Identity-Based Encryption from the Weil Pairing

Dan Boneh[1*], and Matt Franklin[2**]

[1] Computer Science Department, Stanford University, Stanford CA 94305-9045
dabo@cs.stanford.edu
[2] Computer Science Department, University of California, Davis CA 95616-8562
franklin@cs.ucdavis.edu

Abstract. We propose a fully functional identity-based encryption scheme (IBE). The scheme has chosen ciphertext security in the random oracle model assuming an elliptic curve variant of the computational Diffie-Hellman problem. Our system is based on the Weil pairing. We give precise definitions for secure identity based encryption schemes and give several applications for such systems.

1 Introduction

In 1984 Shamir [27] asked for a public key encryption scheme in which the public key can be an arbitrary string. In such a scheme there are four algorithms: (1) setup generates global system parameters and a master-key, (2) extract uses the master-key to generate the private key corresponding to an arbitrary public key string $\mathsf{ID} \in \{0,1\}^*$, (3) encrypt encrypts messages using the public key ID, and (4) decrypt decrypts messages using the corresponding private key.

Shamir's original motivation for identity-based encryption was to simplify certificate management in e-mail systems. When Alice sends mail to Bob at bob@hotmail.com she simply encrypts her message using the public key string "bob@hotmail.com". There is no need for Alice to obtain Bob's public key certificate. When Bob receives the encrypted mail he contacts a third party, which we call the Private Key Generator (PKG). Bob authenticates himself to the PKG in the same way he would authenticate himself to a CA and obtains his private key from the PKG. Bob can then read his e-mail. Note that unlike the existing secure e-mail infrastructure, Alice can send encrypted mail to Bob even if Bob has not yet setup his public key certificate. Also note that key escrow is inherent in identity-based e-mail systems: the PKG knows Bob's private key. We discuss key revocation, as well as several new applications for IBE schemes in the next section.

Since the problem was posed in 1984 there have been several proposals for IBE schemes (e.g., [7, 29, 28, 21]). However, none of these are fully satisfactory. Some solutions require that users not collude. Other solutions require the PKG to spend a long time for each private key generation request. Some solutions

[*] Supported by DARPA contract F30602-99-1-0530 and the Packard Foundation.
[**] Supported by an NSF Career Award.

require tamper resistant hardware. It is fair to say that constructing a usable IBE system is still an open problem. Interestingly, the related notions of identity-based signature and authentication schemes, also introduced by Shamir [27], do have satisfactory solutions [11, 10].

In this paper we propose a fully functional identity-based encryption scheme. The performance of our system is comparable to the performance of ElGamal encryption in $\mathbb{F}_p^*$. The security of our system is based on a natural analogue of the computational Diffie-Hellman assumption on elliptic curves. Based on this assumption we show that the new system has chosen ciphertext security in the random oracle model. Using standard techniques from threshold cryptography [14, 15] the PKG in our scheme can be distributed so that the $\mathsf{master\text{-}key}$ is never available in a single location. Unlike common threshold systems, we show that robustness for our distributed PKG is free.

Our IBE system can be built from any bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ between two groups $\mathbb{G}_1, \mathbb{G}_2$ as long as a variant of the Computational Diffie-Hellman problem in $\mathbb{G}_1$ is hard. We use the Weil pairing on elliptic curves as an example of such a map. Until recently the Weil pairing has mostly been used for attacking elliptic curve systems [22, 13]. Joux [17] recently showed that the Weil pairing can be used for "good" by using it in a protocol for three party one round Diffie-Hellman key exchange. Using similar ideas, Verheul [30] recently constructed an ElGamal encryption scheme where each public key has two corresponding private keys. In addition to our identity-based encryption scheme, we show how to construct an ElGamal encryption scheme with "built-in" key escrow, i.e., where one global escrow key can decrypt ciphertexts encrypted under any public key.

To argue about the security of our IBE system we define chosen ciphertext security for identity-based encryption. Our model is slightly stronger than the standard model for chosen ciphertext security [25, 1]. While mounting a chosen ciphertext attack on the public key ID, the attacker could ask the PKG for the private key of some public key $\mathsf{ID}' \neq \mathsf{ID}$. This private key might help the attacker. Hence, during the chosen ciphertext attack we allow the attacker to obtain the private key for any public key of her choice other than the one on which the attacker is being challenged. Even with the help of such queries the attacker should have negligible advantage in defeating the semantic security of the system.

The rest of the paper is organized as follows. Several applications of identity-based encryption are discussed in Section 1.1. We then give precise definitions and security models in Section 2. Basic properties of the Weil pairing – sufficient for an understanding of our constructions – are discussed in Section 3. Our main identity-based encryption scheme is presented in Section 4. Some extensions and variations (efficiency improvements, distribution of the master-key) are considered in Section 5. Our construction for ElGamal encryption with a global escrow key is described in Section 6. Conclusions and open problems are discussed in Section 7.

1.1 Applications for Identity-Based Encryption

The original motivation for identity-based encryption is to help the deployment of a public key infrastructure. In this section, we show several other unrelated applications.

Revocation of Public Keys Public key certificates contain a preset expiration date. In an IBE system key expiration can be done by having Alice encrypt e-mail sent to Bob using the public key: "`bob@hotmail.com` $\|$ `current-year`". In doing so Bob can use his private key during the current year only. Once a year Bob needs to obtain a new private key from the PKG. Hence, we get the effect of annual private key expiration. Note that unlike the existing PKI, Alice does not need to obtain a new certificate from Bob every time Bob refreshes his certificate.

One could potentially make this approach more granular by encrypting e-mail for Bob using "`bob@hotmail.com` $\|$ `current-date`". This forces Bob to obtain a new private key every day. This might be feasible in a corporate PKI where the PKG is maintained by the corporation. With this approach key revocation is quite simple: when Bob leaves the company and his key needs to be revoked, the corporate PKG is instructed to stop issuing private keys for Bob's e-mail address. The interesting property is that Alice does not need to communicate with any third party to obtain Bob's daily public key. This approach enables Alice to send messages into the future: Bob will only be able to decrypt the e-mail on the date specified by Alice (see [26, 8] for methods of sending messages into the future using a stronger security model).

Delegation of Decryption Keys Another application for IBE systems is delegation of decryption capabilities. We give two example applications. In both applications the user Bob plays the role of the PKG. Bob runs the setup algorithm to generate his own IBE system parameters params and his own master-key. Here we view params as Bob's public key. Bob obtains a certificate from a CA for his public key params. When Alice wishes to send mail to Bob she first obtains Bob's public key params and public key certificate.

1. **Delegation to a laptop.** Suppose Alice encrypts mail to Bob using the current date as the IBE encryption key (she uses Bob's params as the IBE system parameters). Since Bob has the master-key he can extract the private key corresponding to this IBE encryption key and then decrypt the message. Now, suppose Bob goes on a trip for seven days. Normally, Bob would put his private key on his laptop. If the laptop is stolen the private key is compromised. When using the IBE system Bob could simply install on his laptop the seven private keys corresponding to the seven days of the trip. If the laptop is stolen, only the private key for those seven days are compromised. The master-key is unharmed. This is analogous to the delegation scenario for *signature schemes* considered by Goldreich et al. [16].

2. Delegation of duties. Suppose Alice encrypts mail to Bob using the subject line as the IBE encryption key. Bob can decrypt mail using his master-key. Now, suppose Bob has several assistants each responsible for a different task (e.g. one is 'purchasing', another is 'human-resources', etc.). Bob gives one private key to each of his assistants corresponding to the assistant's responsibility. Each assistant can then decrypt messages whose subject line falls within its responsibilities, but it cannot decrypt messages intended for other assistants. Note that Alice only obtains a single public key from Bob (params), and she uses that public key to send mail with any subject line of her choice. The mail can only be read by the assistant responsible for that subject.

More generally, IBE can simplify various systems that manage a large number of public keys. Rather than storing a big database of public keys the system can either derive these public keys from usernames, or simply use the integers $1, \ldots, n$ as distinct public keys.

2 Definitions

Bilinear Map. Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two cyclic groups of order q for some large prime q. In our system, $\mathbb{G}_1$ is the group of points of an elliptic curve over $\mathbb{F}_p$ and $\mathbb{G}_2$ is a subgroup of $\mathbb{F}_{p^2}^*$. Therefore, we view $\mathbb{G}_1$ as an additive group and $\mathbb{G}_2$ as a multiplicative group. A map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ is said to be *bilinear* if $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$ for all $P, Q \in \mathbb{G}_1$ and all $a, b \in \mathbb{Z}$. As we will see in Section 3, the Weil pairing is an example of an efficiently computable non-degenerate bilinear map.

Weil Diffie-Hellman Assumption (WDH). Our IBE system can be built from any bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ for which the following assumption holds: there is no efficient algorithm to compute $\hat{e}(P, P)^{abc} \in \mathbb{G}_2$ from $P, aP, bP, cP \in \mathbb{G}_1$ where $a, b, c \in \mathbb{Z}$. This assumption is precisely defined in Section 3. We note that this WDH assumption implies that the Diffie-Hellman problem is hard in the group $\mathbb{G}_1$.

Identity-Based Encryption. An identity-based encryption scheme is specified by four randomized algorithms: Setup, Extract, Encrypt, Decrypt:

Setup: takes a security parameter k and returns params (system parameters) and master-key. The system parameters include a description of a finite message space $\mathcal{M}$, and a description of a finite ciphertext space $\mathcal{C}$. Intuitively, the system parameters will be publicly known, while the master-key will be known only to the "Private Key Generator" (PKG).

Extract: takes as input params, master-key, and an arbitrary $\mathsf{ID} \in \{0, 1\}^*$, and returns a private key d. Here ID is an arbitrary string that will be used as a public key, and d is the corresponding private decryption key. The Extract algorithm extracts a private key from the given public key.

Encrypt: takes as input params, ID, and $M \in \mathcal{M}$. It returns a ciphertext $C \in \mathcal{C}$.

Decrypt: takes as input params, ID, $C \in \mathcal{C}$, and a private key d. It return $M \in \mathcal{M}$.

These algorithms must satisfy the standard consistency constraint, namely when d is the private key generated by algorithm $\mathsf{Extract}$ when it is given ID as the public key, then

$$\forall M \in \mathcal{M} : \mathsf{Decrypt}(\mathsf{params}, \mathsf{ID}, C, d) = M \quad \text{where} \quad C = \mathsf{Encrypt}(\mathsf{params}, \mathsf{ID}, M)$$

Chosen ciphertext security. Chosen ciphertext security ($\mathsf{IND\text{-}CCA}$) is the standard acceptable notion of security for a public key encryption scheme [25, 1, 9]. Hence, it is natural to require that an identity-based encryption scheme also satisfy this strong notion of security. However, the definition of chosen ciphertext security must be strengthened a bit. The reason is that when an attacker attacks a public key ID in an identity-based system, the attacker might already possess the private keys of users $\mathsf{ID}_1, \ldots, \mathsf{ID}_n$ of her choice. The system should remain secure under such an attack. Hence, the definition of chosen ciphertext security must allow the attacker to obtain the private key associated with any identity ID_i of her choice (other than the public key ID being attacked). We refer to such queries as private key extraction queries. Another difference is that the attacker is challenged on a public key ID of her choice (as opposed to a random public key).

We say that an identity-based encryption scheme is semantically secure against an adaptive chosen ciphertext attack ($\mathsf{IND\text{-}ID\text{-}CCA}$) if no polynomially bounded adversary $\mathcal{A}$ has a non-negligible advantage against the Challenger in the following game:

Setup: The challenger takes a security parameter k and runs the Setup algorithm. It gives the adversary the resulting system parameters params. It keeps the $\mathsf{master\text{-}key}$ to itself.

Phase 1: The adversary issues queries $q_1, \ldots, q_m$ where query q_i is one of:

- Extraction query $\langle \mathsf{ID}_i \rangle$. The challenger responds by running algorithm $\mathsf{Extract}$ to generate the private key d_i corresponding to the public key $\langle \mathsf{ID}_i \rangle$. It sends d_i to the adversary.
- Decryption query $\langle \mathsf{ID}_i, C_i \rangle$. The challenger responds by running algorithm $\mathsf{Extract}$ to generate the private key d_i corresponding to ID_i. It then runs algorithm $\mathsf{Decrypt}$ to decrypt the ciphertext C_i using the private key d_i. It sends the resulting plaintext to the adversary.

These queries may be asked adaptively, that is, each query q_i may depend on the replies to $q_1, \ldots, q_{i-1}$.

Challenge: Once the adversary decides that Phase 1 is over it outputs two plaintexts $M_0, M_1 \in \mathcal{M}$ and an identity ID on which it wishes to be challenged. The only constraint is that ID did not appear in any private key extraction query in Phase 1.

The challenger picks a random bit $b \in \{0, 1\}$ and sets $C = \mathsf{Encrypt}(\mathsf{params}, \mathsf{ID}, M_b)$. It sends C as the challenge to the adversary.

Phase 2: The adversary issues more queries $q_{m+1}, \ldots, q_n$ where query q_i is one of:

– Extraction query $\langle \mathsf{ID}_i \rangle$ where $\mathsf{ID}_i \neq \mathsf{ID}$. Challenger responds as in Phase 1.

– Decryption query $\langle \mathsf{ID}_i, C_i \rangle \neq \langle \mathsf{ID}, C \rangle$. Challenger responds as in Phase 1.

These queries may be asked adaptively as in Phase 1.

Guess: Finally, the adversary outputs a guess $b' \in \{0, 1\}$. The adversary wins the game if $b = b'$.

We refer to such an adversary $\mathcal{A}$ as an IND-ID-CCA attacker. We define adversary $\mathcal{A}$'s advantage in attacking the scheme as: $\mathrm{Adv}(\mathcal{A}) = \left|\Pr[b = b'] - \frac{1}{2}\right|$.
The probability is over the random bits used by the challenger and the adversary. We say that the IBE system is semantically secure against an adaptive chosen ciphertext attack (IND-ID-CCA) if no polynomially bounded adversary has a non-negligible advantage in attacking the scheme. As usual, "non-negligible" should be understood as larger than $1/f(k)$ for some polynomial f (recall k is the security parameter). Note that the standard definition of chosen ciphertext security (IND-CCA) [25, 1] is the same as above except that there are no private key extraction queries and the attacker is challenged on a random public key (rather than a public key of her choice).

Private key extraction queries are related to the definition of chosen ciphertext security in the multiuser settings [4]. After all, our definition involves multiple public keys belonging to multiple users. In [4] the authors show that that multiuser IND-CCA is reducible to single user IND-CCA using a standard hybrid argument. This does not hold in the identity-based settings, IND-ID-CCA, since the attacker gets to choose which public keys to corrupt during the attack. To emphasize the importance of private key extraction queries we note that our IBE system can be easily modified (by removing one of the hash functions) into a system which has chosen ciphertext security when private extraction queries are disallowed. However, the scheme is completely insecure when extraction queries are allowed.

One way identity-based encryption. The proof of security for our IBE system makes use of a weak notion of security called one-way encryption (OWE) [12]. OWE is defined for standard public key encryption schemes (not identity based) as follows: the attacker $\mathcal{A}$ is given a random public key K_{pub} and a ciphertext C which is the encryption of a random message M using K_{pub}. The attacker's goal is to recover the corresponding plaintext. It has advantage ϵ in attacking the system if $\Pr[\mathcal{A}(K_{pub}, C) = M] = \epsilon$. We say that the public key scheme is a one-way encryption scheme (OWE) if no polynomial time attacker has non-negligible advantage in attacking the scheme. See [12] for precise definitions.

For identity-based encryption, we strengthen the definition as follows. We say that an IBE scheme is a one-way identity-based encryption scheme (ID-OWE) if no polynomially bounded adversary $\mathcal{A}$ has a non-negligible advantage against the Challenger in the following game:

Setup: The challenger takes a security parameter k and runs the Setup algorithm. It gives the adversary the resulting system parameters params. It keeps the $\mathsf{master\text{-}key}$ to itself.

Phase 1: The adversary issues private key extraction queries $\mathsf{ID}_1, \ldots, \mathsf{ID}_m$. The challenger responds by running algorithm $\mathsf{Extract}$ to generate the private key d_i corresponding to the public key ID_i. It sends d_i to the adversary. These queries may be asked adaptively.

Challenge: Once the adversary decides that Phase 1 is over it outputs a public key $\mathsf{ID} \neq \mathsf{ID}_1, \ldots, \mathsf{ID}_m$ on which it wishes to be challenged. The challenger picks a random $M \in \mathcal{M}$ and encrypts M using ID as the public key. It then sends the resulting ciphertext C to the adversary.

Phase 2: The adversary issues more extraction queries $\mathsf{ID}_{m+1}, \ldots, \mathsf{ID}_n$. The only constraint is that $\mathsf{ID}_i \neq \mathsf{ID}$. The challenger responds as in Phase 1.

Guess: Finally, the adversary outputs a guess $M' \in \mathcal{M}$. The adversary wins the game if $M = M'$.

We refer to such an attacker $\mathcal{A}$ as an $\mathsf{ID\text{-}OWE}$ attacker. We define adversary's $\mathcal{A}$'s advantage in attacking the scheme as: $\mathrm{Adv}(\mathcal{A}) = \Pr[M = M']$. The probability is over the random bits used by the challenger and the adversary. Note that the definitions of OWE is the same as $\mathsf{ID\text{-}OWE}$ except that there are no private key extraction queries and the attacker is challenged on a random public key (rather than a public key of her choice).

3 Properties of the Weil Pairing

The bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ discussed in Section 2 is implemented via the Weil pairing. In this section we describe the basic properties of this pairing and the complexity assumption needed for the security of our system. To make the presentation concrete we consider a specific supersingular elliptic curve. In Section 5 we describe several extensions and observations for our approach. The complete definition and algorithm for computing the pairing are given in the full version of the paper [2].

Let p be a prime satisfying $p = 2 \bmod 3$ and $p = 6q - 1$ for some prime q. Let E be the elliptic curve defined by the equation $y^2 = x^3 + 1$ over $\mathbb{F}_p$. We state a few elementary facts about this curve:

Fact 1: Since $x^3 + 1$ is a permutation on $\mathbb{F}_p$ it easily follows that $E/\mathbb{F}_p$ contains $p+1$ points. We let O denote the point at infinity. Let $P \in E/\mathbb{F}_p$ be a generator of the group of points of order $q = (p+1)/6$. We denote this group by G_q.

Fact 2: For any $y_0 \in \mathbb{F}_p$ there is a unique point (x_0, y_0) on $E/\mathbb{F}_p$. Hence, if (x, y) is a random non-zero point on $E/\mathbb{F}_p$ then y is uniform in $\mathbb{F}_p$. We use this property to simplify the proof of security.

Fact 3: Let $1 \neq \zeta \in \mathbb{F}_{p^2}$ be a solution of $x^3 - 1 = 0 \bmod p$. Then the map $\phi(x, y) = (\zeta x, y)$ is an automorphism of the group of points on the curve E. Note that when $P = (x, y) \in E/\mathbb{F}_p$ we have that $\phi(P) \in E/\mathbb{F}_{p^2}$, but $\phi(P) \notin E/\mathbb{F}_p$. Hence, $P \in E/\mathbb{F}_p$ is linearly independent of $\phi(P) \in E/\mathbb{F}_{p^2}$.

Fact 4: Since the points P and $\phi(P)$ are linearly independent they generate a group isomorphic to $\mathbb{Z}_q \times \mathbb{Z}_q$. We denote this group of points by $E[q]$.

Let μ_q be the subgroup of $\mathbb{F}_{p^2}^*$ containing all elements of order $q = (p+1)/6$. The Weil pairing on the curve $E/\mathbb{F}_{p^2}$ is a mapping $e : E[q] \times E[q] \to \mu_q$. We define the modified Weil pairing $\hat{e} : G_q \times G_q \to \mu_q$ to be:

$$\hat{e}(P, Q) = e(P, \phi(Q))$$

The modified Weil pairing satisfies the following properties:

1. Bilinear: For all $P, Q \in G_q$ and for all $a, b \in \mathbb{Z}$ we have $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$.
2. Non-degenerate: $\hat{e}(P, P) \in \mathbb{F}_{p^2}$ is an element of order q, and in fact a generator of μ_q.
3. Computable: Given $P, Q \in G_q$ there is an efficient algorithm, due to Miller, to compute $\hat{e}(P, Q)$. This algorithm is described in [2]. Its run time is comparable to a full exponentiation in $\mathbb{F}_p$.

3.1 Weil Diffie-Hellman Assumption

Joux and Nguyen [18] point out that although the Computational Diffie-Hellman problem (CDH) appears to be hard in the group G_q, the Decisional Diffie-Hellman problem (DDH) is easy in G_q. Observe that given $P, aP, bP, cP \in G_q$ we have

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

Hence, the modified Weil pairing provides an easy test for Diffie-Hellman tuples. Consequently, one cannot use the DDH assumption to build cryptosystems in the group G_q. The security of our system is based on the following natural variant of the Computational Diffie-Hellman assumption.

Weil Diffie-Hellman Assumption (WDH): Let $p = 2 \bmod 3$ be a k-bit prime and $p = 6q - 1$ for some prime q. Let $E/\mathbb{F}_p$ be the curve $y^2 = x^3 + 1$ and let $P \in E/\mathbb{F}_p$ be a point of order q. The WDH problem is as follows: Given $\langle P, aP, bP, cP \rangle$ for random $a, b, c \in \mathbb{Z}_q^*$ compute $W = \hat{e}(P, P)^{abc} \in \mathbb{F}_{p^2}$. The WDH Assumption states that when p is a random k-bit prime there is no probabilistic polynomial time algorithm for the WDH problem. An algorithm $\mathcal{A}$ has advantage ϵ in solving WDH if $\quad \Pr\left[\mathcal{A}(P, aP, bP, cP) = \hat{e}(P, P)^{abc}\right] \geq \epsilon$. Joux [17] previously used an analogue of the WDH assumption to construct a one-round three party Diffie-Hellman protocol. Verheul [30] recently used a related hardness assumption.

To conclude this section we point out that the discrete log problem in G_q is easily reducible to the discrete log problem in $\mathbb{F}_{p^2}^*$ (see [22, 13]). To see this observe that given $P \in G_q$ and $Q = aP$ we can define $g = \hat{e}(P, P)$ and $h = \hat{e}(Q, P)$. Then $h = g^a$ and $h, g \in \mathbb{F}_{p^2}^*$. Hence, computing discrete log in $\mathbb{F}_{p^2}^*$ is sufficient for computing discrete log in G_q. For proper security of discrete log in $\mathbb{F}_p^*$ one often uses primes p that are 1024-bits long. Since we need discrete log in G_q to be difficult our system also uses primes p that are at least 1024-bits long.

4 Our Identity-Based Encryption Scheme

We describe our scheme in stages. First we give a basic identity-based encryption scheme which is not secure against an adaptive chosen ciphertext attack. The only reason for describing the basic scheme is to make the presentation easier to follow. Our full scheme, described in Section 4.3, extends the basic scheme to get security against an adaptive chosen ciphertext attack ($\mathsf{IND\text{-}ID\text{-}CCA}$) in the random oracle model.

4.1 MapToPoint

Let p be a prime satisfying $p = 2 \bmod 3$ and $p = 6q-1$ for some prime $q > 3$. Let E be the elliptic curve $y^2 = x^3+1$ over $\mathbb{F}_p$. Our IBE scheme makes use of a simple algorithm for converting an arbitrary string $\mathsf{ID} \in \{0,1\}^*$ to a point $Q_{\mathsf{ID}} \in E/\mathbb{F}_p$ of order q. We refer to this algorithm as $\mathsf{MapToPoint}$. We describe one of several ways of doing so. Let G be a cryptographic hash function $G : \{0,1\}^* \rightarrow \mathbb{F}_p$ (in the security analysis we view G as a random oracle). Algorithm $\mathsf{MapToPoint}_G$ works as follows:
1. Compute $y_0 = G(\mathsf{ID})$ and $x_0 = (y_0^2-1)^{1/3} = (y_0^2-1)^{(2p-1)/3} \bmod p$.
2. Let $Q = (x_0, y_0) \in E/\mathbb{F}_p$. Set $Q_{\mathsf{ID}} = 6Q$. Then Q_{ID} has order q as required.
This completes the description of $\mathsf{MapToPoint}$. We note that there are 5 values of $y_0 \in \mathbb{F}_p$ for which $6Q = (x_0, y_0) = O$ (these are the non-O points of order dividing 6). When $G(\mathsf{ID})$ is one of these 5 values Q_{ID} will not have order q. Since it is extremely unlikely for $G(\mathsf{ID})$ to hit one of these five points, for simplicity we say that such ID's are invalid. It is easy to extend algorithm $\mathsf{MapToPoint}$ to handle these five y_0 values as well.

4.2 BasicIdent

To explain the basic ideas underlying our IBE system we describe the following simple scheme, called $\mathsf{BasicIdent}$. We present the scheme by describing the four algorithms: Setup, $\mathsf{Extract}$, $\mathsf{Encrypt}$, $\mathsf{Decrypt}$. We let k be the security parameter given to the setup algorithm.

Setup: The algorithm works as follows:

Step 1: Choose a large k-bit prime p such that $p = 2 \bmod 3$ and $p = 6q-1$ for some prime $q > 3$. Let E be the elliptic curve defined by $y^2 = x^3+1$ over $\mathbb{F}_p$. Choose an arbitrary $P \in E/\mathbb{F}_p$ of order q.

Step 2: Pick a random $s \in \mathbb{Z}_q^*$ and set $P_{pub} = sP$.

Step 3: Choose a cryptographic hash function $H : \mathbb{F}_{p^2} \rightarrow \{0,1\}^n$ for some n. Choose a cryptographic hash function $G : \{0,1\}^* \rightarrow \mathbb{F}_p$. The security analysis will view H and G as random oracles.

The message space is $\mathcal{M} = \{0,1\}^n$. The ciphertext space is $\mathcal{C} = E/\mathbb{F}_p \times \{0,1\}^n$. The system parameters are $\mathsf{params} = \langle p, n, P, P_{pub}, G, H \rangle$. The $\mathsf{master\text{-}key}$ is $s \in \mathbb{Z}_q$.

Extract: For a given string $\mathsf{ID} \in \{0,1\}^*$ the algorithm builds a private key d as follows:

Step 1: Use $\mathsf{MapToPoint}_G$ to map ID to a point $Q_{\mathsf{ID}} \in E/\mathbb{F}_p$ of order q.

Step 2: Set the private key d_{ID} to be $d_{\mathsf{ID}} = sQ_{\mathsf{ID}}$ where s is the master key.

Encrypt: To encrypt $M \in \mathcal{M}$ under the public key ID do the following: (1) use $\mathsf{MapToPoint}_G$ to map ID into a point $Q_{\mathsf{ID}} \in E/\mathbb{F}_p$ of order q, (2) choose a random $r \in \mathbb{Z}_q$, and (3) set the ciphertext to be

$$C = \langle rP, \;\; M \oplus H(g_{\mathsf{ID}}^r) \rangle \quad \text{where} \quad g_{\mathsf{ID}} = \hat{e}(Q_{\mathsf{ID}}, P_{pub}) \in \mathbb{F}_{p^2}$$

Decrypt: Let $C = \langle U, V \rangle \in \mathcal{C}$ be a ciphertext encrypted using the public key ID. If $U \in E/\mathbb{F}_p$ is not a point of order q reject the ciphertext. Otherwise, to decrypt C using the private key d_{ID} compute:

$$V \oplus H(\hat{e}(d_{\mathsf{ID}}, U)) = M$$

This completes the description of $\mathsf{BasicIdent}$. We first verify consistency. When everything is computed as above we have:
1. During encryption M is Xored with the hash of: g_{ID}^r.
2. During decryption V is Xored with the hash of: $\hat{e}(d_{\mathsf{ID}}, U)$.
These masks used during encryption and decryption are the same since:

$$\hat{e}(d_{\mathsf{ID}}, U) = \hat{e}(sQ_{\mathsf{ID}}, rP) = \hat{e}(Q_{\mathsf{ID}}, P)^{sr} = \hat{e}(Q_{\mathsf{ID}}, P_{pub})^r = g_{\mathsf{ID}}^r$$

Thus, applying decryption after encryption produces the original message M as required. We note that there is no need to devise attacks against this basic scheme since it is only presented for simplifying the exposition. The next section describes the full scheme.

Performance. Algorithms Setup and Extract are very simple algorithms. At the heart of both algorithms is a standard multiplication on the curve $E/\mathbb{F}_p$. Algorithm Encrypt requires that the encryptor compute the Weil pairing of Q_{ID} and P_{pub}. Note that this computation is independent of the message, and hence can be done once and for all. Once g_{ID} is computed the performance of the system is almost identical to standard ElGamal encryption. We also note that the ciphertext length is the same as in regular ElGamal encryption in $\mathbb{F}_p$. Decryption is a simple Weil pairing computation.

Security. Next, we study the security of this basic scheme. The following theorem shows that the scheme is a one-way identity based encryption scheme (ID-OWE) assuming WDH is hard.

Theorem 1. *Let the hash functions H, G be random oracles. Suppose there is an* ID-OWE *attacker $\mathcal{A}$ that has advantage ϵ against the scheme* BasicIdent. *Suppose $\mathcal{A}$ make at most $q_E > 0$ private key extraction queries and $q_H > 0$ hash queries. Then there is an algorithm $\mathcal{B}$ for computing WDH with advantage at least $\frac{\epsilon}{e(1+q_E)\cdot q_H} - \frac{1}{q_H \cdot 2^n}$. Here $e \approx 2.71$ is the base of the natural logarithm. The running time of $\mathcal{B}$ is $O(time(\mathcal{A}))$.*

To prove the theorem we need to define a related Public Key Encryption scheme (not an identity scheme), called PubKeyEnc. PubKeyEnc is described by three algorithms: keygen, encrypt, decrypt.

keygen: The algorithm works as follows:

Step 1: Choose a large k-bit prime p such that $p = 2 \bmod 3$ and $p = 6q - 1$ for some prime $q > 3$. Let E be the elliptic curve defined by $y^2 = x^3 + 1$ over $\mathbb{F}_p$. Choose an arbitrary $P \in E/\mathbb{F}_p$ of order q.

Step 2: Pick a random $s \in \mathbb{Z}_q^*$ and set $P_{pub} = sP$.
Pick a random point $Q_{\mathsf{ID}} \in E/\mathbb{F}_p$ of order q. Then Q_{ID} is in the group generated by P.

Step 3: Choose a cryptographic hash function $H : \mathbb{F}_{p^2} \to \{0,1\}^n$ for some n.

Step 4: The public key is $\langle p, n, P, P_{pub}, Q_{\mathsf{ID}}, H\rangle$. The private key is $d_{\mathsf{ID}} = sQ_{\mathsf{ID}}$.

encrypt: To encrypt $M \in \{0,1\}^n$ choose a random $r \in \mathbb{Z}_q$ and set the ciphertext to be:

$$C = \langle rP, \quad M \oplus H(g^r)\rangle \quad \text{where} \quad g = \hat{e}(Q_{\mathsf{ID}}, P_{pub}) \in \mathbb{F}_{p^2}$$

decrypt: Let $C = \langle U, V\rangle \in \mathcal{C}$ be a ciphertext encrypted using the public key $\langle p, n, P, P_{pub}, Q_{\mathsf{ID}}, H\rangle$. To decrypt C using the private key d_{ID} compute:

$$V \oplus H(\hat{e}(d_{\mathsf{ID}}, U)) = M$$

This completes the description of PubKeyEnc. We now prove Theorem 1 in two steps. We first show that an ID-OWE attack on BasicIdent can be converted to a OWE attack on PubKeyEnc. This step shows that private key extraction queries do not help the attacker. We then show that PubKeyEnc is OWE if the WDH assumption holds. The proofs of these two lemmas appear in the full version of the paper [2].

Lemma 1. *Let G be a random oracle from $\{0,1\}^*$ to $\mathbb{F}_p$. Let $\mathcal{A}$ be an* ID-OWE *attacker that has advantage ϵ against* BasicIdent*. Suppose $\mathcal{A}$ makes at most $q_E > 0$ private key extraction queries. Then there is a* OWE *attacker $\mathcal{B}$ that has advantage $\epsilon/e(1 + q_E)$ against* PubKeyEnc*. Its running time is $O(time(\mathcal{A}))$.*

Lemma 2. *Let H be a random oracle from $\mathbb{F}_{p^2}$ to $\{0,1\}^n$. Let $\mathcal{A}$ be a* OWE *attacker that has advantage ϵ against* PubKeyEnc*. Suppose $\mathcal{A}$ makes a total of $q_H > 0$ queries to H. Then there is an algorithm $\mathcal{B}$ that solves the WDH problem with advantage at least $(\epsilon - \frac{1}{2^n})/q_H$ and a running time $O(time(\mathcal{A}))$.*

Proof of Theorem 1. The theorem follows directly from Lemma 1 and Lemma 2. Composing both reductions shows that an ID-OWE attacker on BasicIdent with advantage ϵ gives an algorithm for WDH with advantage $(\epsilon/e(1+q_E) - 1/2^n)/q_H$, as required. □

4.3 Identity-Based Encryption with Chosen Ciphertext Security

We use a technique due to Fujisaki-Okamoto [12] to convert the BasicIdent scheme of the previous section into a chosen ciphertext secure IBE system (in the sense of Section 2) in the random oracle model. Let $\mathcal{E}$ be a public key encryption scheme. We denote by $\mathcal{E}_{pk}(M;r)$ the encryption of M using the random bits r under the public key pk. Fujisaki-Okamoto define the hybrid scheme $\mathcal{E}^{hy}$ as:

$$\mathcal{E}^{hy}_{pk}(M) \;=\; \mathcal{E}_{pk}(\sigma; H_1(\sigma, M)) \,\Big\|\, G_1(\sigma) \oplus M$$

Here σ is generated at random and H_1, G_1 are cryptographic hash functions. Fujisaki-Okamoto show that if $\mathcal{E}$ is a one-way encryption scheme then $\mathcal{E}^{hy}$ is a chosen ciphertext secure system (IND-CCA) in the random oracle model (assuming $\mathcal{E}_{pk}$ satisfies some natural constraints).

We apply this transformation to BasicIdent and show that the resulting IBE system is IND-ID-CCA. We obtain the following IBE scheme which we call FullIdent. Recall that n is the length of the message to be encrypted.

Setup: As in the BasicIdent scheme. In addition, we pick a hash function $H_1 : \{0,1\}^n \times \{0,1\}^n \to \mathbb{F}_q$, and a hash function $G_1 : \{0,1\}^n \to \{0,1\}^n$.

Extract: As in the BasicIdent scheme.

Encrypt: To encrypt $M \in \{0,1\}^n$ under the public key ID do the following: (1) use algorithm $\mathsf{MapToPoint}_G$ to convert ID into a point $Q_{\mathsf{ID}} \in E/\mathbb{F}_p$ of order q, (2) choose a random $\sigma \in \{0,1\}^n$, (3) set $r = H_1(\sigma, M)$, and (4) set the ciphertext to be

$$C = \langle rP,\;\; \sigma \oplus H(g^r_{\mathsf{ID}}),\;\; M \oplus G_1(\sigma) \rangle \quad \text{where} \quad g_{\mathsf{ID}} = \hat{e}(Q_{\mathsf{ID}}, P_{pub}) \in \mathbb{F}_{p^2}$$

Decrypt: Let $C = \langle U, V, W \rangle \in \mathcal{C}$ be a ciphertext encrypted using the public key ID. If $U \in E/\mathbb{F}_p$ is not a point of order q reject the ciphertext. To decrypt C using the private key d_{ID} do:

1. Compute $\quad V \oplus H(\hat{e}(d_{\mathsf{ID}}, U)) = \sigma$.
2. Compute $\quad W \oplus G_1(\sigma) = M$.
3. Set $r = H_1(\sigma, M)$. Test that $U = rP$. If not, reject the ciphertext.
4. Output M as the decryption of C.

This completes the description of FullIdent. Note that M is encrypted as $W = M \oplus G_1(\sigma)$. This can be replaced by $W = E_{G_1(\sigma)}(M)$ where E is a semantically secure symmetric encryption scheme (see [12]).

Security. The following theorem shows that FullIdent is a chosen ciphertext secure IBE (i.e. IND-ID-CCA), assuming WDH is hard.

Theorem 2. *Let $\mathcal{A}$ be a t-time* IND-ID-CCA *attacker on* FullIdent *that achieves advantage ϵ. Suppose $\mathcal{A}$ makes at most q_E extraction queries, at most q_D decryption queries, and at most q_H, q_{G1}, q_{H1} queries to the hash functions H, G_1, H_1 respectively. Then there is a t_1-time algorithm for WDH that achieves advantage ϵ_1 where*

$$t_1 = FO_{time}(t, q_{G_1}, q_{H_1})$$
$$\epsilon_1 = \Big(FO_{adv}(\epsilon(\tfrac{1}{eq_E} - \tfrac{q_D}{q}),\, q_{G_1},\, q_{H_1},\, q_D) - 1/2^n \Big) / q_H$$

where the functions FO_{time} and FO_{adv} are defined in Theorem 3.

The proof of the theorem is based on the theorem below due to Fujisaki and Okamoto (Theorem 14 in [12]). We state their theorem as it applies to the public key encryption scheme PubKeyEnc of the previous section. Let $\mathsf{PubKeyEnc}^{hy}$ be the result of applying the Fujisaki-Okamoto transformation to PubKeyEnc.

Theorem 3 (FO). *Suppose there is a* $(t, q_{G_1}, q_{H_1}, q_D)$ IND-CCA *attacker that achieves advantage* ϵ *when attacking* $\mathsf{PubKeyEnc}^{hy}$. *Then there is a* (t_1, ϵ_1) OWE *attacker on* PubKeyEnc *where*

$$t_1 = FO_{time}(t, q_{G_1}, q_{H_1}) = t + O((q_{G_1} + q_{H_1}) \cdot n), \quad \text{and}$$
$$\epsilon_1 = FO_{adv}(\epsilon, q_{G_1}, q_{H_1}, q_D) = \frac{1}{2(q_{G_1} + q_{H_1})} \left[(\epsilon + 1)(1 - 2/q)^{q_D} - 1\right]$$

We also need the following lemma to translate between an IND-ID-CCA chosen ciphertext attack on FullIdent and an IND-CCA chosen ciphertext attack on $\mathsf{PubKeyEnc}^{hy}$. The proof appears in the full version of the paper [2].

Lemma 3. *Let* $\mathcal{A}$ *be an* IND-ID-CCA *attacker that has advantage* ϵ *against the IBE scheme* FullIdent. *Suppose* $\mathcal{A}$ *makes at most* $q_E > 0$ *private key extraction queries and at most* q_D *decryption queries. Then there is an* IND-CCA *attacker* $\mathcal{B}$ *that has advantage at least* $\epsilon(\frac{1}{eq_E} - \frac{q_D}{q})$ *against* $\mathsf{PubKeyEnc}^{hy}$. *Its running time is* $O(time(\mathcal{A}))$.

Proof of Theorem 2. By Lemma 3 an IND-ID-CCA attacker on FullIdent implies an IND-CCA attacker on $\mathsf{PubKeyEnc}^{hy}$. By Theorem 3 an IND-CCA attacker on $\mathsf{PubKeyEnc}^{hy}$ implies a OWE attacker on PubKeyEnc. By Lemma 2 a OWE attacker on PubKeyEnc implies an algorithm for WDH. Composing all these reductions gives the required bounds. □

5 Extensions and Observations

Tate pairing and other curves. Our IBE system has some flexibility in terms of the curves being used and the definition of the pairing. For example, one could use the curve $y^2 = x^3 + x$ with its endomorphism $\phi : (x, y) \rightarrow (-x, iy)$ where $i^2 = -1$. We do not explore this here, but note that both encryption and decryption in FullIdent can be made faster by using the Tate pairing. In general, one can use any efficiently computable bilinear pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ between two groups $\mathbb{G}_1, \mathbb{G}_2$ as long as the WDH assumption holds. One would also need a way to map identities in $\{0,1\}^*$ uniformly onto $\mathbb{G}_1$.

Distributed PKG. In the standard use of an IBE in an e-mail system the master-key stored at the PKG must be protected in the same way that the private key of a CA is protected. One way of protecting this key is by distributing it among different sites using techniques of threshold cryptography [14]. Our IBE system supports this in a very efficient and robust way. Recall that the master-key is some $s \in \mathbb{F}_q$. in order to generate a private key the PKG computes $Q_{priv} = sQ_{\mathsf{ID}}$, where Q_{ID} is derived from the user's public key ID. This can easily be distributed in a t-out-of-n fashion by giving each of the n PKGs one share s_i of a Shamir secret sharing of $s \bmod q$. When generating a private key each of the t chosen PKGs simply responds with $Q_{priv}^{(i)} = s_i Q_{\mathsf{ID}}$. The user can then construct Q_{priv} as $Q_{priv} = \sum \lambda_i Q_{priv}^{(i)}$ where the λ_i's are the appropriate Lagrange coefficients.

Furthermore, it is easy to make this scheme robust against dishonest PKGs using the fact that DDH is easy in G_q (the group generated by P). During setup each of the n PKGs publishes $P_{pub}^{(i)} = s_i P$. During a key generation request the user can verify that the response from the i'th PKG is valid by testing that:

$$\hat{e}(Q_{priv}^{(i)}, P) = \hat{e}(Q_{\mathsf{ID}}, P_{pub}^{(i)})$$

Thus, a misbehaving PKG will be immediately caught. There is no need for zero-knowledge proofs as in regular robust threshold schemes. The PKG's master-key can be generated in a distributed fashion using the techniques of [15].

Note that a distributed master-key also enables decryption on a *per-message* basis, without any need to derive the corresponding decryption key. For example, threshold decryption of BasicIdent ciphertext (U, V) is straightforward if each PKG responds with $\hat{e}(s_i Q_{ID}, U)$.

Working in subgroups. The performance of our IBE system can be improved if we work in a small subgroup of the curve. For example, choose a 1024-bit prime $p = 2 \bmod 3$ with $p = aq - 1$ for some 160-bit prime q. The point P is then chosen to be a point of order q. Each public key ID is converted to a group point by hashing ID to a point Q on the curve and then multiplying the point by a. The system is secure if the WDH assumption holds in the group generated by P. The advantage is that Weil computations are done on points of small order, and hence is much faster.

IBE implies signatures. Moni Naor has observed that an IBE scheme can be immediately converted into a public key signature scheme. The intuition is as follows. The private key for the signature scheme is the master key for the IBE scheme. The public key for the signature scheme is the global system parameters for the IBE scheme. The signature on a message M is the IBE decryption key for $\mathsf{ID} = M$. To verify a signature, choose a random message M', encrypt M' using the public key $\mathsf{ID} = M$, and then attempt to decrypt using the given signature on M as the decryption key. If the IBE scheme is IND-ID-CCA, then the signature scheme is existentially unforgeable against a chosen message attack. Note that, unlike most signature schemes, the signature verification algorithm here is randomized. This shows that secure IBE schemes require both public key encryption and digital signatures. We note that the signature scheme derived from our IBE system has some interesting properties [3].

6 Escrow ElGamal encryption

In this section we note that the Weil pairing enables us to add a global escrow capability to the ElGamal encryption system. A single escrow key enables the decryption of ciphertexts encrypted under any public key. Paillier and Yung have shown how to add a global escrow capability to the Paillier encryption system [24]. Our ElGamal escrow system works as follows:

Setup: The algorithm works as follows:

Step 1: Choose a large k-bit prime p such that $p = 2 \bmod 3$ and $p = 6q - 1$ for some prime $q > 3$. Let E be the elliptic curve defined by $y^2 = x^3 + 1$ over $\mathbb{F}_p$. Choose an arbitrary $P \in E/\mathbb{F}_p$ of order q.

Step 2: Pick a random $s \in \mathbb{Z}_q$ and set $Q = sP$.

Step 3: Choose a cryptographic hash function $H : \mathbb{F}_{p^2} \rightarrow \{0,1\}^n$.

The message space is $\mathcal{M} = \{0,1\}^n$. The ciphertext space is $\mathcal{C} = E/\mathbb{F}_p \times \{0,1\}^n$. The system parameters are $\mathsf{params} = \langle p, n, P, Q, H \rangle$. The escrow key is $s \in \mathbb{Z}_q$.

keygen: A user generates a public/private key pair for herself by picking a random $x \in \mathbb{Z}_q$ and computing $P_{pub} = xP$. Her private key is x, her public key is P_{pub}.

Encrypt: To encrypt $M \in \{0,1\}^n$ under the public key P_{pub} do the following: (1) pick a random $r \in \mathbb{Z}_q$, and (2) set the ciphertext to be:

$$C = \langle rP,\ M \oplus H(g^r) \rangle \quad \text{where} \quad g = \hat{e}(P_{pub}, Q) \in \mathbb{F}_{p^2}$$

Decrypt: Let $C = \langle U, V \rangle$ be a ciphertext encrypted using P_{pub}. If $U \in E/\mathbb{F}_p$ is not a point of order q reject the ciphertext. To decrypt C using the private key x do:

$$V \oplus H(\hat{e}(U, xQ)) = M$$

Escrow-decrypt: To decrypt $C = \langle U, V \rangle$ using the escrow key s do:

$$V \oplus H(\hat{e}(U, sP_{pub})) = M$$

A standard argument shows that assuming WDH the system has semantic security in the random oracle model (recall that since DDH is easy we cannot prove semantic security based on DDH). Yet, the escrow agent can decrypt any ciphertext encrypted using any user's public key. The decryption capability of the escrow agent can be distributed using the PKG distribution techniques described in Section 5.

Using a similar hardness assumption, Verheul [30] has recently described an ElGamal encryption system with non-global escrow. Each user constructs a public key with two corresponding private keys, and gives one of the private keys to the trusted third party. Although both private keys can be used to decrypt, only the user's private key can be used simultaneously as the signing key for a discrete logarithm based signature scheme.

7 Summary and open problems

We defined chosen ciphertext security for identity-based systems and proposed a fully functional IBE scheme. The scheme has chosen ciphertext security in the random oracle model assuming WDH, a natural analogue of the computational Diffie-Hellman problem. The WDH assumption deserves further study considering the powerful cryptosystems derived from it. For example, it could be interesting to see whether the techniques of [20] can be used to prove that the WDH assumption is equivalent to the discrete log assumption on the curve for certain primes p.

It is natural to try and build chosen ciphertext secure identity based systems that are secure under standard complexity assumptions (rather than the random oracle model). One might hope to use the techniques of Cramer-Shoup [6] to provide chosen ciphertext security based on DDH. Unfortunately, as mentioned in Section 2 the DDH assumption is false in the group of points on the curve E. However, a natural variant of DDH does seem to hold. In particular, the following two distributions appear to be computationally indistinguishable: $\langle P, aP, bP, cP, abcP\rangle$ and $\langle P, aP, bP, cP, rP\rangle$ where a, b, c, r are random in $\mathbb{Z}_q$. We refer to this assumption as WDDH. It is natural to ask whether there is a chosen ciphertext secure identity-based system strictly based on WDDH. Such a scheme would be the analogue of the Cramer-Shoup system.

References

1. M. Bellare, A. Desai, D. Pointcheval, P. Rogaway, “Relations among notions of security for public-key encryption schemes”, Proc. Crypto ’98, pp. 26–45, 1998.
2. D. Boneh, M. Franklin, “Identity based encryption from the Weil pairing”, Full version available at `http://crypto.stanford.edu/ibe`
3. D. Boneh, B. Lynn, H. Shacham, “Short signatures from the Weil pairing”, manuscript.
4. M. Bellare, A. Boldyreva, S. Micali, “Public-key Encryption in a Multi-User Setting: Security Proofs and Improvements”, Proc. Eurocrypt 2000, LNCS 1807, 2000.
5. J. Coron, “On the exact security of Full-Domain-Hash”, Proc. of Crypto 2000.
6. R. Cramer and V. Shoup, “A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack”, in proc. Crypto ’98, pp. 13–25.
7. Y. Desmedt and J. Quisquater, “Public-key systems based on the difficulty of tampering”, Proc. Crypto ’86, pp. 111-117, 1986.
8. G. Di Crescenzo, R. Ostrovsky, and S. Rajagopalan, “Conditional Oblivious Transfer and Timed-Release Encryption”, Proc. of Eurocrypt ’99.
9. D. Dolev, C. Dwork, M. Naor, “Non-malleable cryptography”, SIAM J. of Computing, Vol. 30(2), pp. 391–437, 2000.
10. U. Feige, A. Fiat and A. Shamir, “Zero-knowledge proofs of identity”, *J. Cryptology*, vol. 1, pp. 77–94, 1988.
11. A. Fiat and A. Shamir, “How to prove yourself: Practical solutions to identification and signature problems”, Proc. Crypto ’86, pp. 186–194, 1986.
12. E. Fujisaki and T. Okamoto, “Secure integration of asymmetric and symmetric encryption schemes”, Proc. Crypto ’99, pp. 537–554, 1999.

13. G. Frey, M. Müller, H. Rück, "The Tate pairing and the discrete logarithm applied to elliptic curve cryptosystems", IEEE Tran. on Info. Th., Vol. 45, pp. 1717–1718, 1999.
14. P. Gemmell, "An introduction to threshold cryptography", in CryptoBytes, a technical newsletter of RSA Laboratories, Vol. 2, No. 7, 1997.
15. R. Gennaro, S. Jarecki, H. Krawczyk, T. Rabin, "Secure Distributed Key Generation for Discrete-Log Based Cryptosystems", *Advances in Cryptology – Eurocrypt '99*, Springer-Verlag LNCS 1592, pp. 295–310, 1999.
16. O. Goldreich, B. Pfitzmann and R. Rivest, "Self-delegation with controlled propagation -or- What if you lose your laptop", proc. Crypto '98, pp. 153–168, 1998.
17. A. Joux, "A one round protocol for tripartite Diffie-Hellman", Proc of ANTS 4, LNCS 1838, pp. 385–394, 2000.
18. A. Joux, K. Nguyen, "Separating Decision Diffie-Hellman from Diffie-Hellman in cryptographic groups", available from `eprint.iacr.org`.
19. S. Lang, "Elliptic functions", Addison-Wesley, Reading, 1973.
20. U. Maurer, "Towards proving the equivalence of breaking the Diffie-Hellman protocol and computing discrete logarithms", Proc. Crypto '94, pp. 271–281.
21. U. Maurer and Y. Yacobi, "Non-interactive public-key cryptography", proc. Eurocrypt '91, pp. 498–507.
22. A. Menezes, T. Okamoto, S. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field", IEEE Tran. on Info. Th., Vol. 39, pp. 1639–1646, 1993.
23. V. Miller, "Short programs for functions on curves", unpublished manuscript.
24. P. Paillier and M. Yung, "Self-escrowed public-key infrastructures" in Proc. ICISC, pp. 257–268, 1999.
25. C. Rackoff, D. Simon, "Noninteractive zero-knowledge proof of knowledge and chosen ciphertext attack", in proc. Crypto '91, pp. 433–444, 1991.
26. R. Rivest, A. Shamir and D. Wagner, "Time lock puzzles and timed release cryptography," Technical report, MIT/LCS/TR-684
27. A. Shamir, "Identity-based cryptosystems and signature schemes", Proc. Crypto '84, pp. 47–53.
28. S. Tsuji and T. Itoh, "An ID-based cryptosystem based on the discrete logarithm problem", *IEEE Journal on Selected Areas in Communication*, vol. 7, no. 4, pp. 467–473, 1989.
29. H. Tanaka, "A realization scheme for the identity-based cryptosystem", Proc. Crypto '87, pp. 341–349, 1987.
30. E. Verheul, "Evidence that XTR is more secure than supersingular elliptic curve cryptosystems", Proc. Eurocrypt 2001.