A Thesis for the Degree of Master of Science

Provably Secure Identification Protocol based on the Bilinear Diffie-Hellman Problem

Myungsun Kim

School of Engineering

Information and Communications University

2002

Provably Secure Identification Protocol based on the Bilinear Diffie-Hellman Problem

Provably Secure Identification Protocol based on the Bilinear Diffie-Hellman Problem

Advisor : Professor Kwangjo Kim

by

Myungsun Kim
School of Engineering
Information and Communications University

A thesis submitted to the faculty of Information and Communications University in partial fulfillment of the requirements for the degree of Master of Science in the School of Engineering

Daejeon, Korea
July. 1. 2002
Approved by

(signed)

Professor Kwangjo Kim
Major Advisor

Provably Secure Identification Protocol based on the Bilinear Diffie-Hellman Problem

Myungsun Kim

We certify that this work has passed the scholastic standards required by Information and Communications University as a thesis for the degree of Master of Science

July. 1. 2002

Approved:

Chairman of the Committee
Kwangjo Kim, Professor
School of Engineering

Committee Member
Hyuncheol Park, Assistant Professor
School of Engineering

Committee Member
Choonsik Park, Ph.D
NSRI

M.S. Myungsun Kim
2000502

Provably Secure Identification Protocol based on the Bilinear Diffie-Hellman Problem

School of Engineering, 2002, 49p.
Major Advisor : Prof. Kwangjo Kim.
Text in English

Abstract

We interact and transact by directing flocks of digital packets towards each other through cyberspace, carrying love notes, digital cash, and secret corporate documents. However, the technical wizardry enabling remote collaborations is founded on broadcasting everything as sequences of zeros and ones that one's own dog wouldn't recognize. How should you know that it *really is* me requesting from a laptop in Fiji a transfer of $100,000,000 to a bank. Fortunately, the magical mathematics of cryptography can help. That is, we need to have techniques that play a role of allowing one party to gain assurance that the identity of another is as declared. Names for such techniques include *identification* or *identity verification*.

To guarantee that an identification protocol withstands the attacks, the designed identification protocol should be strictly proven to be secure. However, the design of provably secure identification protocol has been regarded as a difficult task, but a fundamental task. As in the design of other cryptographic protocols, in provable security for identification schemes, first precise definitions of various attacks is given and then, using complexity theoretical techniques such as cryptographic reductions, their security is analyzed in mathematical way.

In this thesis, we deal with an interactive identification scheme based on the bilinear Diffie-Hellman problem and analyze its security. The scheme is more efficient than the Schnorr scheme and the Okamoto scheme with respect to preprocessing of prover and on-line processing overhead of both parties (prover and verifier). At the same time, security of our scheme is higher than or equal to previous schemes. We prove that this scheme is secure against active attacks as well as passive attacks if the bilinear Diffie-Hellman problem is intractable. Our proof is based on the fact that the computational Diffie-Hellman problem is hard in the additive group of points of an elliptic curve over a finite field, on the other hand, the decisional Diffie-Hellman problem is easy in the multiplicative group of the finite field mapped by a bilinear map. Finally, this scheme is compared with other identification schemes.

Contents

List of Tables

List of Figures

List of Abbreviations

B-DH Bilinear Diffie-Hellman

B-DHIA Bilinear Diffie-Hellman Intractability Assumption

BM Brickell-McCurley

C-DH Computational Diffie-Hellman

DH Diffie-Hellman

D-DH Decisional Diffie-Hellman

DLP Discrete Logarithm Problem

ECDLP Elliptic Curve Discrete Logarithm problem

G-DH Gap Diffie-Hellman

IFP Integer Factorization Problem

FS Fiat-Shamir

FFS Feige-Fiat-Shamir

GQ Guillou-Quisquator

IBE Identity-Based Encryption

ID Identity

MOV Menezes-Okamoto-Vanstone attack

OO Ohta-Okamoto

PPT Probabilistic Polynomial Time

ROM Random Oracle Model

RSA Rivest-Shamir-Adleman (encryption scheme)

TTP Trusted Third Party

ZKIP Zero-Knowledge Interactive Proof

ZKP Zero-Knowledge Proof

List of Notations

$\mathcal{G}$ key generation algorithm which is modeled as PPT

$\mathcal{P}$ a prover, which is a PPT algorithm

$\mathcal{V}$ a verifier, which is a PPT algorithm

$\tilde{\mathcal{P}}$ a dishonest prover, which is a PPT algorithm

$\tilde{\mathcal{V}}$ a dishonest verifier, which is a PPT algorithm

$\bar{\mathcal{P}}$ a honest prover, which is a PPT algorithm

$\bar{\mathcal{V}}$ a honest verifier, which is a PPT algorithm

k a security parameter

$I_{\mathcal{P}}$ a public key of $\mathcal{P}$

$S_{\mathcal{P}}$ a secret key of $\mathcal{P}$

Pub a collection of public parameters of given identification scheme

Sec a collection of secret parameters of given identification scheme

$\mathcal{I}$ an adversary $(\tilde{\mathcal{P}}, \tilde{\mathcal{V}})$

$x \leftarrow \mathcal{S}$ an element x randomly selected according to a probability space $\mathcal{S}$

σ a bit string $\{0,1\}^*$

$\Pr_{\mathcal{S}}[x]$ probability that $\mathcal{S}$ associates with the element x

e the Weil pairing

$\hat{e}$ the modified Weil pairing

$\mathbb{G}$ a cyclic group of a prime order

m the order of $\mathbb{G}$

$K, \overline{K}$ for a field K, algebraic closure

E elliptic curve

$E(K)$ group of K-rational points on E

$\mathcal{O}$ point at infinity (on an elliptic curve)

$E[m]$ group of m-torsion points on the elliptic curve E

P, Q elements of $E[m]$

$\mathrm{div}(f)$ the divisor of a function f

$\in_{\mathcal{R}}$ chosen at random

$\mathbb{Z}$ integers

$\mathbb{Z}/m\mathbb{Z}$, $\mathbb{Z}_m$ integers modulo m

$\mathbb{Z}_m^*$ a group under multiplication modulo m

$cert_A$ certificate of A

$\perp$ the null string

Chapter 1 Introduction

1.1 Interactive identification protocol

Today's computer networks lead our personal and economic lives to rely more and more on our ability to let such ethereal carrier pigeons mediate at a distance what we used to do with face-to-face meetings, paper documents, and a firm handshake. Unfortunately, the technical wizardry enabling remote collaborations is founded on broadcasting everything as sequences of zeros and ones that one's own dog wouldn't recognize. In other words, for interactions through cyberspace to appropriately proceed, there must be techniques to prove one to another that *he is really himself*. This thesis deals with a technique, called an identification protocol or entity authentication protocol, which allows one party to gain assurances that the identity of another is as declared, thereby preventing impersonation.

An identification protocol is considered to be as an interactive protocol and the general setting for the protocol involves a *prover* or claimant $\mathcal{P}$ and a *verifier* $\mathcal{V}$. In general, $\mathcal{P}$ tries to convince the verifier $\mathcal{V}$ of his identity. The verifier is presented with, or presumes beforehand, the purported identity of the prover. The goal is to corroborate that the identity of the prover is indeed $\mathcal{P}$, *i.e.*, to provide entity authentication. Only $\mathcal{P}$ knows the secret value corresponding to his public one, and the secret value allows to convince $\mathcal{V}$ of his identity.

One of the primary purposes of identification is to facilitate access control to a resource, when an access privilege is linked to a particular identity. Examples of these cases are local or remote access to computer accounts, withdrawals from automated cash dispensers, or physical entry to restricted area or border crossings. In many applications such as cellular telephony the motivation for identification is to allow resource usage to be tracked to identified entities, to facilitate appropriate billing. Identification is also typically an inherent requirement in authenticated key establishment protocols.

1.2 Objectives of identification protocols

From the point of view of the verifier, the outcome of an identification protocol is either *acceptance* of the prover's identity as authentic, or *rejection*. More specifically, the objectives of an identification protocol include the following.

1. In the case of honest parties $\mathcal{P}$ and $\mathcal{V}$, $\mathcal{P}$ is able to successfully authenticate himself to $\mathcal{V}$, *i.e.*, $\mathcal{V}$ will complete the protocol having accepted $\mathcal{P}$'s identity.

2. (*transferability*) $\mathcal{V}$ cannot reuse an identification exchange with $\mathcal{P}$ so as to successfully impersonate $\mathcal{P}$ to a third party $\mathcal{A}$.

3. (*impersonation*) The probability is *negligible*[1] that any party $\mathcal{A}$ distinct from $\mathcal{P}$, carrying out the protocol and playing the role of $\mathcal{P}$, can cause $\mathcal{V}$ to complete and accept $\mathcal{P}$'s identity.

4. The previous points hold even if: a polynomially large number of previous authentication between $\mathcal{P}$ and $\mathcal{V}$ have been observed; the adversary $\mathcal{A}$ has participated in previous protocol executions with either or both $\mathcal{P}$ and $\mathcal{V}$; and multiple instances of the protocol, possibly initiated by $\mathcal{A}$, may be run simultaneously.

[1] It typically means "is so small that it is not of practical significance"; the precise definition is given in Chapter 4.

The precise definition of goals for an identification protocol is given with respect to provable security against the attacks in later chapter. Informally speaking, the objectives derive the idea of zero-knowledge-based protocols whose executions do not reveal any partial information which makes $\mathcal{A}$'s task any easier whatsoever.

1.3 Our contributions

In this thesis, we present a formal model for secure identification protocol based on the bilinear Diffie-Hellman problem and make the precise definition of security for this model. To the best of our knowledge, no formal treatment for this cryptographic problem has ever been suggested. This is our first contribution.

We construct a new identification scheme base on the given hard problem, which is a typical instance of the gap Diffie-Hellman problem. In the security model, we prove that the identification scheme is secure against passive and even active attacks if the bilinear Diffie-Hellman problem is intractable. This is the second contribution of the thesis.

1.4 Outline of the thesis

In this thesis, we deal with security concerns regarding identification schemes that guarantee provable security against various attacks.

The rest of this thsis is organized as follows. In Chapter 2, several identification schemes are reviewed and types of attacks are presented in detail. Chapter 3 contains cryptographic primitives and model where our scheme is constructed. We formally state our definition of security as well as basic tools used in our scheme in Chapter 4. Our basic identification scheme is presented based on the bilinear Diffie-Hellman problem and then we transform it into a generalized scheme in Chapter 5. In Chapter 6 we give a proof of security for our scheme. In Chapter 7 we make a comparison with several existing identification schemes and present an exact quantification of resource requirement. In what follows, we compare with other schemes in the light of performance, and end with concluding remarks in Chapter 8.

Chapter 2
Related work

2.1 Relation between identification and signature

Identification schemes are closely related to, but simpler than, digital signature schemes [26], which involves a variable message and typically provide a non-repudiation feature allowing disputes to be resolved by judges after the fact. For identification schemes, the semantics of the message are essentially fixed – a claimed identity at the current instant in time. The claim is either corroborated or rejected immediately, with associated privileges or access either granted or denied in real time. Identifications do not have "lifetimes" as signatures do – disputes need not typically be resolved afterwards regarding a prior identification, and attacks which may become feasible in the future do not affect the validity of a prior identification.

Hence, if we replace "identity" by "authenticity" of messages, identification schemes are nearly equivalent to *signature schemes*. As mentioned by Fiat and Shamir [15] and Shoup [37], the distinction between identification and signature schemes is very subtle. Therefore, two types of schemes can be used interchangeably [15, 21, 28, 29]. With a little additional process, in general we can convert an identification scheme involving a witness-challenge-response sequence to a signature scheme.

Now let's turn to the underlying hard problem. Since Okamoto and Pointcheval [30] initiated the concept of the Gap-problems and proposed that a gap Diffie-Hellman ($\mathsf{G\text{-}DH}$) problem offers a signature scheme, several cryptographic schemes based on such variants of Diffie-Hellman (DH) assumption have been studied. Using the bilinear Diffie-Hellman ($\mathsf{B\text{-}DH}$) problem as an instance of the ($\mathsf{G\text{-}DH}$) problem, Boneh and Franklin [5] and Boneh *et al.* [6] suggested an efficient ID-based encryption (IBE) scheme and a short signature scheme, respectively. These imply that the $\mathsf{B\text{-}DH}$ problem provides identification schemes.

2.2 Types of attacks

In general, an identification scheme is said to *be broken if an adversary succeeds in an impersonation attempt* (making the verifier accept with non-negligible probability). The methods an adversary may employ in an attempt to defeat identification protocol are summarized in Table 2.1 [26]. We can divide them into two types–passive attack and active attack–according to the interaction allowed to the adversary before an impersonation attempt [37, 26].

The weakest form of attack is a *passive attack*, where the adversary is not allowed to interact with the system at all before attempting an impersonation; the only available information to the adversary is the public key of the prover. Other attacks of intermediate level such as *eavesdropping attack* or *honest-verifier attack* are essentially equivalent to the passive attack.

The strongest form of attack is an *active attack*, in which the adversary is allowed to interact with $\mathcal{P}$ several times, posing as $\mathcal{V}$. We may consider active attacks as adaptive chosen ciphertext attacks. We should note that active attacks are quite feasible in practice.

2.3 Fiat-Shamir (FS) scheme

Fiat and Shamir [15] proposed the identification scheme based on the factorization problem. A key generation algorithm constructs a modulus n by

Table 2.1: Types of attacks on identification protocols

Types of attacks	Descriptions
impersonation	a deception whereby one entity purports to be another.
replay attack	an impersonation or other deception involving use of information from a single previous protocol execution, on the same time or a different verifier.
interleaving attack	an impersonation or other deception involving selective combination of information from one or more previous or simultaneously ongoing protocol executions, including possible origination of one or more protocol executions by an adversary itself.
reflection attack	an interleaving attack involving sending information from an ongoing protocol execution back to the originator of such information
forced delay	a forced delay occurs when an adversary intercepts a message, and relays it at some later point in time.
chosen-text attack	an attack on a challenge-response protocol wherein an adversary strategically chooses challenges in an attempt to extract information about the prover's long-term key.

multiplying two distinct random primes, chooses randomly an element $a \in \mathbb{Z}_n^*$, and sets $b = a^2$. The public key is $\langle b, n \rangle$, and the secret key is a.

The protocol repeats the following steps t times:

1. $\mathcal{P}$ chooses $r \in \mathbb{Z}_n^*$ at random, computes $x = r^2$, and sends x to $\mathcal{V}$.

2. $\mathcal{V}$ chooses $\epsilon \in \{0, 1\}$ at random, and sends ϵ to $\mathcal{P}$.

3. $\mathcal{P}$ computes $y = r \cdot a^\epsilon$ and sends y to $\mathcal{V}$; $\mathcal{V}$ accepts if $y^2 = x \cdot b^\epsilon$, and rejects otherwise.

The FS scheme is secure against active attacks if factorization is hard. The FS scheme works as depicted on Figure 2.1.

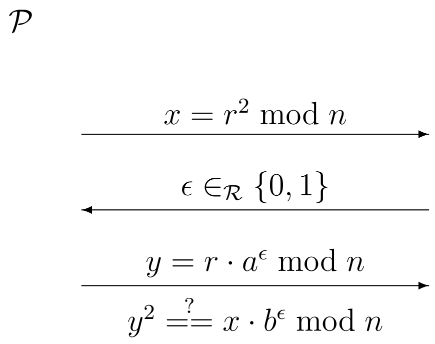
$\mathcal{P}$
$\mathcal{V}$
$x = r^2 \bmod n$
$\epsilon \in_{\mathcal{R}} \{0, 1\}$
$y = r \cdot a^\epsilon \bmod n$
$y^2 \overset{?}{==} x \cdot b^\epsilon \bmod n$

Figure 2.1: The Fiat-Shamir identification protocol

2.4 Feige-Fiat-Shamir (FFS) scheme

This scheme is also based on the factorization problem. A key generation algorithm chooses a modulus n as in the FS scheme. A secret key consists of a list $a_1, \ldots, a_l \in \mathbb{Z}_n^*$ chosen randomly, where l is a given constant, and the corresponding public key consists of $b_1, \ldots, b_l \in \mathbb{Z}_n^*$, where $b_i = a_i^2$ for $1 \leq i \leq l$.

The protocol executes the followings t times in parallel:

1. $\mathcal{P}$ chooses $r \in \mathbb{Z}_n^*$ at random, computes $x = r^2$, and sends x to $\mathcal{V}$.

2. $\mathcal{V}$ randomly chooses $\epsilon_1, \ldots, \epsilon_l \in \{0, 1\}$, sends $\epsilon_1, \ldots, \epsilon_l$ to $\mathcal{P}$.

3. $\mathcal{P}$ computes $y = r \prod_{j=1}^{l} a_j^{\epsilon_j}$ and sends y to $\mathcal{V}$; $\mathcal{V}$ accepts if $x = y^2 \prod_{j=1}^{l} b_j^{\epsilon_j}$, and rejects otherwise.

This scheme is also secure against active attacks if factorization is hard [14]. Figure 2.2 show how the FFS protocol works.

$$\begin{array}{ccc}
\mathcal{P} & & \mathcal{V} \\
& \xrightarrow{x = r^2 \bmod n} & \\
& \xleftarrow{\epsilon_1, \ldots, \epsilon_l \in_{\mathcal{R}} \{0, 1\}} & \\
& \xrightarrow[x \overset{?}{==} y^2 \cdot \prod_{j=1}^{l} b_j^{\epsilon_j} \bmod n]{y = r \cdot \prod_{j=1}^{l} a_j^{\epsilon_j} \bmod n} &
\end{array}$$

Figure 2.2: The Feige-Fiat-Shamir identification protocol

2.5 Other schemes

2.5.1 Variants of the FS scheme

The Guillou-Quisquater (GQ) scheme [21] is an extension of the FS protocol. It allows a reduction in both the number of messages exchanged and memory requirements for user secrets and, like Fiat-Shamir, is suitable for applications on which the prover has limited power and memory. A modification of FS identification by Ong and Schnorr [31] decreases computational complexity, signature size, and the number of communication required, condensing t Fiat-Shamir iterations into one iteration while leaving each user with k private key sizes. The Ohta-Okamoto (OO) version of the extended FS scheme differs from the GQ versions as follows[1]: (1) in OO, rather than TTP computing $S_{\mathcal{P}}$ from identity $I_{\mathcal{P}}$, $\mathcal{P}$ chooses its own secret $S_{\mathcal{P}} \in \mathbb{Z}_n$ and publishes $I_{\mathcal{P}} = S_{\mathcal{P}}^v \bmod n$; and (2) the verification relation $x = J_{\mathcal{P}}^{\epsilon} \cdot y^v \bmod n$ becomes $y^v \equiv x \cdot I_{\mathcal{P}}^{\epsilon}$. A further subsequent version of extended FS scheme by Okamoto [29] is provably as secure as factoring, only slightly less efficient, and amenable to an identity-based variation. proposed in [29].

2.5.2 The Schnorr scheme and its variants

The Schnorr protocol [35] is an alternative of the FS and GQ protocols whose security is based on the intractability of DLP. The design allows pre-computation, reducing the real-time computation for the prover to one multiplication modulo a prime q; it is particularly suitable for provers of limited computational ability. A further important computational efficiency results from the use of a subgroup of order q of the multiplicative group of integers modulo p, where $q|(p-1)$; this also reduces the required number of transmitted bits. Finally, the protocol was designed to require only three passes. The Schnorr protocol is depicted on Figure 2.3. Brickell and McCurley [8] propose a modification of Schnorr's identification scheme, in which q is kept secret and exponent computations are reduced modulo $p-1$ rather than q. A major drawback is that almost 4 times as much computation is required by the prover. Another variant of Schnorr's scheme by Girault [16] was the first identity-based identification scheme based on DLP. A further variation of Schnorr's identification protocol by Okamoto [29] is provably secure; it does, however, involve some additional computation. Popescu [33] shows how the interactive identification scheme based on the elliptic curve discrete logarithm

[1] The notations of the OO and GQ schemes follows those of [26]

problem (ECDLP) is constructed.

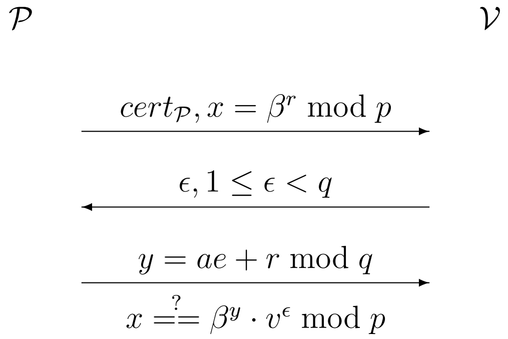
$\mathcal{P}$
$\mathcal{V}$
$cert_{\mathcal{P}}, x = \beta^r \bmod p$
$\epsilon, 1 \leq \epsilon < q$
$y = ae + r \bmod q$
$x \overset{?}{==} \beta^y \cdot v^\epsilon \bmod p$

Figure 2.3: The Schnorr identification protocol

Aside from the above protocols based on the computational intractability of the standard number-theoretic problems, a number of very efficient identification protocols have more recently been proposed based on **NP**-hard problems. Stern [39] proposed a practical zero-knowledge identification scheme based on the **NP** hard *syndrome decoding* problem. Stern [40] proposed another practical identification scheme based on an **NP** hard combinatorial *constrained linear equations* problem, offering a very short key length, which is of particular interest in specific applications. Pointcheval [32] proposed another such scheme based on the **NP**-hard *perceptrons problem*: given an $m \times n$ matrix M with entries ± 1, find an n-vector y with entries ± 1 such that $M_y \geq 0$.

Chapter 3
Preliminaries

3.1 Concrete security

In this thesis, we develop proofs in the frame work of concrete provable security. We provide an exact analysis of the security of the schemes rather than asymptotic approach. That is, we explicitly quantify the *reduction* from the security of a scheme to the security of the underlying "hard" problem on which is based. This enables us to know exactly how much security is maintained by the reduction and thus to determine the strength of the reduction.

In order to quantify the reduction, we define the advantage $\mathsf{Adv}(\mathcal{I})$ that a computationally bounded adversary $\mathcal{I}$ will defeat the security goal of an identification protocol. The advantage is twice the probability that $\mathcal{I}$ will defeat the security goal of the protocol minus one.

3.2 Random oracle model vs. Standard model

In general, to show a cryptosystem is secure cryptographers choose a method for analyzing the security of the cryptosystem. Methods for cryptographers to pick out are divided into three as follows: the *ad hoc* model, the random oracle model, and the standard model.

3.2.1 The *ad hoc* model

Throw in some random padding here, some hash functions there, until one starts *feel good* about it. See if it withstands a few obvious attacks. Then deploy the system, wait for it to get broken, and add some more padding and hashes. Repeat.

Clearly, this approach leaves much to be desired. Even if the cryptosystem is built out of "cryptographically strong" components, these components may interact in some hard-to-predict ways that allow an attacker to break the cryptosystem.

3.2.2 The random oracle model (ROM)

As aforementioned, designing cryptosystems and proving them secure is no easy task, in particular if one wants to have a practical cryptosystem.

To make this task more manageable, Bellare and Rogaway [2] use the notion of a *random oracle model* (ROM). The result of this approach is a reductionist proof, however the proof is only valid in a "parallel universe" where a "magic hash functions" exist—they *do not* exist in the "real world" of computation. We stress that the existence of magic hash functions is not a "hardness assumption," like IFP and DLP; they simply do not exist. Rather, they are a rough-and-ready *heuristic*, much like assuming the earth is flat, and that there is no wind resistance.

To analyze a protocol using ROM one replaces a real-world cryptographic hash function by a *black-box* that when queried outputs a *random bit string*, subject to the restriction that it always outputs the same value on the same input. Having made this replacement, one then gives a reductionist security argument. The right way to view a proof of security in ROM is as a proof of security against a restricted class of adversaries that do not care if the hash function really is a black box. Canetti [9, 10, 11] also pointed out these problems, however many cryptographers including [3, 5, 6] give provable security against the attacks in ROM.

3.2.3 The standard model

This is the preferred approach of modern, mathematical cryptography. Here, one shows with mathematical rigor that any attacker who can break the cryptosystem can be transformed into an efficient algorithm to solve the underlying well-studied problem that is widely believed to be very hard. Turing this logic around: if the "hardness assumption" is correct as presumed, the cryptosystem is secure [36, 23].

This approach is about the best we can do. If we can prove security in this way, then we essentially rule out all possible shortcuts, even ones "we have not yet even imagined." The only way to attack the cryptosystem is a full-frontal attack on the underlying hard problem. This approach is taken in [13, 12, 37, 11]. With this approach, we analyze the adversary to a tightly bounded quantity and quantify the precise resource for him.

3.3 The Weil paring

We can make use of any bilinear map on an elliptic curve to construct a group $\mathbb{G}$ in which the C-DH problem is intractable, but the D-DH problem is tractable [22, 5, 6]. In particular, we make use of bilinear maps, in particular the Weil-pairing.

Let E be an elliptic curve over a base field K and let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two cyclic groups of order m for some large prime m. The *Weil pairing* [38, 24, 4, 5, 6] is defined by a bilinear map e between these groups,

$$e_m : E[m] \times E[m] \longrightarrow \mu_m,$$

where $E[m]$ corresponds to the additive group of points of E/K, and μ_m corresponds to the multiplicative group of an extension field $\overline{K}$ of K. We can define the Weil pairing as follows. Let $S, T \in E[m]$ and choose a function g on E whose divisor satisfies

$$\operatorname{div}(g) = \sum_{R \in E[m]} (T' + R) - (R),$$

with $T' \in E(\overline{K})$ such that $[m]T' = T$. Then

$$e_m = \begin{cases} E[m] \times E[m] & \longrightarrow & \mu_m \\ \mathbb{G}_1 \times \mathbb{G}_1 & \longrightarrow & \mathbb{G}_2 \\ (S, T) & \longmapsto & \frac{g(X+S)}{g(X)} \end{cases}$$

for any point $X \in E(K)$ for which g is both defined and non-zero at X and $X + S$. It can then be shown that the following properties hold. Let $P, Q \in \mathbb{G}_1$.

(i) *Identity*: For all $P \in E[m]$, $e_m(P, P) = 1$.

(ii) *Alternation*: For all $P, Q \in E[m]$, $e_m(P, Q) = e_m(Q, P)^{-1}$.

(iii) *Bilinearity*: For all $P, Q, R \in E[m]$, $e_m(P+Q, R) = e_m(P, R) \cdot e_m(Q, R)$ and $e_m(P, Q + R) = e_m(P, Q) \cdot e_m(P, R)$.

(iv) *Non-degeneracy*: If $e_m(P, Q) = 1$ for all $Q \in \mathbb{G}_1$, then $P = \mathcal{O}$, where $\mathcal{O}$ is a point at infinity.

(v) If $E[m] \subset E(K)$, then $e_m(P, Q) \in K$ for all $P, Q \in E[m]$ (that is $\mu_m \subset K^*$).

(vi) *Compatible*: If $P \in E[m]$ and $Q \in E[mm']$, then $e_{mm'}(P, Q) = e_m(P, m'Q)$.

In addition to these properties, we have an efficient algorithm to compute $e_m(P, Q)$ for all $P, Q \in E[m]$ by [27]. In practice, in our basic scheme, we employ the *modified Weil pairing* $\hat{e}_m(P, Q) = e_m(P, \phi(Q))$, where ϕ is an automorphism on the group of points of E [5, 6]. For more details, we can refer to [4], [5], and [24].

Throughout this thesis, the group $E[m]$ is written by $\mathbb{G}_1$, the group μ_m is written by $\mathbb{G}_2$. For the sake of convenience, we write the Weil pairing and the modified Weil pairing as e and $\hat{e}$ in the place of e_m and $\hat{e}_m$ respectively.

As noted in [5], the existence of the bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ as above has two direct implications to these groups.

The MOV reduction: Menezes, Okamoto, and Vanstone[25] show that DLP in $\mathbb{G}_1$ is no harder than DLP in $\mathbb{G}_2$. To see this, let $P, Q \in \mathbb{G}_1$ be an instance of DLP in $\mathbb{G}_1$ where both P, Q have order m. We wish to find an $\alpha \in \mathbb{Z}_m$ such that $Q = \alpha P$. Let $g = \hat{e}(P, P)$ and $h = \hat{e}(Q, P)$. Then, by bilinearity of $\hat{e}$ we know that $h = g^\alpha$. By non-degeneracy of $\hat{e}$ both g and h have order m in $\mathbb{G}_2$. Hence, we reduced DLP in $\mathbb{G}_1$ to DLP in $\mathbb{G}_2$. It follows that for discrete log to be hard in $\mathbb{G}_1$ we must choose our security parameter so that discrete log hard in $\mathbb{G}_2$.

Decision DH is easy: The D-DH problem [7] in $\mathbb{G}_1$ is to distinguish between the distributions $\langle P, aP, bP, abP\rangle$ and $\langle P, aP, bP, cP\rangle$ where a, b, and c are random in $\mathbb{Z}_m$ and P is random in $\mathbb{G}_1$. Joux and Nguyen [22] point out that D-DH in $\mathbb{G}_1$ is easy. To see this, observe that given $\{P, aP, bP, cP\} \in \mathbb{G}_1^*$ we have

$$c = ab \bmod m \Longleftrightarrow \hat{e}(P, cP) = \hat{e}(aP, bP).$$

The C-DH problem in $\mathbb{G}_1$ can still be hard. Joux and Nguyen [22] give examples of mappings $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ where C-DH in $\mathbb{G}_1$ is believed to be hard even though D-DH in $\mathbb{G}_1$ is easy.

3.4 The bilinear Diffie-Hellman problem

3.4.1 Gap-problems

The computational assumptions when constructing cryptographic schemes can mainly be classified into two types. One is the intractability of an inverting problem such as inverting the RSA function, and computing the Diffie-Hellman (DH) problem. The other is the intractability of a decision problem such as the decisional Diffie-Hellman problem.

In addition to these problems, Okamoto and Pointcheval [30] define a new class of problems, called the Gap-problems. Let $f : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ be any relation. The inverting problem of f is the classical computational version, and we can define a generalization of the decision problem, by the R-decision problem of f, for any relation

$$R : \{0,1\}^* \times \{0,1\}^* \times \{0,1\}^* \to \{0,1\},$$

- The *inverting problem* of f is, given x, to compute any y such as $f(x,y) = 1$ if it exists, or to answer Fail.

- The *R-decision problem* of f is, given (x,y), to decide whether $R(f,x,y) = 1$ or not. Here y may be the null string, $\perp$.

Let us see some examples for the relation, R_1, R_2, R_3, and R_4:

- $R_1(f,x,y) = 1$ iff $f(x,y) = 1$, which formalizes the classical version of decision problems [7].

- $R_2(f,x,\perp) = 1$ iff there exists any z such that $f(x,z) = 1$, which simply answers whether the inverting problem has a solution or not.

- $R_3(f,x,\perp) = 1$ iff z is even, when z such that $f(x,z) = 1$ is uniquely defined. This latter example models the least-significant bit of the pre-image, which is used in many hard-core bit problems.

- $R_4(f,x,\perp) = 1$ iff all the z such that $f(x,z) = 1$ are even.

It is often the case that the inverting problem is strictly stronger than the R-decision problem, namely for all the classical examples we have cryptographic purpose. However, it is not always the case, and the R-decision problem can even be strictly stronger than the inverting one. Informally the Gap-problem deals with the gap of difficulty between these problems. The Gap-problem can be defined as follows:

Definition 3.1 *The R-*gap problem *of f is to solve the inverting problem of f with the help of the oracle of the R-decision problem of f.*

Okamoto and Pointcheval [30] claimed that the DH problems are the typical instance of the Gap-problem. Since the inverting problem can be viewed as the computational problem, the computational Diffie-Hellman (C-DH) problem corresponds to the inverting one, and the decisional Diffie-Hellman (D-DH) problem does to the R-decision one. Here, we describe the gap Diffie-Hellman (G-DH) problem. Let $\mathbb{G}$ be any group of prime order m.

- The C-DH problem: given a triple of $\mathbb{G}$ elements (g, g^a, g^b), find the element $C = g^{ab}$.
- The D-DH problem: given a quadruple of $\mathbb{G}$ elements (g, g^a, g^b, g^c), decide whether $c = ab \pmod{q}$ or not.
- The G-DH problem: given a triple of $\mathbb{G}$ elements (g, g^a, g^b), find the element $C = g^{ab}$ with the help of a D-DH oracle (which answers whether a given quadruple is a DH quadruple or not).

The Tate-pairing is given as a specific example that satisfies the property of the G-DH problem [30]. For example [30], with an elliptic curve $E = J(\mathbb{F}_q)$ of trace $t = 2$ and $m = \#E = q + 1 - t = q - 1$, we have $J_m(\mathbb{F}_q) = J(\mathbb{F}_q)/mJ(\mathbb{F}_q) = E$ and $\mu_m(\mathbb{F}_q) = \mathbb{F}_q^*$. Then

$$e : E \times E \to \mathbb{F}_q^*,$$

which is called officially a bilinear map. We will discuss the bilinear map in the next section. Let us consider a DH quadruple, P, $A = a \cdot P$, $B = b \cdot P$ and $C = c \cdot P$,

$$e(A, B) = e(a \cdot P, b \cdot P) = e(P, P)^{ab} = e(P, ab \cdot P) = e(P, C).$$

And the latter equality only holds with the correct candidate C.

3.4.2 The bilinear Diffie-Hellman problem

Since the $\mathsf{D\text{-}DH}$ problem in $\mathbb{G}_1$ is easy, we cannot use the $\mathsf{D\text{-}DH}$ problem to build cryptosystems in the group $\mathbb{G}_1$. Instead, the security of our protocol is based on a variant of the $\mathsf{C\text{-}DH}$ problem called the bilinear Diffie-Hellman ($\mathsf{B\text{-}DH}$) problem.

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two cyclic groups of prime order m and let P be a generator of $\mathbb{G}_1$. Let $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be a bilinear map.

Definition 3.2 *The* $\mathsf{B\text{-}DH}$ problem *in* $(\mathbb{G}_1, \mathbb{G}_2, \hat{e})$ *is the following: given* (P, aP, bP, cP) *for some* $a, b, c \in \mathbb{Z}_m^*$*, compute* $v \in \mathbb{G}_2$ *such that* $v = \hat{e}(P, P)^{abc}$.

$\mathsf{B\text{-}DH}$ parameter generator: We say that a randomized algorithm $\mathcal{IG}$ is a $\mathsf{B\text{-}DH}$ *parameter generator* if

(1) $\mathcal{IG}$ takes a security parameter $0 < k \in \mathbb{Z}$,

(2) $\mathcal{IG}$ runs in polynomial time in k, and

(3) $\mathcal{IG}$ outputs the description of two groups $\mathbb{G}_1, \mathbb{G}_2$ and the description of a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$.

We require that the groups have the same prime order $m = |\mathbb{G}_1| = |\mathbb{G}_2|$. We denote the output of $\mathcal{IG}$ by $\mathcal{IG}(1^k)$. A concrete example of the $\mathsf{B\text{-}DH}$ parameter generator is given in [5] as follows. Given a security parameter k the $\mathsf{B\text{-}DH}$ parameter generator picks a random k-bit prime n such that $n = 2 \bmod 3$ and $n = 6m - 1$ for some prime m. The group $\mathbb{G}_1$ is the subgroup of order m of the group of points on the elliptic curve $y^2 = x^3 + 1$ over $\mathbb{F}_n$. The group of $\mathbb{G}_2$ is the subgroup of order m of $\mathbb{F}_{n^2}^*$. The bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ is the modified Weil paring defined in Section 3.3.

Note that the isomorphisms from $\mathbb{G}_1$ to $\mathbb{G}_2$ induced by the Weil pairing are one-way functions [5, 6]. For a point $Q \in \mathbb{G}_1^*$ defines the isomorphism $f_Q : \mathbb{G}_1 \rightarrow \mathbb{G}_2$ by $f_Q(P) = \hat{e}(P, Q)$. It is well known that an efficient algorithm for inverting f_Q would imply an efficient algorithm for deciding D-DH in the group $\mathbb{G}_2$. Throughout this thesis the D-DH problem is believed to be hard in the group $\mathbb{G}_2$. Hence, all the isomorphisms $f_Q : \mathbb{G}_1 \rightarrow \mathbb{G}_2$ are believed to be one-way functions.

Chapter 4
Definitions of security

A general approach of proving that an identification scheme is secure is to show that the scheme exhibits a zero-knowledge proof of knowledge. However, the results of Goldreich and Krawczyk [19], together with the argument of Shoup [37] say that any efficient black box simulator for a three round, public coin system can be turned into a prover that succeeds with non-negligible probability.

In this thesis, we make use of a computational reduction from solving a well-established problem to break the cryptosystem rather than zero-knowledge proof techniques. That is to say, the proving method is to use an adversary that breaks the cryptosystem to solve the computational Diffie-Hellman problem.

4.1 Notions of security

We formally define a secure identification scheme, using the same notations as in [34, 17, 18, 37].

If $A(\cdot)$ is a probabilistic algorithm, then for any input x, the notation A_x refers to the probability space that assigns to the string σ the probability space that A, on input x, outputs σ.

If $\mathcal{S}$ is a probability space, then $[\mathcal{S}]$ denotes the set of elements in this space that occur with non-zero probability, and $\Pr_{\mathcal{S}}[x]$ denotes the probability that $\mathcal{S}$ associates with the element x. If $\mathcal{S}$ is any probability space, then

$x \leftarrow \mathcal{S}$ denotes the algorithm which assigns to x an element randomly selected according to $\mathcal{S}$.

The notation $\Pr[p(x_1, x_2 \ldots)|x_1 \leftarrow \mathcal{S}_1; x_2 \leftarrow \mathcal{S}_2; \ldots]$ denotes the probability that the predicate $p(x_1, x_2, \ldots)$ will be true after the ordered execution of the algorithms $x_1 \leftarrow \mathcal{S}_1, x_2 \leftarrow \mathcal{S}_2, \ldots$.

In addition, we use the same conventions in [14]:

1. $\bar{\mathcal{P}}$ represents an honest prover who follows its designated protocol, $\tilde{\mathcal{P}}$ does a polynomial-time cheater, and $\mathcal{P}$ acts as $\bar{\mathcal{P}}$ or $\tilde{\mathcal{P}}$.

2. $\bar{\mathcal{V}}$ represents a valid verifier who follows the designated protocol, $\tilde{\mathcal{P}}$ does an arbitrary polynomial-time algorithm which may try to extract additional information from $\mathcal{P}$, and $\mathcal{V}$ acts as $\bar{\mathcal{V}}$ or $\tilde{\mathcal{V}}$.

3. $(\mathcal{P}, \mathcal{V})$ represents the execution of the two party protocol where $\mathcal{P}$ is the prover and $\mathcal{V}$ is the verifier.

4.2 The bilinear Diffie-Hellman assumption

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two cyclic groups of prime order m and let P be a generator of $\mathbb{G}_1$. Let $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be a modified bilinear map.

Definition 4.1 *An algorithm $\mathcal{A}$ has an* advantage $\mathsf{Adv}^{\mathsf{B\text{-}DH}}(\mathcal{A}) = \epsilon$ *in solving* B-DH *in $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e} \rangle$ if*

$$\mathsf{Adv}^{\mathsf{B\text{-}DH}}(\mathcal{A}) \triangleq \Pr\left[\mathcal{A}(P, aP, bP, cP) = \hat{e}(P, P)^{abc}\right] \geq \epsilon,$$

where the probability is over the random choice of $\langle a, b, c \rangle \in \mathbb{Z}_m^*$, the random choice of $P \in \mathbb{G}_1^*$, and the random bits of $\mathcal{A}$.

The security of our identification scheme is intrinsically based on the intractability of the B-DH problem. We formally describe this assumption as follows, called as it the bilinear Diffie-Hellman intractability assumption (B-DHIA).

An (τ, ϵ)-$\mathsf{B\text{-}DH}$-attacker for the groups is a PPT algorithm Δ running in time τ that given a $\mathsf{B\text{-}DH}$ parameter generator $\mathcal{IG}$ stated in Section 3.4 solves the $\mathsf{B\text{-}DH}$ problem if for sufficiently large k:

$$\Pr\left[\ \Delta(\mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, aP, bP, cP) = \hat{e}(P,P)^{abc}\ \middle|\ \begin{array}{l} \langle \mathbb{G}_1, \mathbb{G}_2, \hat{e} \leftarrow \mathcal{IG}(1^k); \\ P \leftarrow \mathbb{G}_1^*; \\ \langle a, b, c \rangle \leftarrow \mathbb{Z}_m^* \end{array}\ \right] \geq \epsilon.$$

We denote this probability as $\mathsf{Succ}_{\mathcal{IG}}^{\mathsf{B\text{-}DH}}(\Delta)$.

Definition 4.2 (B-DHIA) *Given a* $\mathsf{B\text{-}DH}$ *parameter generator* $\mathcal{IG}$ *the* $\mathsf{B\text{-}DH}$ *problem is* (τ, ϵ)-intractable *if there is no* (τ, ϵ)*-attacker* Δ *for the groups.*

4.3 Secure identification schemes against active attacks

In general, an identification scheme $(\mathcal{G}, \mathcal{P}, \mathcal{V})$ consists of a PPT algorithm $\mathcal{G}$, and two PPT interactive algorithms $\mathcal{P}$ and $\mathcal{V}$. An identification scheme is defined by the following [14, 37]:

1. The algorithm $\mathcal{G}$ is a *key generation algorithm*. It takes a string of the form 1^k as input, and outputs a pair of string $(I_{\mathcal{P}}, S_{\mathcal{P}})$. k is called a security parameter, $I_{\mathcal{P}}$ is called a *public key*, and $S_{\mathcal{P}}$ is called a *secret key*.

2. As input, $\mathcal{P}$ receives the pair $(I_{\mathcal{P}}, S_{\mathcal{P}})$ and $\mathcal{V}$ does I. After an interactive execution of $\mathcal{P}$ and $\mathcal{V}$, $\mathcal{V}$ outputs either 1 (indicating `"accept"`) or 0 (indicating `"reject"`). For given $I_{\mathcal{P}}$ and $S_{\mathcal{P}}$, the output of $\mathcal{V}$ at the end of this interaction is a probability space which is denoted by $\langle \mathcal{P}(I_{\mathcal{P}}, S_{\mathcal{P}}), \mathcal{V}(I_{\mathcal{P}}) \rangle$.

3. A valid prover should always be able to succeed in convincing the verifier. Formally speaking, for all k and for all $(I_{\mathcal{P}}, S_{\mathcal{P}}) \in [\mathcal{G}(1^k)]$, $\langle \mathcal{P}(I_{\mathcal{P}}, S_{\mathcal{P}}), \mathcal{V}(I_{\mathcal{P}}) \rangle = 1$ with probability 1.

An *adversary* $(\tilde{\mathcal{P}}, \tilde{\mathcal{V}})$ is a pair of probabilistic polynomial-time interactive algorithms. For given key pair $(I_{\mathcal{P}}, S_{\mathcal{P}})$, we denote by $\langle \bar{\mathcal{P}}(I_{\mathcal{P}}, S_{\mathcal{P}}), \tilde{\mathcal{V}}(I_{\mathcal{P}}) \rangle$ the string h output by $\tilde{\mathcal{V}}$ after interacting with $\bar{\mathcal{P}}$ several times. For given $I_{\mathcal{P}}$ and $S_{\mathcal{P}}$, yet again $\langle \bar{\mathcal{P}}(I_{\mathcal{P}}, S_{\mathcal{P}}), \tilde{\mathcal{V}}(I_{\mathcal{P}}) \rangle$ is a probability space. The string h (called a `"help string"`) is used as input to $\tilde{\mathcal{P}}$ who attempts to convince $\bar{\mathcal{V}}$. We denote by $\langle \tilde{\mathcal{P}}(h), \bar{\mathcal{V}}(I_{\mathcal{P}}) \rangle$ the output of $\bar{\mathcal{V}}$ after interacting with $\tilde{\mathcal{P}}(h)$.

We adopt the definition of *security against active attacks* (SAA) with respect to such adversaries from [37] as follows.

Definition 4.3 *The advantage in breaking an identification scheme of an adversary $\mathcal{I} = (\tilde{\mathcal{P}}, \tilde{\mathcal{V}})$ is*

$$\mathsf{Adv}^{\mathsf{SAA}}(\mathcal{I}) \triangleq \Pr\left[\sigma = 1 \;\middle|\; \begin{array}{l} (I, S) \leftarrow \mathcal{G}(1^k); \\ h \leftarrow \langle \bar{\mathcal{P}}(I_{\mathcal{P}}, S_{\mathcal{P}}), \tilde{\mathcal{V}}(I_{\mathcal{P}}) \rangle; \\ \sigma \leftarrow \langle \tilde{\mathcal{P}}(h), \bar{\mathcal{V}}(I_{\mathcal{P}}) \rangle \end{array} \right].$$

The probability is taken oven the coin tosses of the key generation algorithm $\mathcal{G}$, and of the adversary.

Definition 4.4 *An adversary $\mathcal{I}$ (τ, ϵ)-breaks an identification scheme if $\mathcal{I} = (\tilde{\mathcal{P}}, \tilde{\mathcal{V}})$ runs in time τ and the advantage $\mathsf{Adv}^{\mathsf{SAA}}(\mathcal{I}) \geq \epsilon$.*

Now we can make a secure definition for an identification scheme against active adversaries.

Definition 4.5 *An identification scheme $(\mathcal{G}, \mathcal{P}, \mathcal{V})$ is* secure against active attacks *if for all adversaries $\mathcal{I} = (\tilde{\mathcal{P}}, \tilde{\mathcal{V}})$, for all constants $c > 0$, and for all sufficiently large k, there is no adversary which can get a probability than ϵ in mounting an active attack within time τ.*

We denote this probability ϵ as $\mathsf{Succ}^{\mathsf{SAA}}(\mathcal{I})$.

4.4 Adversary's resources

The security is formulated as a function of the amount of resources the adversary $\mathcal{I} = (\tilde{\mathcal{P}}, \tilde{\mathcal{V}})$ expends. The resource are:

- $T_{\mathcal{V}}(k)$: a time bound required for $\tilde{\mathcal{V}}$ to run the protocol once with $\bar{\mathcal{P}}$ including $\bar{\mathcal{P}}$'s computing time.
- $N_{\mathcal{V}}(k)$: an iteration bound for $\tilde{\mathcal{V}}$ to run the protocol with $\bar{\mathcal{P}}$.
- $T_{\mathsf{off}}(k)$: an off-line time bound for $\tilde{\mathcal{V}}$ to spend other than running the protocol with $\bar{\mathcal{P}}$.
- $T_{\mathcal{P}}(k)$: a time bound for $\tilde{\mathcal{P}}$ to run the protocol with $\bar{\mathcal{V}}$.

By notation $\mathsf{Adv}(\tau, \ldots)$ or $\mathsf{Succ}(\tau, \ldots)$, we mean the maximum values of $\mathsf{Adv}(\mathcal{I})$ or $\mathsf{Succ}(\mathcal{I})$ respectively, over all adversaries $\mathcal{I} = (\tilde{\mathcal{P}}, \tilde{\mathcal{V}})$ expends at most the specified amount of resources.

Chapter 5
The protocol

5.1 Basic identification scheme

For a security parameter k, a pair of secret and public parameters is generated as follows:

Key generation.
On input k, the key generation algorithm $\mathcal{G}$ works as follows:

1. Generate two cyclic groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of order m for some large prime m and a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$.
2. Generate an arbitrary generator $P \in \mathbb{G}_1$.
3. Choose randomly $a, b, c \in \mathbb{Z}_m^*$ and compute $v = \hat{e}(P, P)^{abc}$.
4. The public parameter is $\mathsf{Pub} = \langle \mathbb{G}_1, \mathbb{G}_2, P, aP, bP, cP, \hat{e}, v \rangle$, and the secret parameter is $\mathsf{Sec} = \langle a, b, c \rangle$. And then publish them.

Protocol actions between $\mathcal{P}$ and $\mathcal{V}$.
As is the case for other identification schemes, this scheme consists of several rounds. The protocol executes just once the following:

1. $\mathcal{P}$ chooses $r_1, r_2, r_3 \in \mathbb{Z}_m^*$ at random, computes $x = \hat{e}(P, P)^{r_1 r_2 r_3}$, $Q = r_1 r_2 r_3 P$, and sends $\langle x, Q \rangle$ to $\mathcal{V}$.

2. $\mathcal{V}$ picks $\omega \in \mathbb{Z}_m^*$ at random, and sends $R = \omega P$ to $\mathcal{P}$.

3. On receiving R, $\mathcal{P}$ sets $S = r_1 r_2 r_3 R$, computes $Y \in \mathbb{G}_1$ such that

$$Y = abcP + (a + b + c)S,$$

and sends it to $\mathcal{V}$; $\mathcal{V}$ accepts $\mathcal{P}$'s proof of identity if both $x = \hat{e}(P, Q)$ and $\hat{e}(Y, P) = v \cdot \hat{e}(aP + bP + cP, Q)^{\omega}$, and rejects otherwise.

This protocol is represented graphically in Figure 5.1. Once after this protocol can be proved to be secure against active adversaries, it can be extended to a generalized protocol.

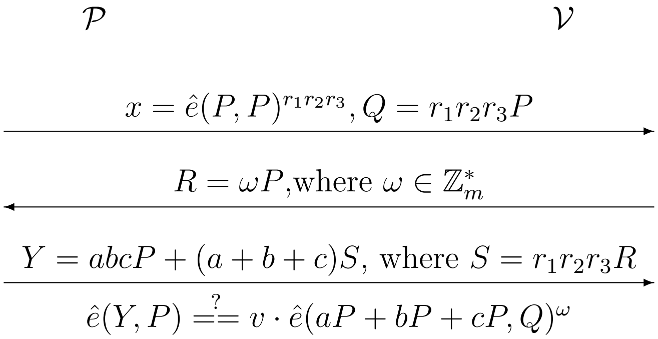
$\mathcal{P}$
$\mathcal{V}$
$x = \hat{e}(P, P)^{r_1 r_2 r_3}, Q = r_1 r_2 r_3 P$
$R = \omega P$,where $\omega \in \mathbb{Z}_m^*$
$Y = abcP + (a + b + c)S$, where $S = r_1 r_2 r_3 R$
$\hat{e}(Y, P) \overset{?}{==} v \cdot \hat{e}(aP + bP + cP, Q)^{\omega}$

Figure 5.1: The SAA identification protocol

5.2 Generalized scheme

We now describe a generalized model of the basic identification scheme. The generalized identification scheme extends the basic scheme in Section 5.1 using k random numbers. The key generation algorithm $\mathcal{G}$ is similar to that of the basic scheme except generating k random numbers.

Key generation.

On input k, the key generation algorithm $\mathcal{G}$ works as follows:

1. Generates two cyclic groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of order m for some large prime m and a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$.

2. Generates an arbitrary generator $P \in \mathbb{G}_1$.

3. Chooses randomly $a_1, \ldots, a_{3k} \in \mathbb{Z}_m^*$ and computes $v_1 = \hat{e}(P, P)^{a_1 a_2 a_3}, \cdots, v_k = \hat{e}(P, P)^{a_{3k-2} a_{3k-1} a_{3k}}$.

4. The public parameter is $\mathsf{Pub} = \langle \mathbb{G}_1, \mathbb{G}_2, P, a_1 P, \ldots, a_{3k} P, \hat{e}, v_1, \cdots, v_k \rangle$, and the secret parameter is $\mathsf{Sec} = \langle a_1, \ldots, a_{3k} \rangle$. And then publishes them.

Protocol actions between $\mathcal{P}$ and $\mathcal{V}$.

The generalized scheme is similar to the basic scheme, however, each round is performed in parallel as follows:

1. $\mathcal{P}$ chooses $r_1, r_2, r_3 \in \mathbb{Z}_m^*$ at random, computes $x = \hat{e}(P, P)^{r_1 r_2 r_3}$, $Q_1 = r_1 r_2 r_3 P$, and sends $\langle x, Q \rangle$ to B.

2. $\mathcal{V}$ picks $\omega_1, \ldots, \omega_k \in \mathbb{Z}_m^*$ at random, and sends $R_1 = \omega_1 P, \ldots, R_k = \omega_k P$ to $\mathcal{P}$.

3. On receiving k random values, $\mathcal{P}$ sets

$$S_1 = r_1 r_2 r_3 R_1, S_2 = r_1 r_2 r_3 R_2, \ldots, S_k = r_1 r_2 r_3 R_k,$$

computes Y such that

$$Y = \sum_{i=1}^{k} a_{3i-2} a_{3i-1} a_{3i} P + \sum_{i=1}^{k} (a_{3i-2} + a_{3i-1} + a_{3i}) S_i$$

and sends it to $\mathcal{V}$; $\mathcal{V}$ accepts if both $x = \hat{e}(P, Q)$ and $\hat{e}(Y, P) = \prod_{i=1}^{k} v_i \cdot \hat{e}(a_{3i-2} P + a_{3i-1} P + a_{3i} P, Q)^{\omega_i}$, and rejects otherwise.

Chapter 6

Security analysis

Let $(\mathcal{G}, \mathcal{P}, \mathcal{V}$ be the identification scheme, where a PPT algorithm $\mathcal{G}$ is a key generation algorithm, and $\mathcal{P}$ and $\mathcal{V}$ are two PPT interactive algorithms. One can state the following security result:

Theorem 6.1 *Under B-DHIA, the basic identification scheme is secure against active adversaries whose running time is defined by τ', and the success probability ϵ' is bounded by Π_1^{-1}.*

As mentioned before, the basic way of proving this theorem is just to show that any adversary $\mathcal{I}$ who succeeds in impersonating with non-negligible probability can be reduced into a polynomial-time probabilistic algorithm $\mathcal{A}$ that (τ, t, ϵ)-breaks $\mathsf{C\text{-}DH}$ problem with non-negligible probability. This will be proved in Lemma 6.3.

For a given public parameter Pub and `"help string"` h, let

$$\Pr[(\tilde{\mathcal{P}}(h), \bar{\mathcal{V}}(\mathsf{Pub}) = 1] = \varepsilon(h, \mathsf{Pub}),$$

where the probability is taken over the coin tosses of $\tilde{\mathcal{P}}$ and $\bar{\mathcal{V}}$. Since we assume that the adversary succeeds in breaking the protocol, there must exist polynomial $\Pi_1(k)$ and $\Pi_2(k)$ such that, for sufficiently large k,

$$\Pr\left[\ \varepsilon(h, \mathsf{Pub}) \geq \tfrac{1}{\Pi_2(k)}\ \middle|\ \begin{array}{l} (\mathsf{Sec}, \mathsf{Pub}) \leftarrow \mathcal{G}(1^k); \\ h \leftarrow (\bar{\mathcal{P}}(\mathsf{Sec}, \mathsf{Pub}), \tilde{\mathcal{V}}(\mathsf{Pub})) \end{array}\ \right] \geq \frac{1}{\Pi_1(k)}.$$

Then we can say the adversary $\mathcal{I} = (\tilde{\mathcal{P}}, \tilde{\mathcal{V}})$ can breaks the B-DH problem at least with probability

$$\mathsf{Succ}^{\mathsf{SAA}}(\mathcal{I}) = \frac{1}{\Pi_1}.$$

Lemma 6.2 *Let $\hat{e}$ be the modified Weil pairing as defined in Section 3.3. The sample space is the set of all triples $\mathcal{S} = \{(P, Q) | P, Q \in E(K)\}$, where E is an elliptic curve over K, and the distribution on the sample points is uniform, i.e., $P, Q \in_{\mathcal{U}} \mathcal{S}$. Let a, b, and c be indeterminates and consider the polynomial*

$$e_{a,b,c}(P, Q) = \hat{e}(P, Q)^{abc}.$$

For all $a, b, c \in \mathbb{Z}_m^*$, define random variable

$$X_i(a, b, c) = e_{a_i,b_i,c_i}(P, Q).$$

Then $\langle X_0, \ldots, X_{\ell(m)-1} \rangle$, where $\ell(m)$ is the order of the extension field $\overline{K}$ of K, are uniformly distributed in $\overline{K}$ and pairwise independent.

Proof: For any pair i, j in positive integers, $i \neq j$, and for any pair of points $P, Q \in E(K)$, there is a unique solution $a, b, c \in \mathbb{Z}_m^*$ to the pair of equations:

$$\begin{aligned} e_{a_i,b_i,c_i}(P, Q) &= \alpha, \\ e_{a_j,b_j,c_j}(P, Q) &= \beta. \end{aligned}$$

Thus, $\Pr\left[(X_i(P, Q) = \alpha) \wedge (X_j(P, Q) = \beta)\right] = \Pr[X_i(P, Q) = \alpha] \cdot \Pr[X_j(P, Q) = \beta] = 1/\ell(m)^2$. ■

Lemma 6.3 *Assume that there exists an adversary $\mathcal{I}$ as above. Let $\mathcal{IG}$ be a* B-DH *parameter generator. Then there exists a polynomial-time probabilistic algorithm $\mathcal{A}$ that (t, ϵ)-breaks* C-DH *problem, whose running time τ is defined by*

$$O((N_{\mathcal{V}}(k) T_{\mathcal{V}}(k) + T_{\mathcal{P}}(k)) \Pi_2(k) + T_{\mathsf{off}}(k))$$

and for a valid C-DH *value* C*, the success probability* ϵ *is bounded by*

$$\mathsf{Succ}^{\mathsf{B\text{-}DH}}_{\mathcal{IG}}(\mathcal{A}) \geq \frac{\Pi_1(k)^{-1}}{16}.$$

Proof: First let E denote an elliptic curve over a field K, with $E[m]$ its group of m-torsion points. From the definition of the Weil pairing, we know that if $p = 0$ or p does not divides m then $E[m] \cong (\mathbb{Z}/m\mathbb{Z}) \times (\mathbb{Z}/m\mathbb{Z})$, where p is the characteristic of the field. Let Φ be a natural map in the modified Weil pairing. Note that, for random $P \in E(K)$, revealing $\hat{e}(P, P)$ gives no information on $\Phi(P)$; *i.e.* the distributions of $\hat{e}(P, P)$ and $\Phi(P)$ are independent from Lemma 6.2.

Throughout this thesis, the underlying probability space consists of the random choice of input $x, y, z \in \mathbb{Z}^*_m$ and $P {\in_R} E(K)$ including the coin tosses of the algorithm.

As a proving method, rather than constructing the algorithm $\mathcal{A}$ *in toto*, we will increasingly construct $\mathcal{A}$ in series of "phases". The algorithm runs in five phases. In the first phase, we generate a public parameter $\mathsf{Pub} = \langle P, aP, bP\rangle$ with the corresponding secret parameter $\mathsf{Sec} = \langle a, b\rangle$.

In this phase we simulate the view that the adversary $\mathcal{I}$ would have if it interacted with a proving holding a "real" witness. In the second phase we make the adversary try to convince a honest verifier. In the third phase we use the approximate witness to solve the C-DH problem, $\hat{e}(P, P)^{ab}$. In the fourth phase, we rerun the adversary $\mathcal{I}$ with the public parameter $\mathsf{Pub} = \langle P, aP, bP, cP\rangle$ with additional value cP and its corresponding secret parameter $\mathsf{Sec} = \langle a, b, c\rangle$. In practice, this phase simply executes the above three phases repeatedly. In the last phase, the final algorithm $\mathcal{A}$ is constructed, which solves the C-DH problem, $\hat{e}(P, P)^{abc}$.

Phase 1. This phase takes as input P, aP, and bP, runs in the expected time

$$O(N_{\mathcal{V}}(k)T_{\mathcal{V}}(k)\Pi_2(k) + T_{\mathsf{off}}(k)),$$

and outputs $(\tilde{a}, \gamma_i{}^f, v, h)$, where $v = \hat{e}(P,P)^{\tilde{a}\gamma_i{}^f}$, and h is a "help string". In addition, we know that

i. $\Pr[\varepsilon(h, \mathsf{Pub}) \geq \Pi_2(k)^{-1}] \geq \Pi_1(k)^{-1}$,

ii. The distribution of $\Phi(\tilde{c})$ is uniform and independent of that of (h, Pub).

This stage runs as follows: We choose $\tilde{a}, \gamma_i{}^f \in \mathbb{Z}_m^*$, at random and compute $v = \hat{e}(P,P)^{\tilde{a}\gamma_i{}^f}$ and $\hat{X}_i \equiv \tilde{a}\gamma_i{}^f \pmod{m}$, where $f \not\equiv (m-1) \pmod{m}$. With the help of D-DH oracle, we can easily verify that $(P, \tilde{a}P, \gamma_i{}^f P, abP)$ is a valid DH value. We then simulate the interaction $(\bar{\mathcal{P}}(\cdot, \mathsf{Pub}), \tilde{\mathcal{V}}(\mathsf{Pub}))$.

To simulate the interaction, we employ a zero-knowledge simulation technique [20, 37]. We then modify the identification protocol as the following:

I. $\bar{\mathcal{P}}$ chooses $\omega_0', r_1, r_2 \in \mathbb{Z}_m^*$ at random, computes $x = \hat{e}(P,P)^{\omega_0' r_1 r_2}$, $Q = \omega_0' r_1 r_2 P$, and sends $\langle x, Q\rangle$ to $\tilde{\mathcal{V}}$.

II. $\tilde{\mathcal{V}}$ chooses $\omega \in \mathbb{Z}_m^*$ at random, sets $R = \omega P$, and sends R to $\bar{\mathcal{P}}$.

III. On receiving R, $\bar{\mathcal{P}}$ checks $\hat{e}(R,P) = \hat{e}\big(\frac{\tilde{a}+\gamma_i{}^f-\omega_1}{(\tilde{a}+\gamma_i{}^f)\omega_0}P, P\big)$. If $\omega_0' \neq \omega_0$, we go back to step I. Otherwise, $\bar{\mathcal{P}}$ sets $S = r_1 r_2 P$, computes $Y = \tilde{a}\gamma_i{}^f P + (\tilde{a} + \gamma_i{}^f - \omega_1)S$, and sends it to $\tilde{\mathcal{V}}$.

When the adversary completes the protocol, we outputs the "help string" h that $\tilde{\mathcal{V}}$ outputs, along with $\hat{X}_i$.

In this step, the distribution of C is uniformly distributed in $\mathbb{G}_2$, and its distribution is independent of every variable other than in the adversary's view up to that point, and is also independent of the hidden variable ω'. Therefore, up to this point, this simulation is perfectly correct, and furthermore, the probability that $\omega_0 = \omega_0'$ is $1/|\mathbb{Z}_m^*|$. If $\omega_0 = \omega_0'$, then

$$\begin{aligned} v \cdot \hat{e}(\tilde{a}P + \tilde{b}P, Q)^{\omega} &= v \cdot \hat{e}(\tilde{a}P + \gamma_i{}^f P, \omega_0' r_1 r_2 P)^{\omega} \\ &= \hat{e}(P,P)^{\tilde{a}\gamma_i{}^f} \cdot \hat{e}(P,P)^{(\tilde{a}+\gamma_i{}^f)\omega_0' r_1 r_2 \omega} \\ &= \hat{e}(P,P)^{\tilde{a}\gamma_i{}^f + (\tilde{a}+\gamma_i{}^f)\omega_0' r_1 r_2 \omega}, \end{aligned}$$

and

$$\begin{aligned}\hat{e}(Y,P) &= \hat{e}(\tilde{a}{\gamma_i}^f P + (\tilde{a} + {\gamma_i}^f - \omega_1)r_1r_2P, P) \\ &= \hat{e}(P,P)^{\tilde{a}{\gamma_i}^f + (\tilde{a}+{\gamma_i}^f-\omega_1)r_1r_2}.\end{aligned}$$

Since $\omega_0 = \omega_0'$ and

$$\begin{aligned}\tilde{a}{\gamma_i}^f + (\tilde{a} + {\gamma_i}^f)\omega_0' r_1 r_2 \omega &\equiv \tilde{a}{\gamma_i}^f + (\tilde{a} + {\gamma_i}^f)\omega_0' r_1 r_2 \frac{\tilde{a} + {\gamma_i}^f - \omega_1}{(\tilde{a} + {\gamma_i}^f)\omega_0} \\ &\equiv \tilde{a}{\gamma_i}^f + (\tilde{a} + {\gamma_i}^f - \omega_1)r_1r_2,\end{aligned}$$

we have $\hat{e}(Y,P) = v \cdot \hat{e}(\tilde{a}P + \tilde{b}P, Q)^{\omega}$.

Moreover, C reveals no information of $\Phi(Q_1), \Phi(Q_2)$, and $\Phi(\mathsf{Sec})$, and the distribution of $\Phi(Y)$ is uniform and independent of $\Phi(\mathsf{Sec})$. From the above result, the expected value of the total number of iteration rounds is $(|\mathbb{Z}_m^*| \cdot N_{\mathcal{V}}(k))$. This completes *Phase 1.*

Phase 2. This phase takes as input h, Pub, and output from *Phase 1*, and runs in time $O(T_{\mathcal{P}}(k)\Pi_2(k))$. It outputs Fail or $\mathsf{Success}$ according to success outputs u such that $u \equiv \tilde{a}{\gamma_i}^f \equiv ab \pmod{m}$, since $\hat{e}(P,P)^u = \hat{e}(P,P)^{\tilde{a}{\gamma_i}^f} = \hat{e}(P,P)^{ab}$. The probability of success, given that $\varepsilon(h, \mathsf{Pub}) \geq \Pi_2(k)^{-1}$, is at least $1/2$.

For the sake of convenience, let $\varepsilon = \varepsilon(h, \mathsf{Pub})$, and assume $\varepsilon \geq \Pi_2(k)^{-1}$.

This stage runs as follows: First run $(\tilde{\mathcal{P}}(h), \bar{\mathcal{V}}(\mathsf{Pub}))$ up to $\lceil \Pi_2(k) \rceil$ times, or until $\bar{\mathcal{V}}$ accepts. If $\bar{\mathcal{V}}$ accepts, let

$$\begin{aligned}\hat{e}(Y,P) &= \hat{e}(\tilde{a}\tilde{b}P + (\tilde{a} + {\gamma_i}^f - \omega_1)S \\ &= \hat{e}(\omega P, P)^{\tilde{a}\tilde{b} + (\tilde{a}+{\gamma_i}^f-\omega_1)r_1r_2} \\ &= v \cdot \hat{e}(\tilde{a}P + {\gamma_i}^f P, Q)^{\omega}\end{aligned}$$

be the accepting conversation. Fixing the coin tosses of $\tilde{\mathcal{P}}$, run the interaction again up to $\lceil 4\Pi_2(k) \rceil$, or until $\bar{\mathcal{V}}$ accepts again with a challenge

$\omega'' \not\equiv \omega \pmod{m}$. In this case, let $\hat{X}_j \equiv \tilde{a}{\gamma_j}^f \pmod{m}$. If $\bar{\mathcal{V}}$ accepts this challenge, then we have another accepting conversation

$$\begin{aligned}
\hat{e}(Y', P) &= \hat{e}(\tilde{a}{\gamma_j}^f P + (\tilde{a} + {\gamma_j}^f - \omega'_1)S \\
&= \hat{e}(\omega P, P)^{\tilde{a}{\gamma_j}^f + (\tilde{a} + {\gamma_j}^f - \omega'_1) r_1 r_2} \\
&= v \cdot \hat{e}(\tilde{a}P + {\gamma_j}^f P, Q)^{\omega''}
\end{aligned}$$

where $u \equiv a{\gamma_i}^f \pmod{m}$, $u \equiv a{\gamma_j}^f \pmod{m}$, and $\omega a {\gamma_i}^f \equiv \omega'' a {\gamma_j}^f \pmod{m}$. Therefore, we can easily calculate $f = \log_{\frac{\gamma_j}{\gamma_i}} \omega - \log_{\frac{\gamma_j}{\gamma_i}} \omega''$.

To show that there is another solution with non-negligible probability, we make use of the same method as employed in [14, 28, 37]. Let M be a Boolean matrix of which rows are indexed by the coin tosses ω' of $\tilde{\mathcal{P}}$ and of which columns are indexed by the challenge ω of $\bar{\mathcal{V}}$. Let $M(\omega', \omega) = 1$ if and only if the pair of (ω', ω) makes $\bar{\mathcal{V}}$ be convinced by $\tilde{\mathcal{P}}$.

Just the same as in [14, 28, 37], we call a row ω' in M "heavy" if the fraction of 1's in this row is at least $3\varepsilon/4$. Then the fraction of 1's in M that lies in heavy rows is at least $1/4$. The reason comes from the following equations: let r be the number of rows in M and c be the number of columns in M, and $\bar{r}$ be the number of non-heavy rows, then the total number of 1's in M is $rc\varepsilon$. Then the total number of 1's that lies in non-heavy rows is $\bar{r}c\frac{3\varepsilon}{4} \leq \left(\frac{3}{4}\right) rc\varepsilon$. Therefore, the fraction of 1's in heavy rows is induced by

$$\begin{aligned}
rc\varepsilon - \bar{r}c\frac{3\varepsilon}{4} &\geq rc\varepsilon - rc\frac{3\varepsilon}{4} \\
&= \frac{1}{4}(rc\varepsilon).
\end{aligned}$$

Now consider an accepting conversations by (ω', ω) such that $M(\omega', \omega) = 1$. Since we have another accepting conversation by (ω'', ω) satisfying that $M(\omega'', \omega) = 1$. Then the fraction of ω'' which satisfies

$$M(\omega'', \omega) = 1 \qquad \omega'' \not\equiv \omega \pmod{m}$$

is at least

$$\begin{aligned} \left|\frac{3\varepsilon}{4} - \frac{1}{|\mathbb{Z}_m^*| - 2}\right| &\geq \left|\frac{3(\Pi_2(k)^{-1})}{4} - \frac{1}{\Pi_2(k)}\right| \\ &= \frac{1}{4}\frac{1}{\Pi_2(k)} = \frac{\Pi_2(k)^{-1}}{4}. \end{aligned}$$

To complete the construction of this phase, we use the simple fact that if ε is a small real number, then $(1-\varepsilon) \leq e^{-\varepsilon}$ [41]. Let ε be a success probability. When an experiment is repeated at least t times, the probability that all of experiments fail is at most $(1-\varepsilon)^t \leq e^{-t\varepsilon}$. Thus, if $t \geq 1/\varepsilon$, the probability that at least one experiment succeeds is at least $1 - e^{-1}$. Therefore, for two accepting conversations, the probability that the above procedure succeeds is at least

$$(1 - e^{-1}) \cdot \frac{1}{4} \cdot (1 - e^{-1}) = \frac{\left(1 - e^{-1}\right)^2}{4}.$$

Thus, by a simple calculation, we can obtain the fact that one of fourteen experiments must succeed, thus the probability that one of seven experiments succeeds is at least 1/2.

Phase 3. This phase takes as input, the output $\hat{X}_i$ from *Phase 1*, and the value u from *Phase 2*. Its running time is $O(\Pi_2(k) \cdot \log(\Pi_2)^2)$. When *Phase 2* succeeds, the probability that it solves the C-DH problem is 1/2.

Recall that $\omega \equiv \frac{\tilde{a}+{\gamma_i}^f-\omega_1}{(\tilde{a}+{\gamma_i}^f)\omega_0} \pmod{m}$, if $\omega' = \omega_0$ then

$$\tilde{a}{\gamma_i}^f \equiv \hat{X}_i \pmod{m}, \tag{6.1}$$

$$f \not\equiv (m-1) \pmod{m} \quad \text{and} \quad f = \log_{\frac{\gamma_j}{\gamma_i}} \omega - \log_{\frac{\gamma_j}{\gamma_i}} \omega'', \tag{6.2}$$

$$u \equiv \tilde{a}{\gamma_i}^f \pmod{m} \quad \text{or} \quad u \equiv \tilde{a}{\gamma_j}^f \pmod{m}, \tag{6.3}$$

and

$$u \equiv \tilde{a}\tilde{b} \equiv ab \pmod{m}.$$

Now consider only the case where *Phase 2* succeeds at least with the probability 1/2. First from Eq. (6.1), we have $\hat{e}(aP, \gamma_i P) = \hat{e}(P, P)^{a\gamma_i}$, and from Eqs. (6.2) and (6.3), we have

$$\begin{aligned}
\hat{e}(P, P)^{u} &= \hat{e}(P, P)^{\tilde{a}\gamma_i{}^{f}} \\
&= \hat{e}(\tilde{a}P, \gamma_i{}^{f} P) \\
&= \hat{e}(\tilde{a}P, \tilde{b}P) \\
&= \hat{e}(P, P)^{\tilde{a}\tilde{b}} = \hat{e}(P, P)^{ab}.
\end{aligned}$$

Then with the probability 1/2, we can solve the C-DH problem from the following equations: This completes *Phase 3.*

It follows that, for sufficiently large k, the overall success probability of the algorithm $\mathcal{A}$ is at least

$$\varepsilon(h, \mathsf{Pub}) \times \frac{1}{2} \times \frac{1}{2} = \Pi_1(k)^{-1} \times \frac{1}{2} \times \frac{1}{2} = \frac{\Pi_1(k)^{-1}}{4}.$$

Phase 4. This phase repeatedly executes *Phase 1* to *Phase 3* to solve the C-DH problem, $\hat{e}(P, P)^{xc}$, where $x \equiv ab \pmod{m}$. If phases from 1 to 3 succeed, it is straightforward that this phase must succeed with the above probability.

Phase 5. If *Phase 4* succeeds with given probability, it is equivalent to solving the C-DH problem

$$\hat{e}(P, P)^{xc} = \hat{e}(P, P)^{abc}$$

with probability

$$\mathsf{Succ}_{\mathcal{IG}}^{\mathsf{B\text{-}DH}}(\mathcal{A}) = \frac{\Pi_1(k)^{-1}}{16}.$$

This completes the proof of Lemma 6.3. ■

Therefore, we can conclude that the basic scheme satisfies the requirement of Definition 4.5. This completes the proof of Theorem 6.1. □

Theorem 6.4 *Under B-DHIA, the generalized identification scheme is secure against active adversaries whose running time is defined by τ'', and the success probability is bounded by ϵ''.*

Proof: We do not describe the full description of proof for the generalized identification scheme in detail. However, the proof is straightforward. At first we assume that there exists an (τ'', ϵ'')-breakable adversary $\mathcal{I} = (\tilde{\mathcal{P}}, \tilde{\mathcal{V}})$ who can break this identification scheme. Next, we can reduce the adversary $\mathcal{I}$ at least with the advantage $\mathsf{Adv}^{\mathsf{SAA}}(\mathcal{I}) = \epsilon''$ into the adversary Δ which can solve the underlying problem with probability $\mathsf{Succ}^{\mathsf{B\text{-}DH}}_{\mathcal{IG}}(\Delta) = \bar{\epsilon}$ in running time $\bar{\tau}$. Clearly, during the reduction from $\mathcal{I}$ to Δ, the running time and the success probability of $\mathcal{I}$ is preserved linearly in Δ. Then As in the proof of Theorem 6.1, we can prove Theorem 6.4. ■

Chapter 7

Comparison with other schemes

In this section, we compare our basic scheme with the prior schemes in terms of not only the computation overhead in the light of key size, communication overhead, processing complexity but also their security.

We assume that an elliptic curve E over a base field K is chosen in the same manner as [5]. That is, let E be the elliptic curve defined by the equation $y^2 = x^3+1$ over $\mathbb{F}_p$, where p is a prime satisfying $p \equiv 2 \pmod 3$ and $p = 6q-1$ for some prime $q > 3$. Note that for the sake of the convenience m is replaced by q. As pointed out in [5], from the practical point of view, we can assume that p and q is a 512-bit prime and a 140-bit prime respectively, since the MOV reduction [25] then leads to a DLP in a finite field of size approximately 2^{1024}.

In addition, we assume that system parameters p and q for our basic scheme, Schnorr, and Okamoto are 512-bit and 140-bit respectively, and the modulus n for FFS, GQ scheme is 512-bit. We assume that the standard binary method is employed for the modular exponentiation as well as for the point multiplication in polynomial basis form. We also assume that the parameters for FFS are $l = 20$ and $t = 1$. Here, we only consider Okamoto scheme as an *Identification scheme 1* proposed in [29]. Note that for the purpose of comparison with arithmetic operations of each scheme, we denote M the cost of modular multiplication over a given finite field and A the cost of point addition over a given elliptic curve. Table 7.1 shows the comparison of identification schemes. If the Weierstraß equation over the affine coordinates

Table 7.1: Comparison of identification schemes

	Our scheme	Schnorr	Okamoto	FFS	GQ
Security proof	Yes	Yes	Yes	Yes	Yes
Secure against active attacks	Yes	No	Yes	Yes	No
Underlying problem	B-DH	DLP	DLP	RSA	RSA
ID-based variant	Possible	Possible	Possible	Possible	Possible
Public key size (bits)	512	512	512	10,240	1,024
Private key size (bits)	420	140	280	10,240	512
Communication overhead (bits)	932	672	812	1,044	1,044
Preprocessing (Prover) (# of field multiplications or point additions)	140A	210M	245M	1M	30M
On-line processing (Prover) (# of field multiplications over a given finite field)	2M	Almost 0M	Almost 0M	10M	31M
On-line processing (Verifier) (# of field multiplications over a given finite field)	141M	210M	248M	11M	35M

in fields of characteristic two is given by $y^2 + xy = x^3 + a_2x^2 + a_6$, then our scheme has $a_2 = 0$. Furthermore, since a generator P of the group $\mathbb{G}_1$ is initially known all parties, we can enable the point multiplication in elliptic curves to be more faster. In fact, the point multiplication consists of point doublings and point additions. The binary method requires $(\ell - 1)$ point doublings and $(W - 1)$ point additions, where ℓ is the bit length and W the Hamming weight of the binary expansion, in general, $W = \ell/2$. Therefore, if the point doublings are pre-computed, the point multiplication requires $\frac{\ell}{2}$A-point addition in average and ℓA-point addition in the worst case [4]. The pre-computation is possible because P is initially given. In these cases, we can estimate that A costs less than or equal to two times M, i.e., A $\leq$ 2M.

From Table 7.1, we can state the properties of our scheme as follows: (1) Our scheme is more efficient than Schnorr and Okamoto with respect to preprocessing of prover and on-line processing overhead of both parties (prover and verifier). (2) However, our scheme requires memory for secret key about two times that of Schnorr and Okamoto. Moreover, its communication overhead increases around four times more than those two schemes.

Chapter 8
Conclusion and further work

In this thesis, we have studied the design and analysis of secure identification protocols against active adversaries. We have reviewed previous works and presented current concerns on the identification protocols. And then we have presented our suggestions to solve the problems.

We have presented a practical construction of a new identification scheme based on the B-DH problem using the Weil pairing. The identification protocol is a typical three-round identification (*i.e.*, commitment-challenge-response protocol). To the best of our knowledge, there is no identification scheme based on the B-DH problem published in the open literature. To guarantee that the protocol gives sound security, first we have identified all possible attacks and next we have settles the standard model as our approach. The standard model is the preferred approach of modern, mathematical cryptography. Using this approach, we have showed that any attacker that can break the identification protocol can be transformed into an efficient algorithm to solve the underlying well-studied problem, the B-DH problem. Finally, we have obtained an exact analysis of the security of the protocol rather than asymptotic ones.

As we discussed the relation between an identification scheme and a signature scheme, our proposal can be extended to a signature scheme using the Weil pairing. Also similar to IBE scheme proposed by Boneh *et al.*, our scheme can be associated with the public identity such as e-mail.

As the future work, it remains as an open problem to implement an algorithm to efficiently compute the Weil pairing as suggested in [42]. The other problem to be solved is that the proposed identification scheme has some more computational complexity.

겹선형 디피-헬만 문제에 기반한 안전성 증명 가능 식별 프로토콜

김명선

네트워크에 기반한 사이버 공간에서 현금, 연애편지, 비빌문서등이 디지털 패킷의 형태로 교환되고 있다. 그러나 디지털 통신에 의해 교환되는 메시지는 사람이 구별 할 수 없는 '0'과 '1'의 연속된 나열에 불과하므로, 개인 식별 (entity identification)이라는 본질적 문제를 수반한다. 개인 식별 문제의 대표적 예로서 "누군가 피지 (Fiji)에서 어떤 은행에 $100,000,000의 송금을 요청한다고 하면 그 은행은 그가 진짜 그인지 어떻게 확인 할 수 있는가?"라는 문제를 고려해보자. 사이버 공간같은 비대면 상황에서 한 사용자가 다른 사용자에 의해서 제시된 신원 (identity)이 맞다는 것을 보장해 주는 기법이 필요하며 이 기법을 식별 프로토콜이라 부른다.

본 본문에서는 일방향 함수 (one-way function)로 알려진 겹선형 디피-헬만 (bilinear Diffie-Hellman) 문제에 기반한 식별 프로토콜을 설계하고 이것의 안전성에 대한 정량적 근거를 제시한다. 식별 프로토콜이 가능한 공격에 대해서 안전하다는 것을 보장하기 위해 식별 프로토콜이 얼마나 안전한 가에 대한 정형적이고 엄격한 수학적 증명이 수반되어 한다. 안전성 증명 가능 (provably secure) 식별 프로토콜의 설계는 어려운 일이나 프로토콜의 설계에 기본적인 작업이다. 다른 암호 프로토콜의 설계와 마찬가지로 본 논문에서 식별 프로토콜의 안전성의 보장을 위해, 먼저 가능한 모든 공격에 대한 구별 및 수학적 정의가 이루어진다. 설계하고자 하는 시스템에 대한 공격 모델을 구성한 후 복잡도 이론 (complexity theory)에 근거한 암호학적 축소 (cryptographic reduction) 기법을 사용하여 주어진 시스템의 안전성이 얼마나 되는지 공격자의 자원에 관한 함수로 제시된다. 공격자 함수는 공격에 필요한 시간 및 성공 확률로 표현된다. 기본 식별 프로토콜에 대한 안

전성 증명이 완료되면 일반화 된 식별 프로토콜로 확장 가능하다.

본 논문에서 제시한 식별 프로토콜은 겹선형 디피-헬만 문제에 기반하여 처음으로 제시된 식별 프로토콜이다. 키의 크기, 통신량, 신원기반 (identity-based) 기법으로의 확장가능성의 측면에서 기존의 기법과 비교할 하다. 또한 능동 공격 (active attacks)을 이용하여 제시된 프로토콜을 손상시키는데 필요한 공격자의 자원을 정확하게 제시하고 있다. 본 논문에서 이용하는 디피-헬만 문제의 겹선형성 (bilinearity)의 구체적 예로서 베일 쌍 (Weil pairing)을 사용한다.

References

1. M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway, “Relations among notions of security for public-key encryption schemes”, *Advances in Cryptology – Crypto 1998*, LNCS 1462, Springer-Verlag, pp. 26–45, 1998.

2. M. Bellare and P. Rogaway, “Random oracles are practical: A paradigm for designing efficient protocols”, *ACM Conference on Computer and Communications Security*, pp. 62–73, 1993.

3. M. Bellare and P. Rogaway, “Optimal asymmetric encryption – How to encrypt with RSA”, *Advances in Cryptology – Crypto 1994*, LNCS 950, Springer-Verlag, pp. 92–111, 1994.

4. I. Blake, G. Seroussi and N. Smart, *Elliptic Curves in Cryptography*, Cambridge University Prress, LNS 265, 1999.

5. D. Boneh and M. Franklin, “ID-based encryption from the Weil-pairing”, *Advances in Cryptology – Crypto 2001*, LNCS 2139, Springer-Verlag, pp. 213–229, 2001.

6. D. Boneh, H. Shacham, and B. Lynn, “Short signatures from the Weil-pairing”, *Advances in Cryptology – Asiacrypt 2001*, LNCS 2248, Springer-Verlag, pp. 514–532, 2001.

7. D. Boneh, “The decison Diffie-Hellman problem”, *Proc. of the 3rd Algorithmic Number Theory Symposium*, LNCS 1423, Springer-Verlag, pp. 48–63, 1998.

8. E. F. Brickell and K. S. McCurley, “An interactive identification scheme based on discrete logarithms and factoring”, *Jounal of Cryptology* 5: 29–39, 1992.

9. R. Canetti, “Towards realizing random oracles: Hash functions that hide all partial information”, *Advances in Cryptology – Crypto 1997*, LNCS 1295, Springer-Verlag, pp. 455–469, 1997.

10. R. Canetti, O. Goldreich, and S. Halevi, “The ramdom oracle methodology, revisited”, *ACM Symposium on the Theory of Computing*, ACM Press, pp. 209–218, 1998.

11. R. Canetti, D. Micciancio, and O. Reingold, “Perfectly one-way probabilistic hash functions”, *ACM Symposium on the Theory of Computing*, pp. 131–140, 1998.

12. J.-S. Coron, “On the security of full domain hash”, *Advances in Cryptology – Crypto 2000*, LNCS 1880, Springer-Verlag, pp. 229–235, 2000.

13. R. Cramer and V. Shoup, “A Practical Public Key Cryptosystem Provably Secure agaisnt Adaptive Chosen Ciphertext Attack”, *Advances in Cryptology – Crypto 1998*, LNCS 1462, Springer-Verlag, pp. 13–25, 1998.

14. U. Feige, A. Fiat, and A. Shamir, “Zero-knowledge proofs of identity”, *Journal of Cryptology*, 1: 77–94, 1988.

15. A. Fiat and A. Shamir, “How to prove yourself: pratical solutions to identification and signature problems”, *Advances in Cryptology – Crypto 1986*, LNCS 263, Springer-Verlag, pp. 186-194, 1987.

16. M. Girault, “An identity-based identification scheme based on discrete logarithms modulo a composite number”, *Advances in Cryptology – Eurocrypt 1990*, LNCS 473, Springer-Verlag, pp. 481–486, 1991.

17. O. Goldreich, *Foundation of Cryptography–Fragments of a Book*, available from `http://theory.lcs.mit.edu/~oded/` (1995).

18. O. Goldreich, *Intoduction to Complexity Theory*, available from `http://www.wisdom.weizmann.ac.il/~oded/` (1999).

19. O. Goldreich and H. Krawczyk, “On the composition of zero-knowledge proof systems”, *Proc. of the 17th ICALP*, LNCS 443, Springer-Verlag, pp. 268–282, 1990.

20. S. Goldwasser, S. Micali, and C. Rackoff, “The knowledge complexity of interactive proof systems”, *SIAM J. Comput.*, 18: 186–208, 1989.

21. L. Guillou and J. Quisquater, “A practical zero-knowledge protocol fitted to security microprocessors minimizing both transmission and memory”, *Advances in Cryptology – Eurocrypt 1988*, LNCS 330, Springer-Verlag, pp. 123–128, 1989.

22. A. Joux and K. Nguyen, “Seperating decision Diffie-Hellman from Diffie-Hellman in cryptographic groups”, available from `eprint.iacr.org`.

23. M. Luby, “Pseudorandomness and Cryptographic Applications”, Princeton University Press, 1996.

24. A. J. Menezes, *Elliptic Curve Public Key Cryptosystems*, Kluwer Academic Publishers, 1993.

25. A. J. Menezes, T. Okamoto, and S. A. Vanstone, “Reducing elliptic curve logarithms to logarithms in a finite field”, *IEEE Trans. Inform. Theory*, 39(1993), pp. 1639–1646.

26. A. J. Manezes, P. C.van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC press, 1996.

27. V. Miller, “Short programs for functions on curves”, unpublished manuscript, 1986.

28. K. Ohta and T. Okamoto, “A modification of the Fiat-Shamir scheme”, *Advances in Cryptology – Crypto 1988*, LNCS 403, Springer-Verlag, pp. 232–243, 1990.

29. T. Okamoto, “Provably secure and practical identification schemes and corresponding signature schemes”, *Advances in Cryptology – Crypto 1992*, LNCS 740, Springer-Verlag, pp. 31–53, 1993.

30. T. Okamoto and D. Pointcheval, “The gap-problem: a new class of problems for the security of cryptographic schemes”, *PKC 2001*, LNCS 1992, Springer-Verlag, pp. 104–118, 2001.

31. H. Ong and C.P. Scnorr, “Fast signature generation with a Fiat Shamir-like scheme”, *Advances in Cryptology – Eurocrypt 1990*, LNCS 473, Springer-Verlag, pp. 432–440, 1991.

32. D. Pointcheval, “A new identification scheme based on the perceptrons problem”, *Advances in Cryptology – Eurocrypt 1995*, LNCS 921, Springer-Verlag, pp. 319–328, 1995.

33. C. Popescu, “An identification scheme based on the elliptic curve discrete logarithm problem”, *IEEE High Performance Computing in the Asia-Pacific Region*, Volume: 2, pp. 624–625, 2000.

34. A.D. Santis, S. Micali, and G. Persiano, “Non-interactive zero-knowledge proof systems”, *Advances in Cryptology – Crypto 1987*, LNCS 293, pp. 52–72, 1988.

35. C.P. Schnorr, “Security of 2^t-root identification and signatures”, *Advances in Cryptology – Crypto 1996*, LNCS 1109, Springer-Verlag, pp. 143–156, 1996.

36. V. Shoup, “Why chosen ciphertext security matters”, IBM Research Report RZ3076(#93122), 1998.

37. V. Shoup, “On the security of a practical identification scheme”, *Journal of Cryptology* 12: 247–260, 1999.

38. J. H. Silverman, *The Arithmetic of Elliptic Curves*, Springer-Verlag, GTM 106, 1986.

39. J. Stern, “A new identification scheme based on syndrome decoding”, *Advances in Cryptology – Crypto 1993* LNCS 773, Springer-Verlag, pp. 13–21, 1994.

40. J. Stern, “Designing identification schemes with keys of short size”, *Advances in Cryptology – Crypto 1994* LNCS 839, Springer-Verlag, pp. 164-173, 1994.

41. D.R. Stinson, *Cryptography: Theory and Practice*, CRC Press, 1995.

42. T. Yamanaka, R. Sakai, and M. Kasahara, “Fast computation of pairings over elliptic curves”, *Proc. of SCIS 2002*, pp. 709–714, Jan. 29 – Feb. 1, 2002, Shirahama, Japan.

Acknowledgements

First, I would like to express my heartfelt gratitude to Prof. Kwangjo Kim, my thesis advisor, for his constant direction, support, and inspiration. Without his encouragement and guidance, I could never have carried out my research. Special thanks are also due to Prof. Hyuncheol Park and Dr. Choonsik Park for their generosity and agreeing to serve as advisory committee members.

I would like to mention the close relationship of my colleagues in C&IS lab. members. There were 9 graduates I have enjoyed ICU life together: Byoungcheon Lee, Jaeseung Go, Hyuncheol Park, Heesun Kim, Boyeon Song, Gookwhan An. Manho Lee, Jinho Kim, Jaegwan Park. There are also 14 current members: Jongseong Kim, Wooseok Ham, Hyunrok Lee, Kyusuk Han, Yan Xie, Vo Duc Liem, Hyungki Choi, Byunggon Kim, Songwon Lee, Hwasun Chang, JaeHyuk Park, Soogil Choi, Zhang Fangguo, and Divyan Munirathnam. Their kind and sincere support lead me to carry out this thesis. I also thanks Jeongmi Choi for her helpful support as a staff member.

In addition, I would like to give my thanks Sangbae Park, Jungyun Lee, Sangwon Lee, Jin Kim, Jun Baek Kim, Chul Joon Choi, Sung Jun Min, Joongman Kim, and Yunkyoung Jeong of AC lab., Saehoon Kang, Sung Pyo Cho, Seung Hun Lee, Joon Ho Jung, and Jung Bae Park of CN lab., for their deep interest. I should also give my thanks to Jeongin Kim, Myungjin Lee, Dongjin Kim, and Moonjoo Kim of NA lab., for their constant encouragement and affection. My additional thanks go to Youngseung Kim of DB lab., and Dukyun Nam, Seungik Lee, and Euisuk Hong of CDS lab.

Also I should mention my close and great friends Jaechul Kim, Jinwoo Shin, Jaehyung Cho, and Moonsun Hwang. Their endless concerns and support make my research fruitful. I cannot forget kind and affectionate advice and concerns of brother Donghoon Lee.

I always hope God bless my oldest friend, Giyoung Lee, and his family.

My love and thanks go to my parents for endless and profound affection and their devotion. Especially, my mother devoted her lifetime in supporting me and my sister. I would like to extend my thanks to my sister Myunglim, her husband Daeyoung Lee, and my nephews and niece to my love.

Last, but not least, I greatly appreciate my lovely sweetheart, Eunyoung Kim, for her belief and constant encouragement.

My father had given his whole life to the family. From everlasting to everlasting, I cherish my memories dear to him. I dedicate this thesis to my father.

Curriculum Vitae

Name : Myungsun Kim

Date of Birth : May. 21. 1970

Sex : Male

Nationality : Korean

Education

1990.3–1994.8	Engineering Sogang University (B.S.)
2000.9–2002.8	Engineering Information and Communications University (M.S.)

Career

2002.3–2002.6	Graduate Teaching Assistant ICE525 Computer Security and Electronic Payment System
2000.9–2001.8	Graduate Research Assistant Study on the Information Security and Authentication Technologies in Distributed Environments The Ministry of Information and Communications

2000.9–2001.8 Graduate Research Assistant
Study on Enabling Technologies for Next Generation Public Key Infrastructure
SECUi.COM

2001.3– Graduate Research Assistant
Development of Electronic Voting System for World Cup 2002
Information Research center for Information Security

2001.12–2001.12 Graduate Research Assistant
e-Biz Security and PKI Applications
LG-EDS Systems

2002.6– Graduate Research Assistant
Study on Technologies for Easy Security
Electronics and Telecommunications Research Institute

Publications

International Papers (in English)

1. Myungsun Kim and Kwangjo Kim, “ A New Identification Scheme based on the Bilinear Diffie-Hellman Problem”, To appear in *ACISP 2002*.

2. Myungsun Kim and Kwangjo Kim, “A New Identification Scheme based on the Gap Diffie-Hellman Problem”, *SCIS2002*, vol. 1/2, pp. 349–352, 2002.

Domestic Papers (in Korean)

1. Myungsun Kim, Jongseung Kim, Jungyun Lee, and Kwangjo Kim, “ A Securely Transferrable Ebooks using Public-Key Infrastructure”, *KIISC 종합학술발표회(CISC2001)*, 종합학술발표논문집, pp. 371–374, 2001.