

Practical Identification Schemes
as Secure as the DL and RSA problems
(Submission to P1363a)

Tatsuaki Okamoto

NTT Laboratories
1-1 Hikarino-oka, Yokosuka-shi, 239-0847 Japan
Email: okamoto@sucaba.isl.ntt.co.jp

March 1999

Abstract

We present a practical three-move interactive identification scheme, ID-DL, and prove it to be as secure as the discrete logarithm problem. ID-DL is almost as efficient as the Schnorr identification scheme, which is not provably secure. We also present another practical identification scheme, ID-RSA, which is proven to be as secure as the RSA problem and almost as efficient as the Guillou-Quisquater identification scheme; the Guillou-Quisquater scheme is not provably secure. The elliptic curve version of ID-DL, ID-ECDL, is also suggested.

The identification schemes described in this contribution were originally presented at Crypto'92 by the author [23].

Contents

1	Background	3
2	Description of Identification Schemes	5
2.1	Overview	5
2.2	ID-DL	5
2.2.1	Key Generation: \mathcal{G}	5
2.2.2	Interactive Protocol: (A, B)	6
2.3	ID-RSA	6
2.3.1	Key Generation: \mathcal{G}	6
2.3.2	Interactive Protocol: (A, B)	7
2.4	ID-ECDL: Elliptic curve version of ID-DL	7
3	Attributes and Advantages of ID-DL and ID-RSA	7
4	Security Assessment of ID-DL, ID-ECDL and ID-RSA	8
5	Limitations	9
6	Intellectual Property Statement	9
	Appendix	12

1 Background

Public-key based identification schemes are very useful and fundamental tools in many applications such as electronic fund transfer and online systems for preventing data access by invalid users.

Identification schemes are typical applications of zero-knowledge interactive proofs [14], and several practical zero-knowledge identification schemes have been proposed [1, 12, 11, 22]. However, the zero-knowledge identification schemes have the following shortcomings in practice, where we simply call “black-box simulation zero-knowledge” “zero-knowledge”, since we do not know of any effective measure to prove zero-knowledgeness except the black-box simulation technique, although “auxiliary-input zero-knowledge” is more general than “black-box simulation zero-knowledge”:

- A zero-knowledge identification scheme requires more than three interactions (three-moves¹) from Goldreich et al.’s result [13] unless the language for the proof is trivial. A zero-knowledge protocol is less practical than the corresponding (three-move) parallel version since interaction over a network often requires more time than that requested for the calculations performed in these identification schemes. Although four-move and five-move zero-knowledge proofs have been proposed [5, 10], these protocols impose considerably larger communication and computation overheads compared to the three-move parallel versions (especially Type 2 below).

Note: Here, the “(three-move) parallel version” denotes two types of protocols. One (Type 1) is just the parallel execution of a zero-knowledge protocol (e.g., the three-move version of the Fiat-Shamir scheme with $k = 1$ and $t = Poly(|n|)$ [12]). The other (Type 2) is a protocol that can be converted to zero-knowledge by executing the protocol repeatedly many times and holding the security parameter of one repetition constant (e.g., the three-move and higher-degree version of the Fiat-Shamir scheme [15, 22]). The communication complexity of the Type 1 protocol is the same as that of the original zero-knowledge protocol. Usually, the communication complexity of the Type 2 protocol is much less than that of the corresponding zero-knowledge protocol (or Type 1).

- No zero-knowledge identification can be converted into a signature scheme using Fiat-Shamir’s technique [12], which is a truly practical way of converting an identification scheme into a signature scheme with a one-way hash function. This is because: if the identification protocol is zero-knowledge, the signature converted from this identification protocol through Fiat-Shamir’s technique can be forged by using the same algorithm as the simulation for proving the zero-knowledgeness of the identification protocol. Therefore, for example, the above-mentioned four-move and five-move zero-knowledge proofs [5, 10] cannot be used to construct a signature scheme.

In contrast, the three-move identification schemes [1, 3, 12, 11, 15, 22, 27], which are the parallel version (Type 2) of zero-knowledge proofs, have the following merits in practice.

¹A scheme is called “one-move” if prover A only sends one message to verifier B , and is called “two-move” if B sends to A and then A sends to B . “ j -move” is defined in the obvious way.

- The communication and computation overheads are smaller than those of the zero-knowledge identification schemes.
- The three-move identification schemes can be converted into practical signature schemes by using Fiat-Shamir’s technique.

How then can we prove the security of the three-move identification schemes? As mentioned above, the zero-knowledge notion seems to be ineffective for this purpose. Feige, Fiat and Shamir [11] have developed an effective measure called “no-useful information transfer” to prove the security of their three-move identification scheme. Ohta and Okamoto [22] have proposed a variant called “no transferable information with (sharp threshold) security level,” which characterizes the security level theoretically. Therefore, only “no-useful information transfer” [11] and its variant [22] have been known to be effective in proving the security of three-move identification schemes.

Some three-move identification schemes [11, 22, 3] have been proven to be secure assuming reasonable primitive problems, in the sense of [11, 22]. The Feige-Fiat-Shamir identification scheme [11], based on square root mod n , has been proven to be as secure as the factoring problem. The Ohta-Okamoto scheme [22], which is the higher (the L -th) degree modification of the Feige-Fiat-Shamir scheme, has been proven to be as secure (with sharp threshold security level $1/K$) as factoring, where $v^{1/L} \bmod n$ has at least K solutions (e.g., $\gcd(L, p-1) = K$; see [22] for more detail conditions). The Brickell-McCurley scheme [3], which is a modification of the Schnorr scheme [27], has been proven to be secure assuming the intractability of finding a factor, q , of $p-1$, given additional information g whose order is q in \mathbf{Z}_p^* , although the security of their scheme also depends on the difficulty of the discrete logarithm.

Although their schemes are efficient, they have several shortcomings in practice: the transmitted information size and memory size cannot be small simultaneously [11], and a priori fixed value v (e.g., v is the identity of a user) cannot be used as a public key [22], (i.e., an identity based scheme [28] cannot be constructed on this scheme). In addition, the security assumption of [3] is stronger than the ordinary factoring problem (or the level of the provable security is lower than those of [11, 22]).

In contrast, other previously proposed practical three-move identification schemes, the Schnorr [27] and Guillou-Quisquater [15] schemes, have some merits compared to [11, 22, 3]: The security of the Schnorr scheme depends on the discrete logarithm, which is a promising alternative if factoring becomes tractable, since we have several different types of discrete logarithms such as elliptic curve logarithms which may be more intractable than factoring. Moreover, the transmitted information size and memory size with these schemes can be small simultaneously, while it is impossible in [11]. The Schnorr scheme is more efficient than [3]. In addition, in the Guillou-Quisquater scheme, a priori fixed value v can be used as the public key. Unfortunately, the Schnorr and Guillou-Quisquater schemes are not provably secure. The difficulty of proving the security of these schemes resides in the fact that the discrete logarithm and RSA inversion have single solutions in restricted domains, that is, $\log_g x \bmod p$ has a single solution (x is in the restricted domain, $\{0, 1, \dots, \text{ord}(g) - 1\}$), and $x^{1/e} \bmod n$ has also a single solution ($\gcd(e, \phi(n)) = 1$, ϕ is the Euler function).

This contribution presents three-move identification schemes that are proven to be as secure as the discrete logarithm or RSA inversion. The schemes inherit almost all of the merits of

the Schnorr and Guillou-Quisquater schemes and at the same time are provably secure. That is, these schemes are almost as efficient as the Schnorr and Guillou-Quisquater identification schemes from all practical viewpoints such as communication overhead, interaction number, required memory size, and processing speed. In addition, the new schemes duplicate the other advantage of the Guillou-Quisquater scheme: they can be used to construct the identity based schemes.

2 Description of Identification Schemes

2.1 Overview

This section describes the two proposed identification schemes, ID-DL and ID-RSA: ID-DL is as secure as the discrete logarithm problem, and ID-RSA is as secure as the RSA problem. These schemes are specified by $(\mathcal{G}, (A, B))$, where \mathcal{G} is the key generation operation, (A, B) is the three-move interactive protocol between A (prover) and B (verifier).

2.2 ID-DL

2.2.1 Key Generation: \mathcal{G}

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

[**Input**] Security parameter l , which is a positive integer.

[**Output**] A pair of public-key, (p, q, g_1, g_2, t, v) , and secret-key, (s_1, s_2) .

The operation of \mathcal{G} , on input 1^l , is as follows:

- Select primes p and q such that $q|p-1$ and $|q|=l$. (e.g., $q \geq 2^{160}$, and $p \geq 2^{1024}$.)
- Select g_1, g_2 of order q in the group \mathbf{Z}_p^* , and an integer $t = O(|p|)$. (e.g., $t \geq 20$.) Here, if g_2 is calculated by $g_2 = g_1^\alpha \bmod p$, α can be discarded after publishing g_2 .
- Select random integers s_1, s_2 in \mathbf{Z}_q , and compute $v = g_1^{-s_1} g_2^{-s_2} \bmod p$.

Remark: (p, q, g_1, g_2, t) can be published by a system manager and used commonly by all system users as a system parameter. The system manager should then also publish some information to confirm to users that these parameters were selected honestly. For example, (s)he publishes some witness that no trapdoor exists in p, g_1, g_2 , or that these values are generated honestly. Since the primality test for p and q is fairly easy for users, they can confirm for themselves that g_1 and g_2 are both of order q . When, as described above, the system parameter is generated and published by each user individually, (s)he does not need to publish such information.

2.2.2 Interactive Protocol: (A, B)

The input and output of (A, B) are as follows:

[**Input**] The common (shared) input between (A, B) is the public-key (p, q, g_1, g_2, t, v) , and the private (secret) input of A is the secret-key (s_1, s_2) .

[**Output**] B 's decision (accept or reject).

The protocol (A, B) is as follows:

1. A picks random numbers $r_1, r_2 \in \mathbf{Z}_q$, and computes

$$x = g_1^{r_1} g_2^{r_2} \bmod p,$$

and sends x to B .

2. B sends a random number $e \in \mathbf{Z}_{2^t}$ to A .
3. A sends to B (y_1, y_2) such that

$$y_1 = r_1 + es_1 \bmod q, \quad \text{and} \quad y_2 = r_2 + es_2 \bmod q.$$

4. B checks that

$$x = g_1^{y_1} g_2^{y_2} v^e \bmod p.$$

If it holds, B accepts, otherwise rejects.

2.3 ID-RSA

2.3.1 Key Generation: \mathcal{G}

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

[**Input**] Security parameter l , which is a positive integer.

[**Output**] A pair of public-key, (a, k, n, v) , and secret-key, (s_1, s_2) .

The operation of \mathcal{G} , on input 1^l , is as follows:

- Select primes p, q, k , and compute $n = pq$ such that $\gcd(k, \phi(n)) = 1$, $|k| = O(l)$, and $|n| = l$ where $\phi(n) = \text{lcm}(p-1, q-1)$. (e.g., $k \geq 2^{20}$, $n \geq 2^{1024}$) Here, p, q can be discarded after publishing n .
- Select random number $s_1 \in \mathbf{Z}_k$, and random numbers $a, s_2 \in \mathbf{Z}_n^*$, and compute $v = a^{-s_1} s_2^{-k} \bmod n$.

Remark: (a, k) can be common among users as the system parameter.

2.3.2 Interactive Protocol: (A, B)

The input and output of (A, B) are as follows:

[**Input**] The common (shared) input between (A, B) is the public-key (a, k, n, v) , and the private (secret) input of A is the secret-key (s_1, s_2) .

[**Output**] B 's decision (accept or reject).

The protocol (A, B) is as follows:

1. A picks random numbers $r_1 \in \mathbf{Z}_k$ and $r_2 \in \mathbf{Z}_n^*$, and computes

$$x = a^{r_1} r_2^k \bmod n,$$

and sends x to B .

2. B sends a random number $e \in \mathbf{Z}_k$ to A .
3. A sends to B (y_1, y_2) such that

$$y_1 = r_1 + es_1 \bmod k, \quad y_2 = a^{\lfloor (r_1 + es_1)/k \rfloor} r_2 s_2^e \bmod n.$$

4. B checks that $x = a^{y_1} y_2^k v^e \bmod n$. If it holds, B accepts, otherwise rejects.

2.4 ID-ECDL: Elliptic curve version of ID-DL

Recently, many researchers have examined elliptic curve cryptosystems, which were firstly proposed by Miller and Kobitz [19, 16]. The elliptic curve cryptosystems which are based on the elliptic curve logarithm over a finite field have some advantages than other systems: the key size can be much smaller than the other schemes since only exponential-time attacks have been known so far if the curve (with trace $t \notin \{0, 1, 2\}$) is carefully chosen [17, 21, 26, 29, 31], and the elliptic curve discrete logarithms might be still intractable even if factoring and the multiplicative group discrete logarithm are broken.

The techniques to construct cryptosystems based on the elliptic curve logarithm over a finite field [16, 17, 19] can be straightforwardly applied to our Identification scheme, ID-DL.

The elliptic curve version of ID-DL (ID-ECDL) has the significant property that ID-ECDL is proven to be as secure as the intractability of the elliptic curve discrete logarithms.

3 Attributes and Advantages of ID-DL, ID-ECDL and ID-RSA

The advantages of the proposed identification schemes, ID-DL and ID-RSA, are their efficiency and provable security. As for efficiency, ID-DL and ID-RSA are almost as efficient as the Schnorr and Guillou-Quisquater (GQ) schemes, respectively. As for security, ID-DL and ID-RSA are as secure as the discrete logarithm and RSA problems, respectively.

Here we compare the properties of ID-DL, ID-ECDL, ID-RSA, Schnorr, Guillou-Quisquater (GQ), and Feige-Fiat-Shamir (FFS).

Table 1: Comparison of Identification Schemes

	<i>ID-DL</i>	<i>Schnorr</i>	<i>ID-ECDL</i>	<i>ID-RSA</i>	<i>GQ</i>	<i>FFS</i>
Provably secure?	Yes	No	Yes	Yes	No	Yes
Primitive problem	DL	DL	ECDL	RSA	RSA	Fact.
System parameter size (bits)	3232	2208	640	1044	20	0
Public key size (bits)	1024	1024	160	2048	2048	20480
Secret key size (bits)	320	160	320	1044	1024	20480
Communication amount (bits)	1364	1204	500	2088	2068	2068
Computation amount (Prover) (# of 1024-bit modular multiplications)	280	240	34	67	61	11
Computation amount (Verifier) (# of 1024-bit modular multiplications)	280	240	34	38	35	11

We assume that moduli p and q for ID-DL and Schnorr are 1024 bits and 160 bits respectively, and modulus n for ID-RSA, Guillou-Quisquater (GQ), and Feige-Fiat-Shamir (FFS) is 1024 bits. The size of the finite field and q for ID-ECDL are 160 bits. The challenge from the verifier is assumed to be 20 bits.

Here, we estimate the performance of unsophisticated implementations, since the purpose of this comparison is to relatively compare some schemes with the same primitive problem (e.g., our scheme 1 and Schnorr), and many sophisticated techniques (e.g., [20, 2]) can be fairly evenly applied to the schemes with the same primitive problem. We assume the standard binary method and the extended binary method (4.6.3 ex.27 in [18]) for the modular exponentiation. In addition, we assume that the group addition on the elliptic curve costs 5 times as much as the modular multiplication of the finite field does.

4 Security Assessment of ID-DL, ID-ECDL and ID-RSA

This section shows our security assessment of ID-DL, ID-ECDL and ID-RSA (See Appendix for the proofs of these theorems).

Definition 4.1 *The discrete logarithm is (non-uniformly) intractable, if any family of boolean circuits, which, given properly chosen (g_1, g_2, p, q) in the same distribution as the output of key generator G , can compute the discrete logarithm $\alpha \in \mathbf{Z}_q$ ($g_2 = g_1^\alpha \pmod p$) with non-negligible probability, must grow at a rate faster than any polynomial in the size of the input, $|p|$.*

Definition 4.2 *RSA inversion is (non-uniformly) intractable, if any family of boolean circuits, which, given properly chosen (a, k, n) in the same distribution as the output of key generator G , can compute $a^{1/k} \pmod n$ with non-negligible probability, must grow at a rate faster than any polynomial in the size of the input, $|n|$.*

Theorem 4.3 *Identification scheme ID-DL is secure if and only if the discrete logarithm is intractable.*

Corollary 4.4 *Identification scheme ID-ECDL is secure if and only if the elliptic curve discrete logarithm is intractable.*

Theorem 4.5 *Identification scheme ID-RSA is secure if and only if the RSA inversion problem is intractable.*

5 Limitations

There is no specific limitation in the proposed identification schemes.

6 Intellectual Property Statement

NTT has filed patent applications only in Japan on the techniques used in this contribution. NTT will license any resulting patent in a reasonable and non-discriminatory fashion.

References

- [1] T. Beth, "Efficient Zero-Knowledge Identification Scheme for Smart Cards," Proceedings of Eurocrypt'88, LNCS 330, Springer-Verlag, pp.77-86 (1988).
- [2] E.F. Brickell, D.M. Gordon, K.S. McCurley, and D.Wilson, "Fast Exponentiation with Precomputation", to appear in the Proceedings of Eurocrypt'92.
- [3] E.F.Brickell, and K.S.McCurley, "An Interactive Identification Scheme Based on Discrete Logarithms and Factoring," Journal of Cryptology, Vol.5, No.1, pp.29-39 (1992).
- [4] E.F.Brickell, and K.S.McCurley, "Interactive Identification and Digital Signatures," AT&T Technical Journal, pp.73-86, November/December (1991).
- [5] M.Bellare, S.Micali and R.Ostrovsky, "Perfect Zero-Knowledge in Constant Rounds," Proceedings of STOC, pp.482-493 (1990).
- [6] M.Bellare, S.Micali and R.Ostrovsky, "The (True) Complexity of Statistical Zero-Knowledge," Proceedings of STOC, pp.494-502 (1990).
- [7] D.Chaum, "Security without Identification: Transaction Systems to Make Big Brother Obsolete," Comm. of the ACM, 28, 10, pp.1030-1044 (1985).
- [8] L.Chen, I.Damgård, "Security Bounds for Parallel Versions of Identification Protocols," Manuscript (1992).
- [9] U.Feige and A.Shamir, "Witness Indistinguishable and Witness Hiding Protocols," Proceedings of STOC, pp.416-426 (1990).

- [10] U.Feige and A.Shamir, “Zero Knowledge Proofs of Knowledge in Two Rounds,” Proceedings of Crypto’89, LNCS 435, Springer-Verlag, pp.526-544 (1990).
- [11] U.Feige, A.Fiat and A.Shamir, “Zero Knowledge Proofs of Identity,” Proceedings of STOC, pp.210-217 (1987).
- [12] A.Fiat and A.Shamir, “How to Prove Yourself: Practical Solutions to Identification and Signature Problems”, Proceedings of CRYPTO ’86, LNCS 263, Springer-Verlag, pp.186–194 (1987).
- [13] O.Goldreich and H.Krawczyk “On the Composition of Zero-Knowledge Proof Systems,” Proceedings of ICALP, LNCS 443, Springer-Verlag, pp.268-282 (1990).
- [14] S.Goldwasser, S.Micali and C.Rackoff, “The Knowledge Complexity of Interactive Proofs,” SIAM J. Comput., 18, 1, pp.186-208 (1989).
- [15] L.S.Guillou, and J.J.Quisquater, “A Practical Zero-Knowledge Protocol Fitted to Security Microprocessors Minimizing both Transmission and Memory,” Proceedings of Eurocrypt ’88, LNCS 330, Springer-Verlag, pp.123-128 (1988).
- [16] N.Koblitz, *A Course in Number Theory and Cryptography*, Berlin: Springer-Verlag, (1987).
- [17] N.Koblitz, “CM-Curves with Good Cryptographic Properties,” Proceedings of Crypto ’91 (1992).
- [18] D.E.Knuth, *The Art of Computer Programming*, Vol.2, 2nd Ed. Addison-Wesley (1981).
- [19] V.Miller, “Uses of Elliptic Curves in Cryptography,” Proceedings of Crypto ’85, LNCS 218, Springer-Verlag, pp.417-426 (1986).
- [20] P.L.Montgomery, “Modular Multiplication without Trial Division,” Math. of Computation, Vol.44, pp.519–521 (1985).
- [21] A.J.Menezes, T.Okamoto, S.A.Vanstone, “Reducing Elliptic Curve Logarithms to Logarithms in a Finite Field”, Proceedings of STOC, pp.80-89 (1991).
- [22] K.Ohta, and T.Okamoto, “A Modification of the Fiat-Shamir Scheme,” Proceedings of Crypto ’88, LNCS 403, Springer-Verlag, pp.232-243 (1990).
- [23] T.Okamoto, “Provably Secure and Practical Identification Schemes and Corresponding Signature Schemes,” Proceedings of Crypto’92, LNCS 740, Springer-Verlag, pp.31-53 (1993).
- [24] S.C.Pohlig, and M.E.Hellman, “An Improved Algorithm for Computing Logarithms over $GF(p)$ and Its Cryptographic Significance,” IEEE Trans. Inform. Theory, 24, pp.106–110 (1978)
- [25] R.Rivest, A.Shamir and L.Adleman, “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”, Communications of the ACM, Vol.21, No.2, pp.120-126 (1978).

- [26] Satoh, T. and Araki, K.: Fermat Quotient and the Polynomial Time Discrete Log Algorithm for Anomalous Elliptic Curves, Preprint (September, 1997).
- [27] C.P. Schnorr, "Efficient Signature Generation by Smart Cards," Journal of Cryptology, Vol.4, No.3, pp.161-174 (1991).
- [28] A.Shamir, "Identity-Based Cryptosystems and Signature Scheme," Proceedings of Crypto '84, LNCS 196, Springer-Verlag, pp.47-53 (1986).
- [29] Semaev, I.A.: Evaluation of Discrete Logarithms on Some Elliptic Curves, to appear in Mathematics of Computation.
- [30] K.Sakurai, and T.Itoh, "On the Discrepancy between Serial and Parallel of Zero-Knowledge Protocols," These proceedings.
- [31] Smart, N.P.: The Discrete Logarithm Problem on Elliptic Curves of Trace One, Preprint (September, 1997).

Appendix: Proofs of Theorems

1 Definition of Secure Identification

1.1 Identification

Definition 1.1 *An identification scheme consists of two stages:*

1. *Initialization: In this stage, each user (e.g., A) generates a secret key (e.g., SK_A) and a public key (e.g., PK_A) by using probabilistic polynomial-time generation algorithm G on input of the key size. A link between each user and its public key is established. Note that in some schemes a part of the public key can be commonly shared among all users as a system parameter.*
2. *Operation: In this stage, any user (e.g., A) can demonstrate its identity to a verifier by performing some identification protocol related to its public key (e.g., PK_A), where the input for the verifier is the public key (e.g., PK_A). At the conclusion of this stage, the verifier either outputs “accept” or “reject”.*

1.2 Security of Identification schemes

We define a *secure* identification scheme based on the definition (the “no useful information transfer”) given by Feige et. al. [11].

Definition 1.2 *A prover A (resp. verifier B) is a “good” prover denoted by \overline{A} (resp. “good” verifier denoted by \overline{B}), if it does not deviate from the protocols dictated by the scheme. Let \tilde{A} be a fraudulent prover who does not complete the Initialization stage of Definition 1.1 as A and may deviate from the protocols (so another person/machine can simulate \tilde{A}). \tilde{B} is not a good B . \tilde{A} and \tilde{B} are assumed to be polynomial time bounded machines, which may be nonuniform.*

An identification scheme (A, B) is secure if

1. *$(\overline{A}, \overline{B})$ succeeds with overwhelming probability.*
2. *There is no coalition of \tilde{A}, \tilde{B} with the property that, after a polynomial number of executions of $(\overline{A}, \overline{B})$ and relaying a transcript of the communication to \tilde{A} , it is possible to execute (\tilde{A}, \tilde{B}) with non-negligible probability of success. The probability is taken over the distribution of the public key and the secret key as well as the coin tosses of $\overline{A}, \overline{B}, \tilde{A}$, and \tilde{B} , up to the time of the attempted impersonation.*

Remark: When an identification scheme is “witness hiding” [9] and an interactive proof of “knowledge” [11], this scheme is secure in the sense of Definition 1.2. This is roughly because if there exists (\tilde{A}, \tilde{B}) with non-negligible probability of success, we can construct a knowledge extractor (from the “knowledge soundness”), which leads to contradiction with “witness hiding”. Thus there are two ways to prove the security of Definition 1.2: One is to prove it directly as in [11, 22], and the other way is to prove that a scheme is “witness hiding” and an interactive proof of “knowledge”. Some schemes such as [22] seem to be proven only in the former way, since the

knowledge soundness is sometimes hard to prove (e.g., [22]). Here we will prove our schemes in the former way, since it is compatible with the way to prove it by a variant of Definition 1.2, [22], to be described below, although we can also use the latter approach.

2 Lemma

Here, we show a definition and lemma in preparation for the proofs.

Definition 2.1 *Let RA denote \tilde{A} 's random tape, and RB denote \overline{B} 's random tape. The possible outcomes of executing $(\tilde{A}, \overline{B})$ can be summarized as a large Boolean matrix H whose rows correspond to all possible choices of RA . Its columns correspond to all possible choices e of RB , and its entries are 1 if \overline{B} accepts \tilde{A} 's proof, and 0 if otherwise.*

When the success probability of \tilde{A} is ε (or the rate of 1-entries in H is ε), we call a row heavy if its ratio of 1's is at least $\varepsilon/2$.

Lemma 2.2 *If, given A 's public key (p, q, g_1, g_2, t, v) , the success probability, ε , of \tilde{A} is greater than 2^{-t+1} , then there exists a probabilistic algorithm which runs in expected time $O(\|\tilde{A}\|/\varepsilon)$ and outputs the history of two accepted executions of $(\tilde{A}, \overline{B})$, (x, e, y_1, y_2) and (x, e', y'_1, y'_2) , where $e \neq e'$. Here, $\|\tilde{A}\|$ denotes the time complexity of \tilde{A} . The success probability ε is taken over the coin tosses of \tilde{A} and \overline{B} .*

Proof:

Assume that at least $1/2$ of the 1's in H are not located in heavy rows. Then the fraction of non-heavy rows in H , which we denote τ , is estimated as follows: $\tau \geq \frac{2^t \varepsilon / 2}{2^t \varepsilon / 2 - 1} > 1$. This is a contradiction. Therefore, at least $1/2$ of the 1's in H are located in heavy rows. Since ε is greater than 2^{-t+1} and the width of H is 2^t , a heavy row contains at least two 1's. To find two 1's in the same row, we thus adopt the following strategy:

1. Probe $O(1/\varepsilon)$ random entries in H (or pick (RA, e) randomly and check it, and repeat this until successful).
2. After the first 1 is found (or accepted (x, e, y_1, y_2) with RA is found), probe $O(1/\varepsilon)$ random entries along the same row (or probe (x, e', y'_1, y'_2) with the same RA).

Since at least $1/2$ of the 1's in H are located in heavy rows, this strategy succeeds with constant probability in $O(1/\varepsilon)$ probes. ¶

3 Proof of Theorem 4.3 for ID-DL

Theorem 4.3

Identification scheme ID-DL is secure if and only if the discrete logarithm is intractable.

Proof:

(Only if:)

Suppose that the discrete logarithm is not intractable. Clearly a (nonuniform) polynomial time machine can calculate (s'_1, s'_2) satisfying $v = g_1^{-s'_1} g_2^{-s'_2} \pmod p$ with non-negligible probability. Thus Identification scheme 1 is not secure.

(If:)

To prove the ‘‘If’’ part, we show that if Identification scheme 1 is not secure, then, given (g_1, g_2, p, q) with the same distribution as the output of key generator G , the discrete logarithm $\alpha \in \mathbf{Z}_q$ ($g_2 = g_1^\alpha \pmod p$) can be computed by a polynomial time machine P with non-negligible probability.

Assume that Identification scheme 1 is not secure. Then (\tilde{A}, \tilde{B}) can be accepted with non-negligible probability ε after $O(|p|^c)$ executions of (\tilde{A}, \tilde{B}) . The complete history of the executions of (\tilde{A}, \tilde{B}) and (\tilde{A}, \tilde{B}) can be simulated by one polynomial time procedure P , which may be nonuniform, if P knows \tilde{A} 's secret key.

To calculate the discrete logarithm $\alpha \in \mathbf{Z}_q$ ($g_2 = g_1^\alpha \pmod p$), given (g_1, g_2, p, q) , P firstly chooses $s_1^*, s_2^* \in \mathbf{Z}_q$ randomly, and calculates $v = g_1^{-s_1^*} g_2^{-s_2^*} \pmod p$.

Then, using (s_1^*, s_2^*) as \tilde{A} 's secret key, P simulates (\tilde{A}, \tilde{B}) as well as (\tilde{A}, \tilde{B}) . So, for (v, g_1, g_2, p, q) , after simulating $O(|p|^c)$ executions of (\tilde{A}, \tilde{B}) , P tries to find two accepted interactions of (\tilde{A}, \tilde{B}) , (x, e, y_1, y_2) and (x, e', y'_1, y'_2) ($e \neq e'$). From Lemma 2.2, this is possible with overwhelming probability, since ε is non-negligible i.e. greater than 2^{-t+1} .

P can then calculate $(s_1, s_2) = ((y_1 - y'_1)/(e - e') \pmod q, (y_2 - y'_2)/(e - e') \pmod q)$ by

$$\begin{aligned} y_1 &= r_1 + es_1 \pmod q, & y_2 &= r_2 + es_2 \pmod q, \\ y'_1 &= r_1 + e's_1 \pmod q, & y'_2 &= r_2 + e's_2 \pmod q. \end{aligned}$$

There are q solutions of (s_1, s_2) that satisfy $v = g_1^{-s_1} g_2^{-s_2} \pmod p$, given (v, g_1, g_2, p, q) . Even an infinitely powerful \tilde{B} cannot determine from x 's, y_1 's, and y_2 's sent by \tilde{A} during the execution of (\tilde{A}, \tilde{B}) which (s_1, s_2) satisfying $v = g_1^{-s_1} g_2^{-s_2} \pmod p$ is actually used. To prove this, for two different solutions, (s_1, s_2) and (s_1^*, s_2^*) satisfying $v = g_1^{-s_1} g_2^{-s_2} \equiv g_1^{-s_1^*} g_2^{-s_2^*} \pmod p$, we show that even an infinitely powerful \tilde{B} cannot determine which solution was used from x 's, y_1 's, and y_2 's. When $r_1^* = r_1 + e(s_1 - s_1^*) \pmod q$ and $r_2^* = r_2 + e(s_2 - s_2^*) \pmod q$, the following three equations hold.

$$\begin{aligned} x &= g_1^{r_1} g_2^{r_2} \equiv g_1^{r_1^*} g_2^{r_2^*} \pmod p, \\ y_1 &= r_1 + es_1 \equiv r_1^* + es_1^* \pmod q, \\ y_2 &= r_2 + es_2 \equiv r_2^* + es_2^* \pmod q. \end{aligned}$$

In addition, the distributions of (r_1, r_2) and (r_1^*, r_2^*) are exactly equivalent even if they satisfy the above relation. Hence, although P knows (s_1^*, s_2^*) , (s_1, s_2) , which is calculated by P by simulating the operations of (\tilde{A}, \tilde{B}) and (\tilde{A}, \tilde{B}) , is independent from (s_1^*, s_2^*) .

Therefore, (s_1^*, s_2^*) which was randomly chosen by P at first is different with probability $(q - 1)/q$ from (s_1, s_2) . Thus, α can be calculated with probability $(q - 1)/q$ from (s_1, s_2) and (s_1^*, s_2^*) such that $\alpha = (s_1 - s_1^*)/(s_2^* - s_2) \pmod q$. The total success probability of P is non-negligible.

This contradicts the intractability assumption of the discrete logarithm. ¶

4 Proof of Theorem 4.5 for ID-RSA

Theorem 4.5 Identification scheme 2 is secure if and only if RSA inversion is intractable.

Proof:

(Only if:)

Suppose that the RSA inversion is not intractable. Clearly a (nonuniform) polynomial time machine can calculate (s'_1, s'_2) satisfying $v = a^{-s'_1} s'^{-k}_2 \pmod n$ with non-negligible probability. Thus Identification scheme 2 is not secure.

(If:)

To prove the “If” part, we can prove this in a manner similar to the “if” part proof of Theorem 4.3. So we only sketch the different points here.

First, P chooses $s_1^* \in \mathbf{Z}_k$, and $s_2^* \in \mathbf{Z}_n^*$ randomly, and calculates $v = a^{-s_1^*} s_2^{*-k} \pmod n$.

Then, for (a, k, n, v) , P finds (x, e, y_1, y_2) and (x, e', y'_1, y'_2) ($e \neq e'$) by the technique of Lemma 2.2.

Next, P calculates $s_1 = (y_1 - y'_1)/(e - e') \pmod k$, and $r_1 = y_1 - es_1 \pmod k$. P then calculates X, Y as follows:

$$X = \frac{y_2/a^{\lfloor (r_1 + es_1)/k \rfloor}}{y'_2/a^{\lfloor (r_1 + e's_1)/k \rfloor}} \pmod n (= s_2^{e-e'} \pmod n),$$

$$Y = 1/(va^{s_1}) \pmod n (= s_2^k \pmod n).$$

Since $\gcd(k, e - e') = 1$ (as k is prime), P can compute α, β satisfying $\alpha(e - e') + \beta k = 1$ by the extended Euclidean algorithm. Hence P calculates $s_2 = X^\alpha Y^\beta \pmod n$.

There are k solutions of (s_1, s_2) that satisfy $v = a^{-s_1} s_2^{-k} \pmod n$, given (v, n, a, k) . Even an infinitely powerful \tilde{B} cannot determine from x 's, y_1 's, and y_2 's which (s_1, s_2) was actually used.

P then obtains $(s_1, s_2), (s_1^*, s_2^*)$ ($s_i \neq s_i^*$) such that $v = a^{s_1} s_2^k \equiv a^{s_1^*} s_2^{*k} \pmod n$, so $a^{(1/k)(s_1 - s_1^*)} \equiv s_2^*/s_2 \pmod n$. After repeating the above procedure, P obtains another $(s'_1, s'_2), (s'^*_1, s'^*_2)$ ($s'_i \neq s'^*_i$) such that $a^{(1/k)(s'_1 - s'^*_1)} \equiv s'^*_2/s'_2 \pmod n$ with non-negligible probability. If $\gcd(s_1 - s_1^*, s'_1 - s'^*_1) = 1$, then P can calculate $a^{1/k} \pmod n$. The probability that $\gcd(s_1 - s_1^*, s'_1 - s'^*_1) = 1$ is more than a constant, since s_1^*, s'^*_1 is selected randomly and s_1, s'_1 is independent from s_1^*, s'^*_1 . Thus, the total success probability of P is non-negligible.

This contradicts the intractability assumption of RSA inversion. ¶