The group manager is only able to open the signature to identify the group member in case of any legal dispute or emergencies.

Participants: A group signature scheme consists of a group manager, a set of group members and a set of signature verifiers. Group manager is responsible for admitting, deleting and revoking anonymity of group signature in case of any legal dispute. Group members sign the message

whereas the verifier verifies the message.

Communication: We have to assume that all channels used during the communication are asynchronous which means the sender after putting a message in the channel need not wait for the receiver to get the message out of the channel. The communication channel between the signer and the receiver is assumed to be anonymous.

Basic terms used in the group signature schemes are group public key which is used by the verifier to check the validity of the signature, group secret key which is used by the signer to generate the signature and the group manager's secret key which is used to trace back the identity of the signer. A group signature scheme is comprised of the following procedures.

setup phase:In this phase group manager computes the public key and the secret key.Group manager implements the group key generation algorithm. On inputting a security parameter the algorithm returns the group public key as well as the group managers secret key. The group manager keeps the secret key and reveals the group public key.

Join phase:This phase establishes an interactive protocol between the group manager and the user that results in the user becoming a valid group member. In this phase a new group member joins the group. Group member chooses a secret key. Using this secret the member generates another parameter using one-way trap door function and sends the generated parameter to the group manager. Then group manager using his own secret key generates the signing key for the group member and returns the signing key to the group member. This phase

Sign phase:This phase establishes an interactive protocol between the group member and the verifier where a group signature has to verified by the verifier which is generated by a valid group member. This is the signature phase.Group member signs the message using the signing key pairs.The group member generates the signature of knowledge and sends the signature to the verifier for verification.

Verify phase: In this phase a deterministic algorithm is implemented to verify the validity of a group signature using given a group public key and a signed message. Check the validity of the signature. Verifier receives the signature from the signer. Verifies the signature using the signature of knowledge. Accepts the message if the verification phase returns true and rejects the message if the verification phase returns false.

Open phase: In this phase a deterministic algorithm is implemented to determine the identity

of the signer, given a signed message and the group manager's secret key. Group manager takes the signature as input and with the help of the secret parameter gives the identity of the signer as output.The open algorithm is implimented in case of any legal dispute or in emergencies.

2.5.4 Applications

Group signatures have many applications. In particular, they can be used as foundation for anonymous credential systems in various applications. Group signature scheme could be used by the employees of a large company, where each employee can sign a document on behalf of the whole company. Another application of the group signature is the use of keycards to the restricted areas.

2.6 Cryptographic Background

2.6.1 Random Number Generation

A random number generator is a computational device designed to generate a sequence of numbers that lack any pattern, i.e. appear random.

The many applications of randomness have led to the development of several different methods for generating random data. Many of these have existed since ancient times, including dice, coin flipping, the shuffling of playing cards, the use of yarrow stalks (by divination) in the I Ching, and many other techniques. Because of the mechanical nature of these techniques, generating large amounts of sufficiently random numbers (important in statistics) required a lot of work and/or time. Thus, results would sometimes be collected and distributed as random number tables. Nowadays, after the advent of computational random number generators, a growing number of government-run lotteries, and lottery games, are using RNGs instead of more traditional drawing methods. RNGs are also used today to determine the odds of modern slot machines.

Random number generators have applications in gambling, statistical sampling, computer simulation, cryptography, completely randomized design, and other areas where producing an unpredictable result is desirable. They are used in cryptography so long as the seed is secret. Sender and receiver can generate the same set of numbers automatically to use as keys.

Random Number Generation plays a vital role in Group Signatures. A basic property of any

Group Signature is that it should be untraceable and Random Number Generators help satisfy the property. Each time a member signs a message, randomness in the algorithm ensures that the signatures are different from each other and that no outsider can reveal the identity of the signer from the signature, neither can he claim that two signatures are signed by the same member. Random Number Generators also help reduce the burden of assigning values to parameters required to setup the group.

2.6.2 Primality Test

A primality test is an algorithm for determining whether an input number is prime. Amongst other fields of mathematics, it is used for cryptography. Unlike integer factorization, primality tests do not generally give prime factors, only stating whether the input number is prime or not. As of 2010, factorization is a computationally difficult problem, whereas primality testing is comparatively easy (its running time is polynomial in the size of the input). Some primality tests prove that a number is prime, while others like Miller-Rabin prove that a number is composite. Therefore we might call the latter compositeness tests instead of primality tests.

Primality tests come in two varieties: deterministic and probabilistic.

2.6.2.1 Deterministic Algorithm A deterministic primality testing algorithm accepts an integer and always outputs a prime or a composite. Deterministic tests determine with absolute certainty whether a number is prime. Until recently, all deterministic algorithms were so insufficient at finding larger primes that they were considered infeasible. In 2002, Agrawal, Kayal and Saxena announced that that they had found an algorithm for primality testing with polynomial time complexity of $O((\log^{12} n))$.

2.6.2.2 Probabilistic Algorithm Probabilistic tests can potentially (although with very small probability) falsely identify a composite number as prime (although not vice versa). However, they are in general much faster than deterministic tests. Numbers that have passed a

probabilistic prime test are therefore properly referred to as probable primes until their primality can be demonstrated deterministically.

2.6.2.2.1 Fermat's Test The first probabilistic, method we discussed in the Fermat Primality test:

If n is a prime, then $a^{n-1} \equiv 1 \pmod{n}$

Note that this means if n is prime, the congruence holds. It does not mean that if the congruence holds, n is prime. The integer can be prime or composite. We can define the following as Fermats test:[11,12]

If n is a prime, then $a^{n-1} \equiv 1 \bmod n$

If n is composite, it is possible that $a^{n-1} \equiv 1 \bmod n$

All primes pass the Fermats test. Composite may also pass the Fermats test as well. The bit operation complexity of Fermats test is same as the complexity of an algorithm that calculates the exponentiation.

2.6.2.2.2 Square Root Test: In modular arithmetic, if n is a prime the square root of 1 is either +1 or -1. If n is composite the square root is +1 or -1, but there may be other roots. This is known as square root Primality test.[11,12]

If n is a prime, sqrt (1) mod n=+1 or-1

If n is a composite, sqrt (1) mod n=+1 or-1 and possibly other values.

2.6.2.2.3 Miller-Rabin Primality Test: The Miller-Rabin Primality test combines the Fermats test and square-root test in a very elegant and efficient way to find a strong pseudo prime (a prime with a very high probability of being a prime). In this test we write n-1 as the product of an odd number and a power of two.

$$\text{n-1} = \text{m}^* \, 2^k$$

In other words, instead of calculating $a^{n-1} \pmod{n}$ in one step, we can do it in k+1 steps. The benefit is that in each step, the square root test can be performed. If the square root test fails

we stop and declare that n is a composite number. In each step we assure ourselves that the Fermats test is passed and the square root test is satisfied between all pairs of adjacent steps, if applicable. It is a probabilistic method. There exists a proof that each time the number passes the Miller-Rabin Primality Test, the probability that it is not a prime is 1/4. If the number passes m tests (with m different bases) the probability that it is not a prime is $(1/4)^m$. [12]

2.6.3 Discrete Logarithm

In mathematics, specifically in abstract algebra and its applications, discrete logarithms are group-theoretic analogues of ordinary logarithms. In particular, an ordinary logarithm $\log_a(b)$ is a solution of the equation $a^x = b$ over the real or complex numbers. Similarly, if g and h are elements of a finite cyclic group G then a solution x of the equation $g^x = h$ is called a discrete logarithm to the base g of h in the group G.
In general, let G be a finite cyclic group with n elements. We assume that the group is written multiplicatively. Let b be a generator of G; then every element g of G can be written in the form $g = b^k$ for some integer k. Furthermore, any two such integers k1 and k2 representing g will be congruent modulo n. We can thus define a function

$$\log_b : G \to Z_n$$

(where Z_n denotes the ring of integers modulo n) by assigning to each g the congruence class of k modulo n. This function is a group isomorphism, called the discrete logarithm to base b.
The familiar base change formula for ordinary logarithms remains valid: If c is another generator of G, then we have

$$\log_c(g) = \log_c(b) * \log_b(g)$$

No efficient classical algorithm for computing general discrete logarithms $\log_b g$ is known. The naive algorithm is to raise b to higher and higher powers k until the desired g is found; this is sometimes called trial multiplication. This algorithm requires running time linear in the size of

the group G and thus exponential in the number of digits in the size of the group. There exists an efficient quantum algorithm due to Peter Shor.
More sophisticated algorithms exist, usually inspired by similar algorithms for integer factorization. These algorithms run faster than the naive algorithm, but none of them runs in polynomial time (in the number of digits in the size of the group).

- Baby-step giant-step
- Pollard's rho algorithm for logarithms
- Pollard's kangaroo algorithm (aka Pollard's lambda algorithm)
- Pohlig-Hellman algorithm
- Index calculus algorithm
- Number field sieve
- Function field sieve

2.6.4 Integer Factorization

In number theory, integer factorization or prime factorization is the breaking down of a composite number into smaller non-trivial divisors, which when multiplied together equal the original integer.
When the numbers are very large, no efficient integer factorization algorithm is known; an effort concluded in 2009 by several researchers factored a 232-digit number (RSA-768) utilizing hundreds of machines over a span of 2 years. The presumed difficulty of this problem is at the heart of certain algorithms in cryptography such as RSA. Many areas of mathematics and computer science have been brought to bear on the problem, including elliptic curves, algebraic number theory, and quantum computing.
Not all numbers of a given length are equally hard to factor. The hardest instances of these problems (for currently known techniques) are semiprimes, the product of two prime numbers. When they are both large, randomly chosen, and about the same size (but not too close), even the fastest prime factorization algorithms on the fastest computers can take enough time to make the search impractical.

Chapter 3

Group Signature

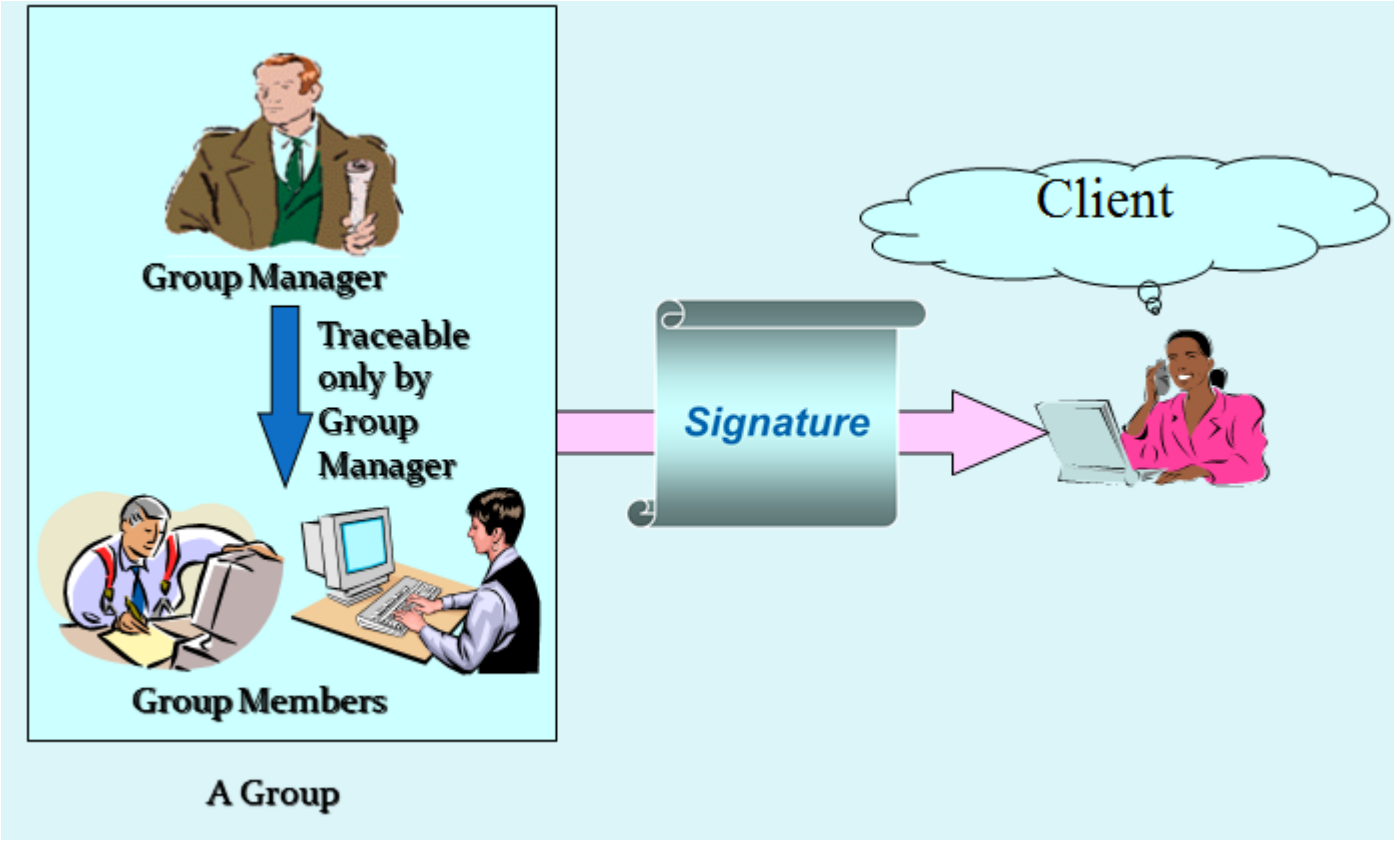

Figure 3: Group Signature Model

3 Group Signature

The basic model of a group signature scheme consists of a group manager, group members and a verifier. The group members generate signatures on behalf of the group. The verifier checks the signature for validity and accepts or rejects the message. The verifier cannot trace the identity of the signer. In case of any legal disputes, the group manager can trace the identity of the signer using an open algorithm. No one including the group manager can sign a message on behalf of any other group member.

3.1 Algorithm

The scheme consists of four kinds of participants, a trusted authority for generating secrets keys of all signers, a group manager for managing the memberships and identifying the sign- ers,

several signers (group members) for issuing group signatures and several verifers for checking them.

Generally, the trusted authority can be appointed by the government.

3.1.1 Setup Phase

The Group Manager chooses

- an integer n = p*q, where

  p = 2*p'*f+1

  q = 2*q'*f+1

  p, q, p', q', f are all primes.

- an integer g of order f both modulo p and q.

- an integer e that is co-prime with p-1 and q-1.

- compute d such that

  e*d $\equiv$ 1 mod ϕ(n).

Public : g, n, y_G

3.1.2 Key Generation

- Each group member A chooses a random $S_A \in Z_f$ and computes $y_A = g^{S_A}$ mod n.

It gives y_A to the Group Manager.

- Group Manager computes

  $x_A = (x_G * y_A)^{-d}$

  $y_G = g^{x_G}$ mod n

y_G is group's identity or Group Public Key and x_G is Group Manager's private key.

- Group Manager gives x_A to member A secretly.

- The secret key of member A is the pair (x_A , S_A).

Group Manager knowing the part of A's secret key may not sign message under A's name.

3.1.3 Signature Generation

$$s = (m * y_G)^d \bmod n$$

$$v = (S_A + k) \bmod n$$

$$r = s + m * g^{-k*v*y_A^{-1}} \bmod n$$

Compute t from the following equation

$$s + t \equiv x_A^e * k * v$$

Signature is (r, s, t).

3.1.4 Signature Verification

The verifier recovers the message m as

$$m = (r\text{-}s) * (y_G)^{s+t} \bmod n$$

The verifier then checks if

$$s^e \equiv (y_G * m) \bmod n$$

If condition is satisfied, then the message is accepted else the message is rejected.

3.1.5 Open

The Group Manager has with him the pair (x_A , y_A) corresponding to each member A.

For each pair (x_A , y_A), it checks if

$$r = s + m * g^{-y_A^{-1} * (s+t) * x_A^{-e}}$$

If the above condition is satisfied for a particular pair (x_A , y_A), then the signer is identified.

3.1.6 Correctness

$$\begin{aligned} m &= (r\text{-}s) * (y_G)^{s+t} \bmod n \\ &= m * g^{-k*v*y_A^{-1}} * (g^{x_G})^{s+t} \bmod n \\ &= m * g^{-k*v*y_A^{-1}} * (g^{x_A^{-e} * y_A^{-1}}) * x_A^e * k * v \bmod n \\ &= m * g^{-k*v*y_A^{-1}} * g^{k*v*y_A^{-1}} \bmod n \end{aligned}$$

$= m$

Thus the correctness of the scheme is proved.

3.2 Security Analysis

The security of our scheme is based on two computational hard problems, namely, Discrete Logarithmic Problem (DLP) and the Integer Factorization Problem (IFP).

3.2.1 Discrete Logarithmic Problem

The Group's public key Y_G is calculated as

$$Y_G = g^{X_G} \bmod n$$

This is a discrete logarithmic problem because to calculate X_G, we will have to calculate the discrete logarithm of Y_G to the base g. This is a computational hard problem and hence is difficult to solve and thus our scheme is secure.

The Group member's secret key Y_A is computed as

$$Y_A = g^{S_A} \bmod n$$

The group manager has access to the secret key Y_A of the group members. Thus he can calculate S_A as the discrete logarithm of Y_A to the base g. But as DLP is a computationally hard problem, the group manager cannot compute S_A and hence cannot forge a signature. Thus our scheme is secure.

3.2.2 Integer Factorization Problem

The problem of factoring semi primes and getting the prime factors is known as the Integer Factorization Problem. In our scheme, the group public parameter n is calculated as p * q. this is a integer factorization problem and is difficult to solve in polynomial time. Hence our scheme is secure.

3.3 Performance Comparison

Table 4.1 Comparative Study of Two Group Signature Schemes

	Kim's Scheme	Proposed Scheme
Group Signature Generation	6E + 5M + 1H + 1I	3E + 6M + 1I
Group Signature Verification	1H + 3E + 2M	2E + 2M

The table above shows the comparative study of two group signature schemes. Our proposed scheme uses a lot less operations than Kim's Scheme and is more secure than Kim's scheme. Our scheme is devoid of any hash functions which greatly reduces the complexity of the scheme. Due to less number of operations, the signing and verifying phase takes less time than that of Kim's. Our scheme also supports message recovery which reduces the overhead of message transmission along with the signature.

Chapter 4

Implementation Details

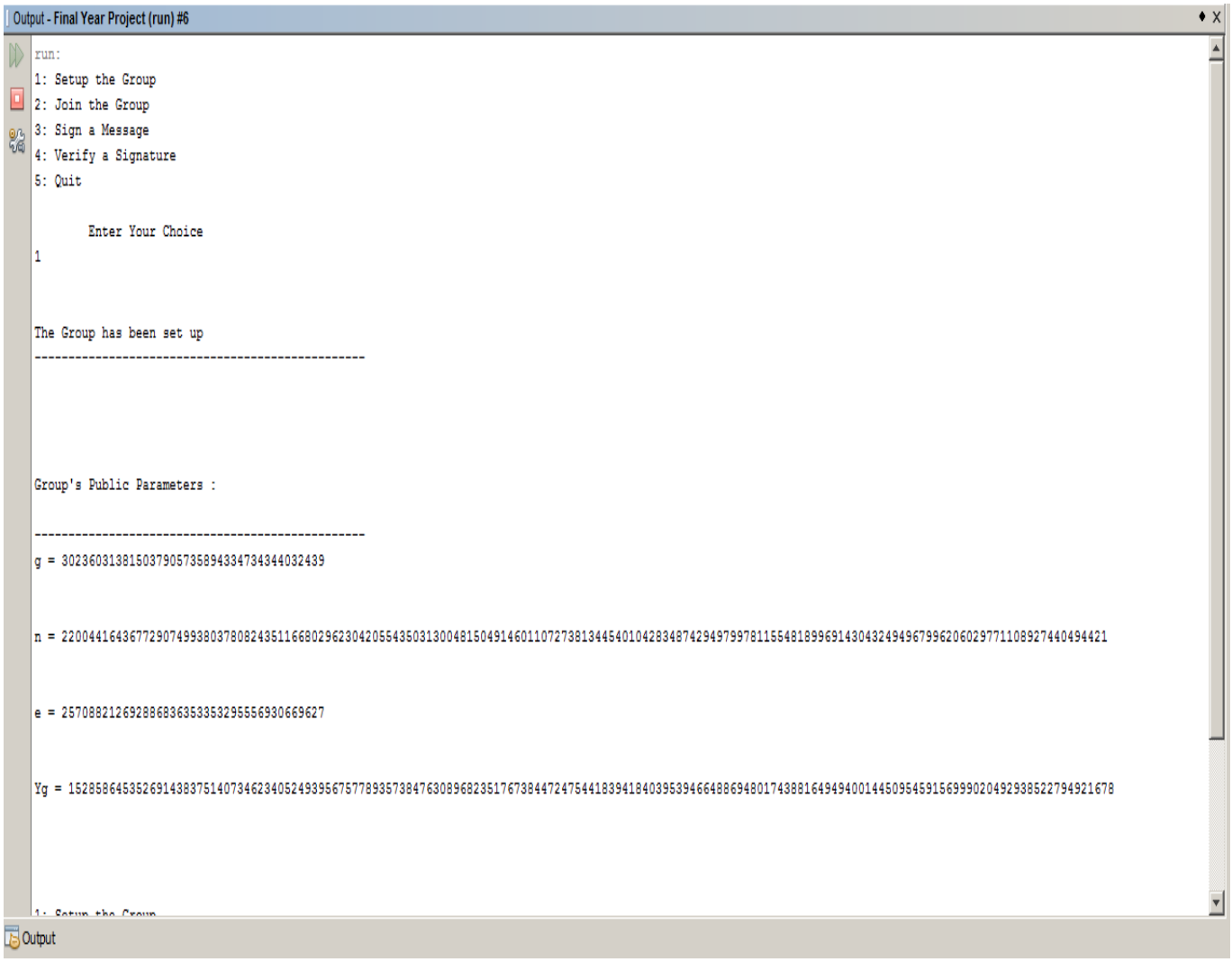

Figure 4: Setup Phase

4 Implementation Details

4.1 Setup Phase

The Group is setup and the parameters are :

Group Public Key :

n = 18291910092932794844090777370693311292061639602746081785367172805829003654245813347809486859
6133449763759538774733337940908779051045643082606718230299737 61

Y_G = 152984806391491651368034738078708153024712400575908364775401178291171732931435927626772942 6
490317850358369186070411053776097316566170219420874337664350 7208

e = 337911890180741807609006649785298649007

g = 198549536939412216370458096146808349283

f = 279868171109530510507348767591159311867

Group Private Key :

d = 174674987366221871862460417981163654200028338684539209601418114541919456398032998355918309624335345321741510560996853978342551183792219148250489075810912833

Group Manager Private Key :

X_G = 283264683514102680339010755811311902333

4.2 Join Phase

A member has been added with the following details

UserName : gouri

Password : riku

S_A : 230680347119742372313205819307064301599

Y_A : 143010467864560130349929980645738704975816740135350356711785534293779574815389155657332892416447677589932429277168458091022082809951028320925680722862582733

X_A : 161451794229153117814912155669960258176274096641312834347379902818195960523763323197563340876152573614010143133258063631030142315055608617790637533714668888

4.3 Sign Phase

The message has been signed.

Message (m) : hi there how are you

Signature :

R : 106340516244516809760877386037228228231490868994649651607138441434356531636679905891104648867641869116453652514150493697680397156987665028061836175304504553939831228887393103019206879121001887372509618687511180321575633

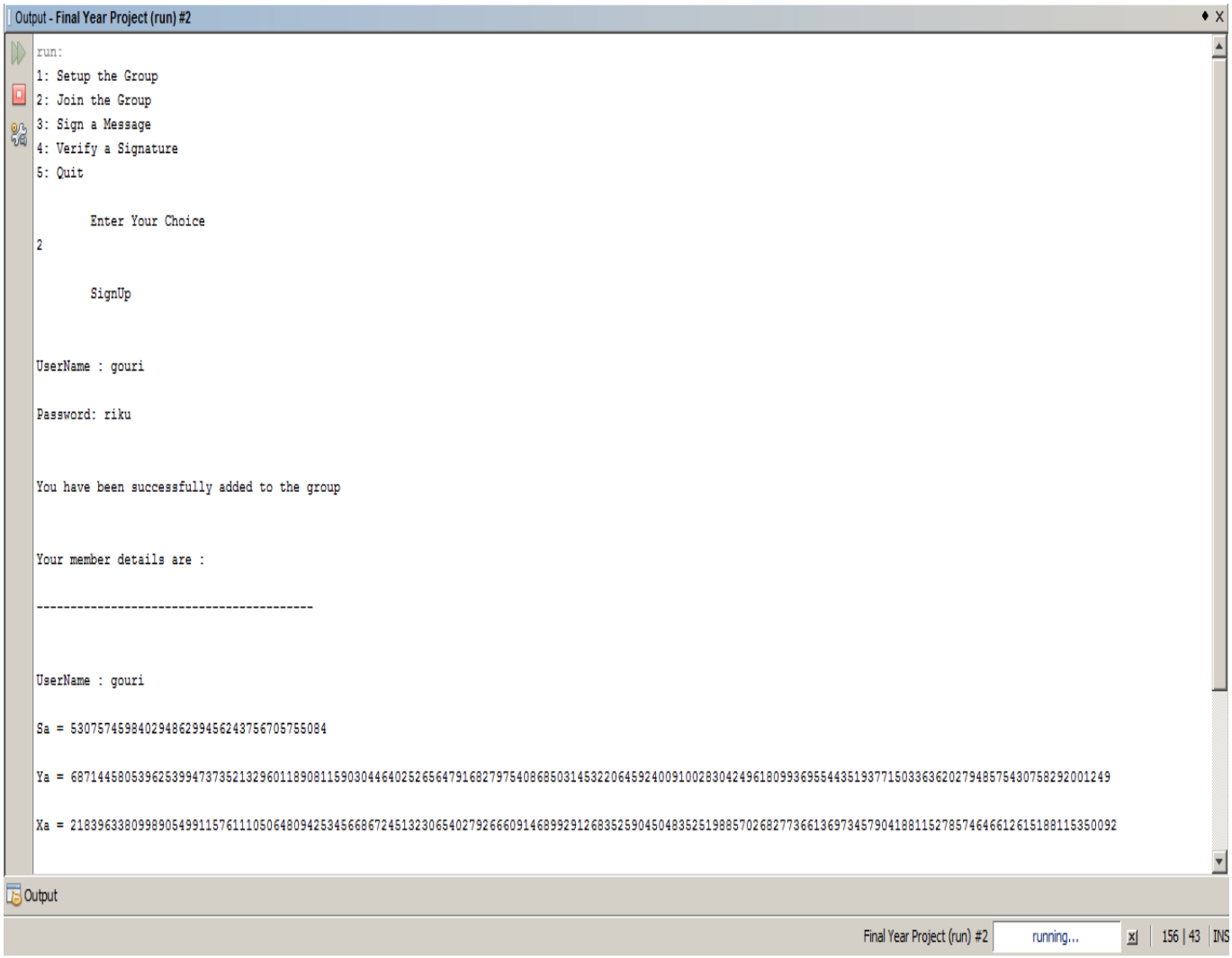

Figure 5: Join Phase

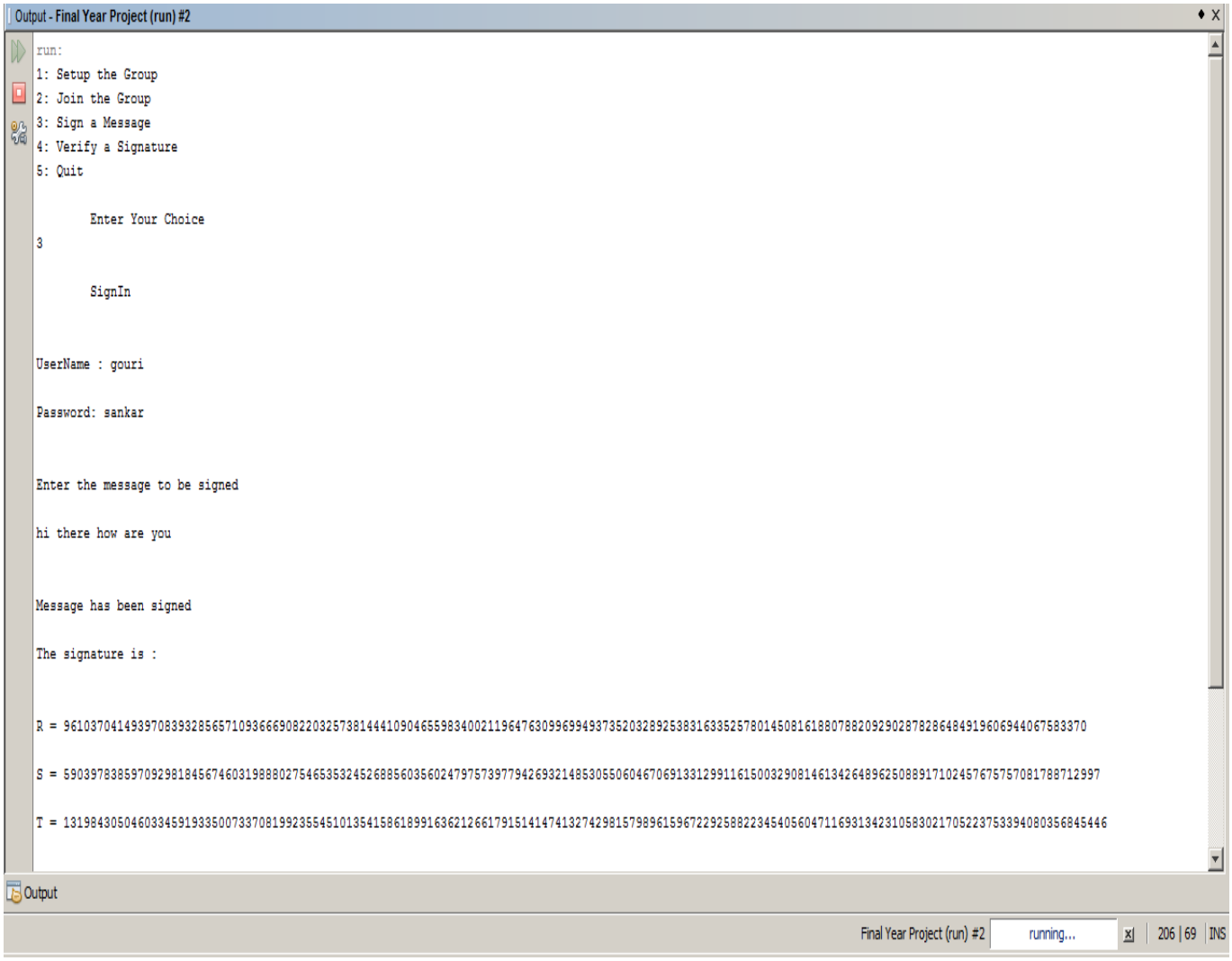

Figure 6: Sign Phase

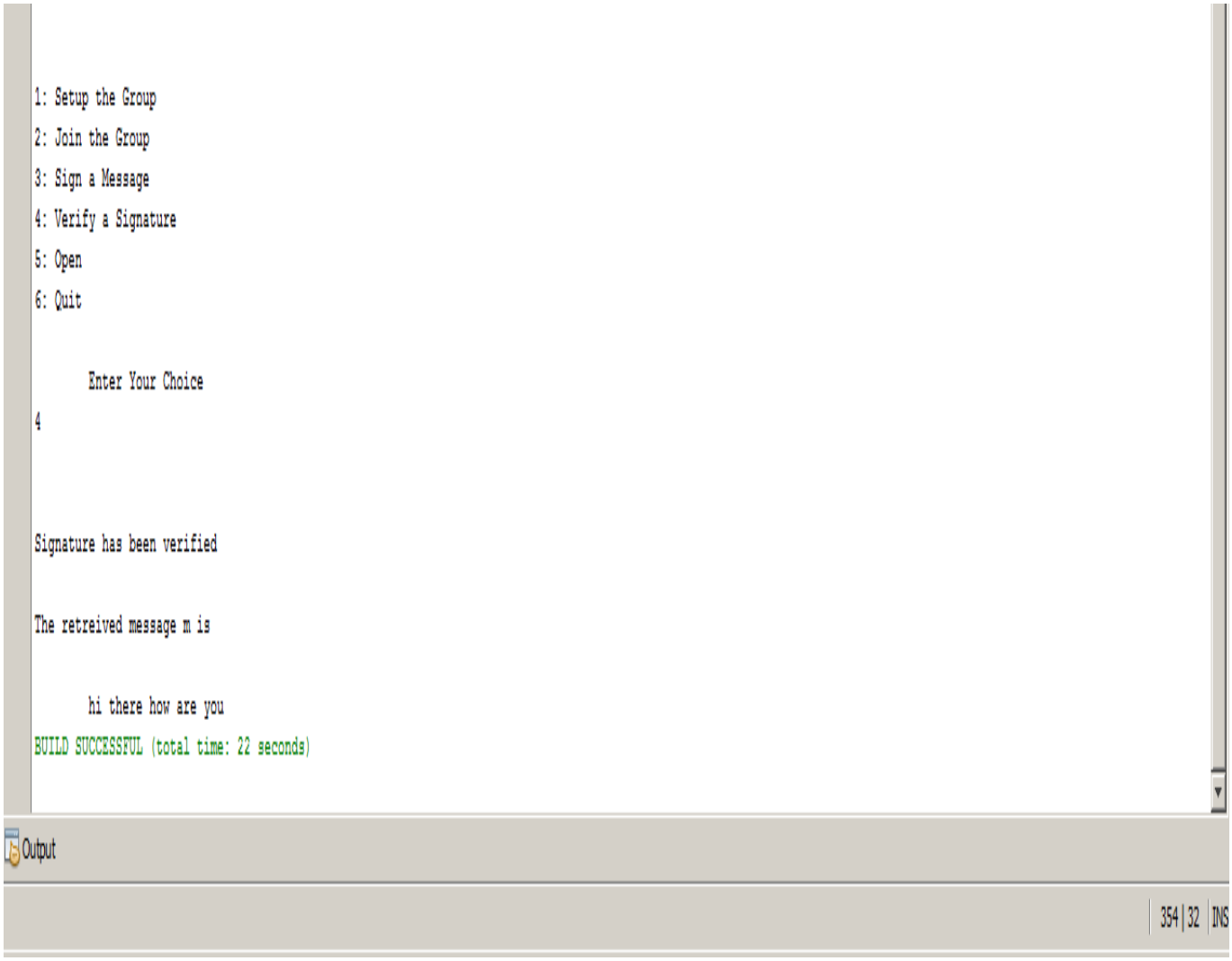

Figure 7: Verify Phase

T : 3183557552532357281099284952606924895607624604756280430133072110870220906079738168378075507228036665379181126933560915169328056796190277415315723821111560

4.4 Verify Phase

The signature has been verified and the message retreived.

Chapter 5

Conclusion

5 Conclusion

Our proposed Group Signature scheme uses safe primes to further enhance the security of the scheme. Our scheme is devoid of any one-way hash functions which further reduces the complexity of the scheme. In any general Group Signature scheme, message was send along with the signature to be used in the verification phase. In our scheme, the overload of sending the message is avoided as the message can be recovered from the signature parameters. So, in our scheme, complexity is reduced by not using hash functions but in the same time, the security is improved by using safe primes.

References

[1] Jan Camenisch and Markus Stadler. Efficient Group Signature Schemes for Large Groups. *Advances in Cryptology, CRYPTO '97*, pages 410 – 424, 1997.

[2] Heijst Chaum, D. Group signatures. *Springer, Berlin*, 547:257 – 265, 1992.

[3] Ikkwon Yie Kitae Kim Haeryong Park, Seongan Lim and Junghwan Song. Strong unforgeability in group signature schemes. *Computer Standards and Interfaces*, 31:856 – 862, June 2009.

[4] Tseng and Jan. A novel ID-based group signature. *Information Science*, 120:131 – 141, 1999.

[5] Joye Ateniese, Camenisch and Tsudik. A practical and provably secure coalition-resistant group signature scheme. *Springer, Berlin*, 1880:255 – 270, February 2000.

[6] J. Camenisch. Efficient and generalized group signatures. *Springer, Berlin,*, 1233:465 – 479, 1997.

[7] Pedersen T.P. Chen, L. New group signature schemes. *Springer, Berlin,*, 950:171 – 181, 1995.

[8] Sung Jun Park Seung Joo Kim and Dong Ho Won. Convertible group signatures. *Advances in Cryptology ASIACRYPT '96*, 1163/1996:311 – 321, 1996.

[9] Kim S. Won D. Park, S. Id-based group signature. *Electron. Lett*, 33:1616 – 1617, 1997.

[10] J.M. Pollard. Theorems on factorization and primality testing. *Proc. Cambridge Philos. Soc.*, 76:521 – 528, 1974.

[11] B Forouzan. *Cryptography and Network Security*. TMH.

[12] W Stallings. *Cryptography and Network Security*. Prentice Hall.