Tatra Mt. Math. Publ. **29** (2004), 147–155

TATRA MOUNTAINS
*Mathematical Publications*

# CRYPTOGRAPHIC PRIMITIVES BASED ON GROUPS OF HIDDEN ORDER

Wandi Wei — Tran van Trung —
Spyros S. Magliveras — Frederick Hoffman

ABSTRACT. Many of the known attacks against a number of cryptographic primitives are based on knowing the order of the underlying group. We propose a new idea that involves hiding the order of the group. As a first example, we examine the feasibility of combining the intractability of the integer factorization and discrete logarithm problems to extend the ElGamal cryptosystem to the case where the order of the group is hidden.

## 1. Introduction

There are two enduring problems on which public key cryptography has relied over the past two decades: i) The difficulty of *factorization* of large integers, and ii) the difficulty of solving the *Discrete Logarithm Problem* (DLP). All known efficient cryptanalytic attacks on the DLP require knowledge of the order of the underlying group $G$. On the other hand, a system like RSA, based on i), can be viewed as one in which the order of the underlying group is hidden.

We note that the DLP is not an intrinsically intractable problem. Rather, it depends on the particular representation of the underlying cyclic group. For example, the problem is trivial in the additive cyclic group $\mathbb{Z}_n$ under addition modulo $n$. If $\alpha \in \mathbb{Z}_n$ is a generator, then $\gcd(\alpha, n) = 1$, and the problem becomes:

$$\alpha x = \beta\,,$$

which is clearly solvable by finding $\alpha^{-1}$ in the ring $\mathbb{Z}_n$ by means of the Euclidean algorithm. In general the DLP is believed to be difficult in the cyclic multiplicative group $\mathbb{F}_q^*$ of an appropriate finite field $\mathbb{F}_q$, and for a cyclic component of an appropriately chosen elliptic curve $\mathcal{E}$ over a finite field $\mathbb{F}_q$. Thus, we see that the "intractability" of DLP relates to the particular representation

2000 Mathematics Subject Classification: 94A60.
Keywords: public key cryptography, discrete logarithm, factorization, RSA cryptosystem, ElGamal cryptosystem, group, hidden-order.

of the cyclic group in question. If $n = |G|$, Shank's[1] and Pollard's algorithms [5, 6] solve the DLP in time $O(\sqrt{n})$. Moreover, if $n$ has relatively small factors, the Silver-Pohlig-Hellman algorithm provides a significant reduction to the time complexity. The *index calculus* approach for solving the DLP in $\mathbb{F}_q^*$ may be viewed as a time-space trade-off and can be extremely time-efficient at the expense of space needed to store the accumulated linear equations.

On the other hand, what could be said of a situation in which the attacker of an instance of the ElGamal cryptosystem did not know the order of the underlying group? How could such a system be designed? And how can the notion of *discrete logarithm* be extended to arbitrary, non-abelian groups? Can practical proposals be made which go beyond cyclic groups represented in the familiar ways? An even more ambitious question would ask how can we build secure and practical systems based on group-theoretic methods, which rely on the generally undecidable *word problem.* We believe that the high road to building such systems would be to concentrate on combinatorial group-theoretic methods. A system like the ones proposed in [2, 7], and analyzed in [3] are perhaps models on which to build. In the past, not much effort has been expanded on the substance of the above questions.

In this paper we present an example that demonstrates the efficacy of hiding the group order.

### 1.1 The ElGamal scheme

In what follows we remind the reader of the ElGamal public key cryptosystem. The message space is a large cyclic group $G$ in which DLP is hard. Let $\alpha$ be a generator known by all communicants. Alice chooses a random integer $k$ which she keeps secret, and publishes $\beta = \alpha^k$. If Bob wants to send Alice message $x$, he chooses a random integer $R$, and sends Alice the pair $(y_1, y_2) = (\alpha^R, \beta^R x)$. Now, Alice can compute $\beta^R$ by:

$$z = \beta^R = \left(\alpha^k\right)^R = \left(\alpha^R\right)^k = y_1^k\,.$$

Hence, Alice can recover the message $x$ by:

$$x = z^{-1}(zx) = z^{-1}y_2\,.$$

A third party who intercepts $(y_1, y_2)$ and can solve the DLP can also recover $x$, either by finding the secret key $k$, from $\beta = \alpha^k$, or by finding $R$ from $y_1 = \alpha^R$. Clearly, Alice as well as the interceptor needs to have access to an efficient algorithm for computing $z^{-1}$ in $G = \langle\alpha\rangle$. Of course, if $|G|$ is known, $z^{-1}$ can be computed efficiently by computing $z^{|G|-1}$ if not by a more efficient method.

[1] also known as the *Baby step–Giant step* algorithm

## 2. A public-key cryptosystem of ElGamal type

We propose a public-key cryptosystem which uses computations in $\mathbb{Z}_n$, where $n$ is the product of two distinct large primes $p$ and $q$. This cryptosystem is close to the ElGamal cryptosystem. We prove, however, that our cryptosystem is at least as strong as each of the ElGamal and RSA cryptosystems. Our proof shows an interesting fact that it is possible to compare the strength of a cryptosystem of ElGamal type with RSA using the same modulus. The original ElGamal cryptosystem does not allow such a comparison.

### 2.1 Description of the cryptosystem

Let $n = p_1 p_2$ be the product of two large distinct primes $p_1$ and $p_2$ such that the discrete logarithm problem is intractable in each $\mathbb{Z}_{p_i}^*$. For better security we may select $p_i$ to be of the form $p_i = 2q_i + 1$ where $q_i$ are (Sophie Germain) primes. Let $\alpha_i$ be a primitive element in $\mathbb{Z}_{p_i}$, and $a_i$ an element of $\mathbb{Z}_{p_i}^*$ with the property that

$$\gcd(p_1 - 1, p_2 - 1) \mid (a_1 - a_2). \tag{2.1}$$

Note that this condition permits the choice $a_1 = a_2$. We are thankful to the anonymous referee who pointed out that no particular weakness appears to result from this choice.

In what follows repeated use of the *Chinese Remainder Theorem* (CRT) will be made, without additional reference, to obtain solutions of certain systems of congruence equations.

Let

$$\beta_i = \alpha_i^{a_i} \pmod{p_i}, \qquad \text{for } i = 1, 2.$$

Under condition (2.1), the congruence equations

$$a \equiv a_1 \pmod{p_1 - 1},$$
$$a \equiv a_2 \pmod{p_2 - 1},$$

have a unique solution $a$ modulo $t$, where

$$t = (p_1 - 1)(p_2 - 1)/\gcd(p_1 - 1, p_2 - 1) = \phi(n)/\gcd(p_1 - 1, p_2 - 1).$$

Since $p_1$ and $p_2$ are distinct primes, the system of congruence equations

$$\alpha \equiv \alpha_1 \pmod{p_1},$$
$$\alpha \equiv \alpha_2 \pmod{p_2},$$

has a unique solution $\alpha$ modulo $n$. Note that $\alpha$ generates a subgroup of order $t$ in the multiplicative group of units $\mathbb{Z}_n^*$ of the ring $\mathbb{Z}_n$.

Also, the system of congruences

$$\beta \equiv \beta_1 \pmod{p_1},$$
$$\beta \equiv \beta_2 \pmod{p_2},$$

has a unique solution $\beta$ modulo $n$.

Let

$$K_{\text{pub}} = \{n, \alpha, \beta\},$$
$$K_{\text{pri}} = \{p_1, p_2, a\},$$

be the public key and private key of the scheme, respectively, and assign

$$K = (K_{\text{pub}}, K_{\text{pri}}).$$

To *encrypt* a message $x \in \mathbb{Z}_n$, compute the ciphertext $e_{K_{\text{pub}}}(x) = (y_1, y_2)$ as follows: Choose a (secret) random number $R \in \mathbb{Z}_n$ and compute

$$y_1 = \alpha^R \pmod{n},$$
$$y_2 = x\beta^R \pmod{n}.$$

To *decrypt* the ciphertext $(y_1, y_2)$, compute

$$d_K(y_1, y_2) = y_2(y_1^a)^{-1} \pmod{n}. \tag{2.2}$$

If the $\alpha_i$ were chosen not to necessarily be primitive elements of $\mathbb{Z}_{p_i}$, but of rather large order in $\mathbb{Z}_{p_i}^*$, a slightly more general system can be constructed in which $\alpha$ is not necessarily of largest possible order. We denote this more general system also by $(n, \alpha, \beta)$.

## 2.2 Proof of correctness

Since

$$a \equiv a_i \pmod{p_i - 1} \qquad \text{for } i = 1, 2,$$

we have that:

$$\alpha_i^a \equiv \alpha_i^{a_i} \pmod{p_i}.$$

Therefore, computing modulo $p_i$ yields:

$$\begin{aligned} y_2(y_1^a)^{-1} &\equiv (x\beta^k)\big(\alpha^{ka_i}\big)^{-1} \\ &\equiv (x\beta_i^k)\big(\alpha_i^{ka_i}\big)^{-1} \\ &\equiv \big(x(\alpha_i^{a_i})^k\big)\big(\alpha_i^{ka_i}\big)^{-1} \\ &\equiv x\big(\alpha_i^{ka_i}\big)\big(\alpha_i^{ka_i}\big)^{-1} \\ &\equiv x \pmod{p_i}, \end{aligned}$$

for $i = 1, 2$.

Since $p_1$ and $p_2$ are distinct primes, we conclude that:

$$y_2(y_1^a)^{-1} \equiv x \pmod{n}.$$

## 3. Security of the cryptosystem

Under the assumption that the integer factorization problem and the discrete logarithm problem are computationally intractable, we show that our public-key cryptosystem is at least as secure as each of the RSA and ElGamal public-key cryptosystems.

**Remark 3.1.** Suppose that an RSA public key instance $(n, b)$ is observed by a potential attacker $B$. Now, $B$ can establish what we may call an *associated* public key $(n, \alpha, \beta)$ for our cryptosystem as follows. $B$ randomly selects a positive integer $\beta$ which is relatively prime to $n$ and computes $\alpha = \beta^b \bmod n$. $B$ now verifies that $\alpha$ is not of small order[2]. It follows that $\beta = \alpha^a \bmod n$ where $a$ is the private key in the RSA instance $(n, b)$, i.e., where $ab \equiv 1 \bmod \phi(n)$. Now, $B$ can treat $(n, \alpha, \beta)$ as the public key of an instance $S$ of our (more general) cryptosystem. In what follows, we will assume that an oracle $\mathcal{O}$ exists that breaks this (more general) instance of our cryptosystem. That is, given $(n, \alpha, \beta)$ and a ciphertext $y$, the oracle returns the correct plaintext $x$ for $y$ in the system $S$ with public key $(n, \alpha, \beta)$. We proceed to prove the following theorem.

**THEOREM 3.1.** *Our public-key cryptosystem is stronger than each of the* RSA *and the ElGamal public-key cryptosystems in the sense that if there is an oracle that can break our system, then it can also break each of the* RSA *system and the ElGamal system.*

Proof. The theorem is an immediate consequence of the following two lemmas. □

**LEMMA 3.1.** *Suppose $\mathcal{O}$ is an oracle that can break our public-key cryptosystem. Then $\mathcal{O}$ can break the* RSA *public-key cryptosystem.*

Proof. An attacker $B$ can employ $\mathcal{O}$ to break the RSA public key cryptosystem as follows: Let $(n, b)$ be the public-key of the particular instance of RSA, and $y$ the ciphertext corresponding to some plaintext $x$ which $B$ seeks to recover.

By the remark just before Theorem 3.1 $B$ constructs an instance of our cryptosystem with public key $(n, \alpha, \beta)$ without knowledge of the private key $(p_1, p_2, a)$, of the RSA cryptosystem instance.

Then, $B$ sets $y_1 = y$, chooses an arbitrary $y_2$, and considers $(y_1, y_2)$ as the ciphertext of some plaintext under our system. Now, let $x$ be the plaintext

[2]If $p_1$ and $p_2$ arise from Sophie Germain primes $q_1$ and $q_2$, respectively, then $\phi(n) = 4q_1q_2$, $t = 2q_1q_2$ and there is an involution in the underlying group.

determined by querying the oracle $\mathcal{O}$ with $(y_1, y_2)$. Then $x$ must satisfy the equation

$$x = y_2 y_1^{-a} \pmod{n}.$$

Therefore,

$$y_1^a \equiv y_2 x^{-1} \pmod{n}.$$

Thus, querying $\mathcal{O}$ allows $B$ to compute the value $y_1^a \bmod n$, for any given $y_1$, even though $B$ does not know $a$. This in fact is nothing less than determining the corresponding plaintext $x$ of ciphertext $y$ under the RSA cryptosystem. □

**Lemma 3.2.** *Suppose $\mathcal{O}$ is an oracle that can break our public-key cryptosystem. Then $\mathcal{O}$ can break the ElGamal public-key cryptosystem.*

Proof. An attacker $B$ can employ $\mathcal{O}$ to break the (original) ElGamal public key cryptosystem as follows:

Let $(p_1, \alpha_1, \beta_1)$ be the public key of an instance $\mathcal{E}_1$ of the ElGamal cryptosystem and let $(y_1', y_2')$ be the ciphertext under $\mathcal{E}_1$ of some plaintext $x_1$. Let $a_1$ be the private key in $\mathcal{E}_1$, so that $\beta_1 = \alpha_1^{a_1} \pmod{p_1}$. Of course $a_1$ is not known to $B$. $B$ wishes to attack $\mathcal{E}_1$ and recover $x_1$.

$B$ proceeds to construct his own instance of the ElGamal system $\mathcal{E}_2$: $(p_2, \alpha_2, \beta_2)$ by arbitrarily choosing $p_2$ as an odd prime, $\alpha_2 \in \mathbb{Z}_{p_2}^*$, and $\beta_2 = \alpha_2^{a_2} \pmod{p_2}$ for an integer $a_2$ such that $1 \neq a_2 \in \mathbb{Z}_{p_2-1}^*$. For simplicity and easy computation $B$ chooses an appropriate small[3] prime $p_2$. Then, $B$ defines $n = p_1 p_2$, $\alpha$ as the unique solution modulo $n$ to the congruence equations:

$$\alpha \equiv \alpha_1 \pmod{p_1},$$
$$\alpha \equiv \alpha_2 \pmod{p_2},$$

and $\beta$ as the unique solution modulo $n$ to:

$$\beta \equiv \beta_1 \pmod{p_1},$$
$$\beta \equiv \beta_2 \pmod{p_2}.$$

Now, $B$ creates under $\mathcal{E}_2$ a plaintext-ciphertext pair $\big(x_2, (y_1'', y_2'')\big)$.

Let $y_1 \pmod{n}$ be the unique solution to the equations

$$y_1 \equiv y_1' \pmod{p_1},$$
$$y_1 \equiv y_1'' \pmod{p_2},$$

[3]In fact it is almost imperative to choose a small prime $p_2$ so that the resulting $n = p_1 p_2$ will be of about the same magnitude as $p_1$, affording a fair comparison between the original ElGamal and our cryptosystem.

and $y_2$ the unique solution to the congruence equations

$$\begin{aligned} y_2 &\equiv y_2' \pmod{p_1}, \\ y_2 &\equiv y_2'' \pmod{p_2}. \end{aligned}$$

Regarding $(y_1, y_2)$ as the ciphertext for some plaintext under our system with public key $(n, \alpha, \beta)$ $B$ employs the oracle $\mathcal{O}$ to recover the corresponding plaintext, say $x$. Then $x$ must satisfy

$$x = y_2 y_1^{-a} \pmod{n}, \tag{3.1}$$

where $a$ satisfies

$$\begin{aligned} a &\equiv a_1 \pmod{p_1 - 1}, \\ a &\equiv a_2 \pmod{p_2 - 1}. \end{aligned} \tag{3.2}$$

though the value of $a$ is unknown to $B$.

Let

$$x_1 = x \pmod{p_1}.$$

Then (3.1) gives

$$x_1 = y_2' y_1'^{\,-a_1} \pmod{p_1},$$

which means that $x_1$ is the plaintext corresponding to the ciphertext $(y_1', y_2')$ under the ElGamal cryptosystem $\mathcal{E}_1$. □

**Remark 3.2.** A reasonable (security) assumption for $\mathcal{E}_1$ is that 3 does not divide $p_1$. In the proof of Lemma 3.2, a good choice for the parameters of $\mathcal{E}_2$ would be $p_2 = 7$ and $\alpha_2 = 3$. Then, $\gcd(p_1 - 1, p_2 - 1) = 2$, so by Condition (2.1) $a_1$ and $a_2$ must have the same parity. If $a_1$ is odd, then $B$ may choose $a_2 = 5$ and $\beta_2 = \alpha_2^{a_2} = 3^5 \equiv 5 \pmod{7}$. Otherwise, if $a_1$ is even, $B$ could choose $a_2 = 2$ and $\beta_2 = 3^2 \equiv 2 \pmod{7}$. Condition must hold if the system of congruences (3.2) is to have a solution for $a$. Since the attacker has no knowledge of $a_1$ or its parity, he/she will need to make two attempts, the first with parameters $(a_2, \beta_2) = (5, 5)$ and the second with $(a_2, \beta_2) = (2, 2)$.

**Remark 3.3.** It is worth emphasizing that in our system the underlying group $G = \langle \alpha \rangle$, of hidden order $t$, is a subgroup of $\mathbb{Z}_n^*$. The system's security is based on the assumption that the factorization problem is intractable for a given composite integer $n$. On the other hand, as the system is ElGamal-like, we also assume the intractability of the DLP for a cyclic group of order $t$. Here, the integer $t$ can be chosen to be the product of two large primes such that $t$ is at least of 1024 bit long, while at the same time we choose the modulus $n$ to be of the same order of magnitude as $t$. By current estimates of the difficulty of the DLP and the factorization problem, such choices appear to be sufficient to maintain the integrity of the complete system.

## 4. Closing remarks

All known attacks on DLP assume knowledge of the underlying cyclic group. In this paper we propose an ElGamal-like public-key cryptosystem in which the order of the underlying cyclic group is hidden. We accomplish this by relying on the intractability of the integer factorization problem, fusing the RSA and original ElGamal systems. The underlying group of the new system is $\mathbb{Z}_n^*$, where $n$ is the product of two appropriate large primes. We show that the new system is at least as secure as each of RSA and the original ElGamal in the sense that if an oracle $\mathcal{O}$ can break our system it can also break RSA and the original ElGamal systems.

During our development a secondary but interesting fact emerges, namely that it is possible to compare the strength of a cryptosystem of ElGamal type with RSA using the same modulus. The original ElGamal cryptosystem does not allow such a comparison.

The main interest in this article is that it directs attention to the important, yet often overlooked, security measure of hiding the order of the underlying group. For future work, one may wish to seek public key scenarios based on combinatorial group theory and the undecidability of the word problem.

REFERENCES

[1] ELGAMAL, T.: *A public key cryptosystem and a signature scheme based on discrete logarithms*, IEEE Trans. Inform. Theory **31** (1985), 469–472.

[2] GARZON, M.—ZALCSTEIN, Y.: *The complexity of Grigorchuk groups with application to cryptography*, Theoret. Computer Science **88** (1991), 83–98.

[3] GONZÁLEZ VASCO, M. I.—HOFHEINZ, D.—MARTÍNEZ, C.—STEINWANDT, R.: *On the security of two cryptosystems using non-abelian groups*, Des. Codes Cryptogr. (to appear).

[4] HALLGREN, S.—RUSSELL, A.—TA-SHMA, A.: *The hidden subgroup problem and quantum computation using group representations*, SIAM J. Comput. **32** (2004), 916–934.

[5] MENEZES, A.—VAN OORSCHOT, P.—VANSTONE, S.: *Handbook of Applied Cryptography*, CRC Press, Singapore, 1996.

[6] STINSON, D. R.: *Cryptography Theory and Practice*, Chapman & Hall/CRC, 2nd Ed., Singapore, 2002.

[7] WAGNER, N. R.—MAGYARIK, M. R.: *A public key cryptosystem based on the word problem*, in: Advances in Cryptology. (G. R. Blakley and D. Chaum, ed.), Proceedings of CRYPTO '84, Lecture Notes in Comput. Sci., Vol. 196, Springer-Verlag, Berlin, 1985, pp. 19–36.

Received November 10, 2003

*Wandi Wei, Spyros S. Magliveras, Frederick Hoffman*
*Department of Mathematical Sciences*
*Florida Atlantic University*
*Boca Raton, Florida 33431-0991*
*U. S. A.*

*E-mail*: wei@brain.math.fau.edu
spyros@fau.edu
hoffman@fau.edu

*Tran van Trung*
*Institute for Experimental Mathematics*
*University of Duisburg-Essen*
*Ellernstrasse 29*
*45326 Essen*
*GERMANY*

*E-mail*: trung@exp-math.uni-essen.de