PSEC–3: Provably Secure Elliptic Curve Encryption Scheme – V3 (Submission to P1363a)

Tatsuaki Okamoto[1] and David Pointcheval[2]

[1] NTT Labs, 1-1 Hikarinooka, Yokosuka-shi 239-0847 Japan.
E-mail: okamoto@isl.ntt.co.jp.

[2] Dépt d'Informatique, ENS – CNRS, 45 rue d'Ulm, 75230 Paris Cedex 05, France.
E-mail: David.Pointcheval@ens.fr – URL: http://www.di.ens.fr/~pointche.

May 2000

Abstract.

We describe a new version of the elliptic curve encryption schemes PSEC (Provably Secure Elliptic Curve). PSEC–3 is a public-key encryption system that uses the elliptic curve El Gamal trapdoor function and two random functions (hash functions) as well as any semantically secure symmetric encryption scheme, such as the one-time pad, or any classical block-cipher.

Furthermore, we define a new problem, the Elliptic Curve Gap Diffie-Hellman problem (EC—Gap–DH) which is likely stronger than the more classical Elliptic Curve Decision Diffie-Hellman (EC–DDH) problem. Indeed, its tractability would imply the equivalence between the Computational and the Decisional versions of the Elliptic Curve Diffie-Hellman problem.

PSEC–3 therefore has several outstanding properties as follows:

1. with the one-time pad, PSEC–3 is semantically secure or non-malleable against chosen-ciphertext attacks (IND-CCA2 or NM-CCA2), in the random oracle model, under the Elliptic Curve Gap Diffie-Hellman (EC–Gap-DH) assumption.
2. with any symmetric encryption, PSEC–3 is semantically secure or non-malleable against chosen-ciphertext attacks (IND-CCA2 or NM-CCA2), in the random oracle model, under the Elliptic Curve Gap Diffie-Hellman (EC–Gap-DH) assumption, if the underlying symmetric encryption is simply semantically secure against passive attacks.
3. if the underlying random functions are replaced by practical random-like functions (e.g., SHA and MD5), PSEC–3 is as efficient as the basic Elliptic Curve El Gamal scheme, for the encryption process but also for the decryption process, which is the major novelty of this new proposal.

The encryption scheme described in this contribution is obtained by using a new result on conversion techniques using random functions by the authors.

Table of Contents

1 Background: Provable Security **3**

2 Description of PSEC–3 **3**
2.1 Overview 3
2.2 Setup: $\mathcal{G}$ 4
2.3 Key Generation: $\mathcal{K}$ 4
2.4 Encryption: $\mathcal{E}$ 5
2.5 Decryption: $\mathcal{D}$ 5

3 Security Assessment of PSEC–3 **6**

4 Attributes and Advantages of PSEC–3 **7**
4.1 Security of OTP—PSEC–3 7
4.2 Security of PSEC–3 with any Symmetric Encryption 7
4.3 Efficiency 8

5 Limitations **8**

6 Intellectual Property Statement **9**

1 Background: Provable Security

During a long time, heuristic security has been accepted by all the people and even the standard organizations. After many recent attacks against such "heuristically secure schemes" [6, 8, 17, 15], everybody realized the importance of provable security.

For public-key encryption, the strongest security notion, among all those that have been defined to capture the standard adversary scenarios, is by now called the *chosen-ciphertext security*. Indeed, it prevents [1] both the distinction of encrypted messages (semantic security [16]) and the malleability of ciphertexts [11] for an adversary who can ask the decryption of any ciphertext of her choice (the adaptive chosen-ciphertext attacks [28]).

A promising way to construct a practical public-key encryption scheme that reaches the chosen-ciphertext security is to convert a primitive trapdoor one-way function (such as RSA [30] or El Gamal [12]) by using *random functions*. Here, some hash functions, such as MD5 [29] or SHA-1 [19], are assumed to behave like ideally random functions. This so-called *random oracle model* [3] has already been widely used to provide efficient and provably secure schemes, for both signature [5, 27, 20, 2] and public-key encryption [4].

Although security in the random oracle model cannot be guaranteed formally when a practical random-like function is used in place of the random oracle, this paradigm often yields much more efficient schemes than those in the *standard model* and gives strong security arguments.

Two typical primitives of the trapdoor one-way functions are deterministic one-way permutations (e.g., the RSA function [30]) and probabilistic one-way functions (e.g., El Gamal [10, 12], Okamoto-Uchiyama [23] and Paillier [25] functions).

Bellare and Rogaway [4] presented a generic and efficient way to convert a trapdoor one-way permutation into a chosen-ciphertext secure scheme, in the random oracle model. The scheme created this way from the RSA function is often called OAEP. However, their method cannot be applied to probabilistic trapdoor one-way functions such as El Gamal, because it requires the permutation property.

Very recently the authors, together with other people [13, 14, 26, 22] proposed some generic conversions from any probabilistic trapdoor one-way function into a chosen-ciphertext secure encryption scheme. The first two conversions led to the EPOC [24] and PSEC [21] IEEE P1363a proposals.

The most recent conversion can apply to any (partially) trapdoor one-way function into a chosen-ciphertext secure encryption scheme, in an optimal way, from the computational point of view. Indeed, all the previous conversions required a re-encryption in the decryption phase to check the validity of the ciphertext. This new conversion just needs the basic decryption, without re-encryption. Furthermore, this conversion can be combined with a symmetric encryption scheme to reach high-speed rates.

2 Description of PSEC–3

2.1 Overview

This section describes the proposed third version of the public-key encryption scheme, PSEC, which is specified by a quadruple $(\mathcal{G}, \mathcal{K}, \mathcal{E}, \mathcal{D})$, where $\mathcal{G}$ sets the system up, gener-

ating the common parameters, $\mathcal{K}$ produces the key-pair for each user, $\mathcal{E}$ is the encryption algorithm and $\mathcal{D}$ the decryption algorithm.

In that description, we assume $\mathsf{SymE} = (\mathsf{E}, \mathsf{D})$ to be a symmetric encryption scheme which uses a $kLen$-bit key.

Remark 1. A typical way to realize SymE is the one-time pad:

$$\mathsf{E}_k(m) = K \oplus m \qquad \mathsf{D}_k(c) = K \oplus c,$$

where $\oplus$ denotes the bit-wise exclusive-or operation. Therefore, $mLen = kLen$, where $mLen$ denotes the maximal message-size which can be securely encrypted.

2.2 Setup: $\mathcal{G}$

The input and output of the setup algorithm $\mathcal{G}$ are as follows:

[Input] Security parameter k $(= pLen)$, which is a positive integer.

[Output] An elliptic curve-based cyclic group, defined by a finite field, an elliptic curve and a generator:

- q for a finite field $\mathbb{F}_q$
- two elliptic curve coefficients a and b, elements of $\mathbb{F}_q$, that define an elliptic curve E
- a positive prime integer p dividing the number of points on E
- a curve point P of order p.

Here the size of p should be k (i.e., $|p| = k$). Therefore, set $pLen \leftarrow k$, and $qLen \leftarrow |q|$ as well as $mLen$ and $kLen$ depending on SymE and $hLen$ linearly dependent in k.

Moreover, one selects two hash functions

$$G : \{0,1\}^{qLen} \longrightarrow \{0,1\}^{kLen}$$
$$H : \{0,1\}^{|C_1|+2\cdot qLen+mLen} \longrightarrow \{0,1\}^{hLen}$$

Here, $|C_1| = qLen + 1$, if a point is represented by its x-coordinate and 1 bit signature. ($|C_1| = 2{\cdot}qLen$, if a point is represented by a pair of its x-coordinate and y-coordinate.)

2.3 Key Generation: $\mathcal{K}$

The input and output of $\mathcal{K}$ are as follows:

[Input] The common parameters $(q, a, b, p, P, G, H, qLen, pLen, hLen, mLen)$ defined by the setup algorithm.

[Output] A pair $(\mathsf{pk}, \mathsf{sk})$ of matching public and secret keys, where sk is randomly chosen in $\mathbb{Z}_p^*$ and pk is a point on E, $\mathsf{pk} \leftarrow \mathsf{sk} \cdot P$.

2.4 Encryption: $\mathcal{E}$

The input and output of $\mathcal{E}$ are as follows:

[Input] Plaintext $m \in \{0,1\}^{mLen}$ along with the public-key pk, as well as the common parameters $(q, a, b, p, P, G, H, qLen, pLen, hLen, mLen)$.

[Output] Ciphertext $c = (C_1, c_1', c_2, c_3)$.

The operation of $\mathcal{E}$, on input m, pk and $(q, a, b, p, P, G, H, qLen, pLen, hLen, mLen)$, is as follows:

- Select $R \in \{0,1\}^{qLen}$ uniformly.
- Select $r \in \mathbb{Z}_p^\star$ uniformly and compute the points on E, $C_1 \leftarrow r \cdot P$ and $T \leftarrow r \cdot \mathsf{pk}$.
- Compute $c_1' \leftarrow x_T \oplus R$, $K \leftarrow G(R)$ and $c_3 \leftarrow H(C_1, c_1', R, m)$, where x_T is the x-coordinate of T.
- Compute $c_2 \leftarrow \mathsf{E}_K(m)$.

2.5 Decryption: $\mathcal{D}$

The input and output of $\mathcal{D}$ are as follows:

[Input] Ciphertext $c = (C_1, c_1', c_2, c_3)$ along with public-key pk and the common parameters $(q, a, b, p, P, G, H, qLen, pLen, hLen, mLen)$ but also the secret-key sk.

[Output] Plaintext m or null string.

The operation of $\mathcal{D}$ is as follows:

- Compute the point on E, $T' \leftarrow \mathsf{sk} \cdot C_1$ (which should be equal to the above T-point) and $R' \leftarrow c_1' \oplus x_{T'}$ (which should be equal to the above R).
- Compute $K' \leftarrow G(R')$ and $m' \leftarrow \mathsf{D}_{K'}(c_2)$.
- Check whether the following equation holds or not:

$$c_3 \stackrel{?}{=} H(C_1, c_1', R', m').$$

- If it holds, output m' as the decrypted plaintext. Otherwise, output null string.

Remark 2. Since the domains of G and H are fixed by the parameters of $qLen$ and others, only $R' \in \{0,1\}^{qLen}$ is accepted by the decryption procedure, $\mathcal{D}$. (Note that the domains of G and H in the conversion of [22] are fixed by the domain of the underlying encryption function and other parameters.) More explicitly, in $\mathcal{D}$, check if $R' \in \{0,1\}^{qLen}$ (see a recent remark [18]).

3 Security Assessment of PSEC–3

This section reviews some results on the security of PSEC–3. They are easily obtained from [22].

Definition 1 (EC–DH Assumption). Let $\mathcal{G}$ be the setup algorithm of PSEC–3, and (q, a, b, p, P) be a part of the common parameters. Let r and s be uniformly selected in $\mathbb{Z}_p$ and set $R \leftarrow r \cdot P$ and $S \leftarrow s \cdot P$.

The *Elliptic Curve Diffie-Hellman (EC–DH) problem* is *intractable*, if for any probabilistic polynomial time machine $\mathcal{A}$, for any constant c, for sufficiently large k $(= pLen)$,

$$\Pr[\mathcal{A}(q, a, b, p, P, R, S) = x_T] < 1/k^c,$$

where $T = rs \cdot P$ and x_T is the x-coordinate of T. The probability is taken over the coin flips of $\mathcal{G}$ and $\mathcal{A}$ as well as the random choice of r and s.

The assumption that the *Elliptic Curve Diffie-Hellman problem* is *intractable* is called the *Elliptic Curve Diffie-Hellman assumption*.

Definition 2 (EC–DDH Assumption). Let $\mathcal{G}$ be the setup algorithm of PSEC–3, and (q, a, b, p, P) be a part of the common parameters. Let r, s and t be uniformly selected in $\mathbb{Z}_p$, and set $R \leftarrow r \cdot P$, $S \leftarrow s \cdot P$, $T \leftarrow t \cdot P$ and $U \leftarrow rs \cdot P$. Let b be a random coin. If $b = 0$, set $v \leftarrow x_T$, otherwise set $v \leftarrow x_U$.

The *Elliptic Curve Decision Diffie-Hellman (EC–DDH) problem* is *intractable*, if for any probabilistic polynomial time machine $\mathcal{A}$, for any constant c, for sufficiently large k $(= pLen)$,

$$\Pr[\mathcal{A}(q, a, b, p, P, R, S, v) = b] < 1/2 + 1/k^c.$$

The probability is taken over the coin flips of $\mathcal{G}$ and $\mathcal{A}$ as well as the random choice of r, s, t and b.

The assumption that the *Elliptic Curve Decision Diffie-Hellman problem* is *intractable* is called the *Elliptic Curve Decision Diffie-Hellman assumption*.

Definition 3 (EC–GDH Assumption). The *Elliptic Curve Gap Diffie-Hellman (EC–GDH) problem* is *intractable*, if the EC–DH problem is still intractable even for an adversary who has access to an oracle that perfectly answers the EC–DDH problem.

The assumption that the *Elliptic Curve Gap Diffie-Hellman problem* is *intractable* is called the *Elliptic Curve Gap Diffie-Hellman assumption*.

Definition 4 (Security of Symmetric Encryption). Let $\mathcal{A}$ be an adversary that runs in two stages. In the first stage, $\mathcal{A}$ endeavors to come up with a pair of equal-length messages, m_0 and m_1, along with some state information s, where $|m_0| = |m_1| \leq kLen^a$, for some constant a. In the second stage, $\mathcal{A}$ is given a ciphertext $c \leftarrow \mathsf{E}_K(m_b)$, where $K \in \{0,1\}^{kLen}$ and $b \in \{0,1\}$ are randomly and uniformly chosen.

SymE is *secure against passive attacks* if for any probabilistic polynomial time machine $\mathcal{A}$, for any constant d, for sufficiently large $kLen$,

$$\Pr[\mathcal{A}(kLen, m_0, m_1, s, c) = b] < 1/2 + 1/kLen^d.$$

The probability is taken over the coin flips of $\mathcal{A}$ as well as the random choice of K and b.

Theorem 1 (OTP—PSEC–3). *Let* SymE *be the one-time pad, and thus* $mLen = kLen$. *Let* $hLen = pLen/a$ *for some constant* a. *OTP—PSEC–3 is chosen-ciphertext secure in the random oracle model, provided that the EC–GDH assumption holds.*

Theorem 2 (PSEC–3). *Let* $hLen = pLen/a$ *for some constant* a. *PSEC–3 is chosen-ciphertext secure in the random oracle model, provided that the EC–GDH assumption holds and that the underlying* SymE *is secure against passive attacks, for suitable* $kLen$ *and* $mLen$.

Remark 3. We can also give the concrete efficiency analysis of the reduction for proving the security, and show that our reduction is efficient [22], and even optimal since the probability of breaking the EC–GDH problem is almost the same as the advantage of an adaptive adversary in breaking the chosen-ciphertext security.

4 Attributes and Advantages of PSEC–3

4.1 Security of OTP—PSEC–3

If the Elliptic Curve Gap Diffie-Hellman (EC–GDH) assumption holds, PSEC–3 with one-time pad is secure in the strongest sense, in the random oracle model, if the parameters are appropriately selected.

Note that this assumption is quite new. But one can easily show that if the EC–GDH problem is not intractable, then the EC–DH and EC–DDH problems are equivalent. However this latter equivalence is very unlikely, and certainly more unlikely than the tractability of the EC–DDH problem.

The elliptic curve version of the Cramer-Shoup (EC–CS) scheme [9] is provably secure in the *standard model* (i.e., without any ideal assumption), however it is based on that likely stronger number theoretic assumption, the EC–DDH assumption. Furthermore, as we see below, it is less efficient than ours.

4.2 Security of PSEC–3 with any Symmetric Encryption

If the Elliptic Curve Gap Diffie-Hellman (EC–GDH) assumption holds and the underlying symmetric encryption is secure against passive attacks, PSEC–3 with the symmetric encryption is secure in the strongest sense, in the random oracle model, if the parameters are appropriately selected.

The advantage of this scheme is that security in the strongest sense is guaranteed for the total system that integrates the asymmetric and symmetric encryption schemes. Therefore, even if the underlying symmetric encryption is secure only against passive attacks (we do not care about active attacks), PSEC–3 guarantees security against adaptive chosen-ciphertexts attacks (IND-CCA2).

An additional property of PSEC–3 (as other PSEC versions) is authentication and integrity without using any MAC function. That is, the recipient can confirm whether the decrypted message is the same as the one the originator sent.

Finally, it also provides a key distribution with session key encryption and then a symmetric multi-message encryption which achieves chosen-ciphertext security. Indeed,

the ciphertext can be split, the first part (C_1, c'_1) is a constant overhead, and then the second part (c_2, c_3) can be reiterated with many plaintexts:

- Select $R \in \{0,1\}^{qLen}$ uniformly.
- Select $r \in \mathbb{Z}_p^\star$ uniformly and compute the points on E, $C_1 \leftarrow r \cdot P$ and $T \leftarrow r \cdot \mathsf{pk}$.
- Compute $c'_1 \leftarrow x_T \oplus R$ and $K \leftarrow G(R)$.
- Then, for any message m_i, compute $c_{2,i} \leftarrow \mathsf{E}_K(m_i)$ and $c_{3,i} \leftarrow H(C_1, c'_1, R, m_i)$, and send the tuple $(C_1, c'_1, c_{2,i}, c_{3,i})$.

4.3 Efficiency

The OTP-variant is the most efficient scheme among all the EC–DH based encryption schemes (see Figure 1). Since it furthermore allows symmetric integration with multi-message encryption, it achieves an unbeatable efficiency.

Scheme	Security	Number Theoretical Assumption	Hash Function Assumption	Encryption Cost	Decryption Cost
OTP—PSEC–3	IND-CCA2	EC–GDH	Random Oracle	2E + 2H	1E + 2H
OTP—PSEC–2	IND-CCA2	EC–DH	Random Oracle	2E + 2H	2E + 2H
OTP—PSEC–1	IND-CCA2	EC–DDH	Random Oracle	2E + 1H	2E + 1H
EC–Cramer-Shoup	IND-CCA2	EC–DDH	UOWHF	5E + 1H	6E + 1H
EC–El Gamal	IND-CPA	EC–DDH	None	2E	1E

Fig. 1: Comparison of the different EC–DH-based encryption schemes, where "E" and "H" denote the costs of an exponentiation and a hashing, respectively.

Under the most practical environment of using public-key cryptosystems, where a public-key cryptosystem is used associated with symmetric encryption (e.g., triple-DES, IDEA or any candidate of the AES), a typical example of the parameters is as follows: first, any message and any list of messages can be encrypted (i.e., $mLen = \star$), then $gLen = kLen = 128$ (or 256, according to the block-cipher), $hLen = 64$, and $pLen = k = 160$. The encryption and decryption speeds of PSEC–3 are exactly the same as those of the basic elliptic curve El Gamal scheme [12], plus just 2 more hashings. Therefore, it is more than three times faster than those of the elliptic curve Cramer-Shoup scheme [9].

5 Limitations

Recently Canetti *et al.* [7] have demonstrated that it is possible to devise cryptographic protocols which are provably secure in the random oracle model but for which no complexity assumption property instantiates the random-oracle-modeled hash function. However,

the examples they used to make the random oracle model paradigm fail were very contrived, so the concerns induced by these examples do not appear to apply to any of the concrete practical schemes that have been proven secure in the random oracle model.

6 Intellectual Property Statement

NTT has filed patent applications on the techniques used in this contribution. NTT will license any resulting patent in a reasonable and non-discriminatory fashion. A letter to this effect will be provided.

References

[1] M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway. Relations among Notions of Security for Public-Key Encryption Schemes. In *Crypto '98*, LNCS 1462, pages 26–45. Springer-Verlag, Berlin, 1998.

[2] M. Bellare and P. Rogaway. PSS: Provably Secure Encoding Method for Digital Signatures. Submission to IEEE P1363a. August 1998. Available from `http://grouper.ieee.org/groups/1363/`.

[3] M. Bellare and P. Rogaway. Random Oracles Are Practical: a Paradigm for Designing Efficient Protocols. In *Proc. of the 1st CCS*, pages 62–73. ACM Press, New York, 1993.

[4] M. Bellare and P. Rogaway. Optimal Asymmetric Encryption – How to Encrypt with RSA. In *Eurocrypt '94*, LNCS 950, pages 92–111. Springer-Verlag, Berlin, 1995.

[5] M. Bellare and P. Rogaway. The Exact Security of Digital Signatures – How to Sign with RSA and Rabin. In *Eurocrypt '96*, LNCS 1070, pages 399–416. Springer-Verlag, Berlin, 1996.

[6] D. Bleichenbacher. A Chosen Ciphertext Attack against Protocols based on the RSA Encryption Standard PKCS #1. In *Crypto '98*, LNCS 1462, pages 1–12. Springer-Verlag, Berlin, 1998.

[7] R. Canetti, O. Goldreich, and S. Halevi. The Random Oracles Methodology, Revisited. In *Proc. of the 30th STOC*, pages 209–218. ACM Press, New York, 1998.

[8] J. S. Coron, D. Naccache, and J. P. Stern. On the Security of RSA Padding. In *Crypto '99*, LNCS 1666, pages 1–18. Springer-Verlag, Berlin, 1999.

[9] R. Cramer and V. Shoup. A Practical Public Key Cryptosystem Provably Secure against Adaptive Chosen Ciphertext Attack. In *Crypto '98*, LNCS 1462, pages 13–25. Springer-Verlag, Berlin, 1998.

[10] W. Diffie and M. E. Hellman. New Directions in Cryptography. *IEEE Transactions on Information Theory*, IT–22(6):644–654, November 1976.

[11] D. Dolev, C. Dwork, and M. Naor. Non-Malleable Cryptography. In *Proc. of the 23rd STOC*. ACM Press, New York, 1991.

[12] T. El Gamal. A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. *IEEE Transactions on Information Theory*, IT–31(4):469–472, July 1985.

[13] E. Fujisaki and T. Okamoto. How to Enhance the Security of Public-Key Encryption at Minimum Cost. In *PKC '99*, LNCS 1560, pages 53–68. Springer-Verlag, Berlin, 1999.

[14] E. Fujisaki and T. Okamoto. Secure Integration of Asymmetric and Symmetric Encryption Schemes. In *Crypto '99*, LNCS 1666, pages 537–554. Springer-Verlag, Berlin, 1999.

[15] M. Girault and J. F. Misarsky. Cryptanalysis of Countermeasures Proposed for Repairing IOS/IEC 9796-1. In *Eurocrypt '2000*, LNCS. Springer-Verlag, Berlin, 2000.

[16] S. Goldwasser and S. Micali. Probabilistic Encryption. *Journal of Computer and System Sciences*, 28:270–299, 1984.

[17] F. Grieu. A Chosen Message Attack on ISO/IEC 9796-1 Signature Scheme. In *Eurocrypt '2000*, LNCS. Springer-Verlag, Berlin, 2000.

[18] M. Joye, J. J. Quisquater, and M. Yung. On the Power of Misbehaving Adversaries and Security Analysis of EPOC. Manuscript, February 2000.

[19] NIST. *Secure Hash Standard* (SHS). Federal Information Processing Standards PUBlication 180–1, April 1995.

[20] K. Ohta and T. Okamoto. On Concrete Security Treatment of Signatures Derived from Identification. In *Crypto '98*, LNCS 1462, pages 354–369. Springer-Verlag, Berlin, 1998.

[21] T. Okamoto, E. Fujisaki, and H. Morita. PSEC: Provably Secure Elliptic Curve Encryption Scheme. Submission to IEEE P1363a. March 1999. Available from `http://grouper.ieee.org/groups/1363/`.

[22] T. Okamoto and D. Pointcheval. OCAC: an Optimal Conversion for Asymmetric Cryptosystems, 2000. Manuscript, available from the authors.

[23] T. Okamoto and S. Uchiyama. A New Public Key Cryptosystem as Secure as Factoring. In *Eurocrypt '98*, LNCS 1403, pages 308–318. Springer-Verlag, Berlin, 1998.

[24] T. Okamoto, S. Uchiyama, and E. Fujisaki. EPOC: Efficient Probabilistic Public-Key Encryption. Submission to IEEE P1363a. November 1998. Available from `http://grouper.ieee.org/groups/1363/`.

[25] P. Paillier. Public-Key Cryptosystems Based on Discrete Logarithms Residues. In *Eurocrypt '99*, LNCS 1592, pages 223–238. Springer-Verlag, Berlin, 1999.

[26] D. Pointcheval. The Composite Discrete Logarithm and Secure Authentication. In *PKC '2000*, LNCS 1751, pages 113–128. Springer-Verlag, Berlin, 2000.

[27] D. Pointcheval and J. Stern. Security Proofs for Signature Schemes. In *Eurocrypt '96*, LNCS 1070, pages 387–398. Springer-Verlag, Berlin, 1996.

[28] C. Rackoff and D. R. Simon. Non-Interactive Zero-Knowledge Proof of Knowledge and Chosen Ciphertext Attack. In *Crypto '91*, LNCS 576, pages 433–444. Springer-Verlag, Berlin, 1992.

[29] R. Rivest. The MD5 Message-Digest Algorithm. RFC 1321, The Internet Engineering Task Force, April 1992.

[30] R. Rivest, A. Shamir, and L. Adleman. A Method for Obtaining Digital Signatures and Public Key Cryptosystems. *Communications of the ACM*, 21(2):120–126, February 1978.