A Cryptosystem Based on DLP $\gamma \equiv \alpha^a \beta^b \bmod p$

Sunil Kumar Kashyap, Birendra Kumar Sharma, and Amitabh Banerjee
(Corresponding author: Birendra Kumar Sharma)

School of Studies in Mathematics, Pt.Ravishankar Shukla University, Raipur-492010, Chhattisgarh, India. (Email: sharmabk_nib@sancharnet.in)

(Received July 20, 2005; revised and accepted Aug. 23 & Sept. 15, 2005)

Abstract

This paper introduces mainly the concept of the Public Key Cryptosystem or PKC, whose security is based on the new Discrete Logarithm Problem known as DLP-III and DLP-IV with the two distinct exponentiations, i.e. $\gamma \equiv \alpha^a \beta^b \bmod p$ in the multiplicative group of the finite field Z_p^* of the order $p-1$. We show that the proposed Public Key Cryptosystem based on DLP-IV provides more security because of double computation comparing with the well known the Discrete Logarithm Problem II or DLP-II with one exponentiation, i.e. $\beta \equiv \alpha^a \bmod p$ in the multiplicative group of the finite field Z_p^* of the order $p-1$ at the same efficiency level.

Keywords: Discrete logarithm problem or DLP, public key cryptosystem or PKC,

1 Introduction

It is well known that the security of any Public Key Cryptosystem (PKC) depends upon hardness of a mathematical problem. The PKC known as RSA [4] was based on factoring problem whereas the PKC given by ElGamal [2] was based on the Discrete Logarithm Problem (DLP). Prior to this, DLP-I used in Diffie and Hellman protocol [1] was with one exponentiation and one unknown random integer in the finite cyclic group G of the order n as below:

1) Discrete Logarithm Problem I (DLP-I):
The Diffie and Hellman protocol [1] was having the discrete logarithm problem with one exponentiation and one random integer as follows: $\beta = \alpha^a$, in the finite cyclic group G of the order n, where α is generator of G, $\beta \in G$, and a is the random integer such that $0 \le a \le n-1$. Here, the difficulty of computing the value of the random integer a is called the discrete logarithm problem of β to the base α. It is denoted as:

$$a = \log_\alpha \beta.$$

The difficulty of said above problem was the computation of the value of the unknown random integer. First improvement in this direction was to solve above DLP under the multiplicative group of the finite field Z_p^* of the order $p-1$ where p is the large prime.

2) Discrete Logarithm Problem II (DLP-II):
The improvement in the above problem was formulated as below: $\beta \equiv \alpha^a \bmod p$, in the multiplicative group of the finite field Z_p^* of the order $p-1$, where p is a prime number, α is a primitive element under modulo p, Z_p^*, and a is the random integer.

It was ElGamal [2] who proposed PKC based on DLP-II. Later, several modifications were made in DLP-II to improve the security of the PKC. In this paper, using the logic that double computation of two distinct DLP would provide double security at same efficiency level, we first propose, a discrete logarithm problem - III or DLP-III with the two different exponentiations and two random and distinct integers in the finite cyclic group G of the order n. Consequently, we then obtain another DLP-IV in the multiplicative group of the finite field Z_p^* of the order $p-1$ where p is the large prime.

3) Discrete Logarithm Problem III (DLP-III):
The discrete logarithm problem with the two different exponentiations and the two random and distinct integers, we define as below: $\gamma \equiv \alpha^a \beta^b$, in the finite cyclic group G of order n, such that $\alpha \neq \beta^i$ and $a \neq b^i$, where α and β be two distinct generators of G, $\gamma \in G$, and a, b be the two distinct random integers.

4) Discrete Logarithm Problem IV (DLP-IV):
Next, as a direct consequence of the discrete logarithm problem - III, we propose DLP-IV as below: $\gamma \equiv \alpha^a \beta^b \bmod p$, in the multiplicative group of the finite field Z_p^* of order $p-1$, such that $\alpha \neq \beta^i$ and $a \neq b^i$, where p is a prime number, α and β be two primitive elements under modulo p, $\gamma \in Z_p^*$, and a, b be two distinct random integers.

We assert that computing the values of the two distinct random integers a and b in DLP-III and DLP-IV with the

two distinct exponentiations respectively are more difficult as compare to DLP-I and DLP-II with one exponentiation. The Shanks Baby-Step Giant-Step method [5], Pollard-ρ method [5], Pohling-Hellman method [3] and Index-Calculus method [5] are the best known methods for computing any DLP. Using these methods, we demonstrate that double computation is required in DLP-III and DLP-IV as compare to DLP-I and DLP-II respectively, making them more difficult. For the simple reason, the algorithms corresponding DLP-III and DLP-IV would require the more time and space. As result, the design of public key cryptosystems based on the discrete logarithm problem III and IV becomes more secure at the same efficiency level as compare to the all those public key cryptosystems, which are based on the discrete logarithm problem with one exponentiation.

It is important to mention that their efficiency remains the same. Because, any programming for the purpose of the computation of the two Discrete Logarithm Problems or DLPs with two different parameters would take equal time as the computation of one Discrete Logarithm Problem or DLP. Resultant, this makes the new DLPs, i.e. DLP-III and DLP-IV equally efficient as compare to the previous DLP, i.e. DLP-I and DLP-II respectively. In the following, we only need to recall the computing algorithms for DLP-III and DLP-IV and to show that those require the double computation.

2 The Algorithm for Computing the Discrete Logarithm Problem

2.1 The Shanks Baby-Step Giant-Step Algorithm

First, we give the Shanks Baby-Step Giant-Step Algorithm [5] as follows:

1) $m \leftarrow \sqrt{n}$
2) for $j \leftarrow 0$ to $m-1$, do compute α^{mj}
3) sort the m ordered pairs (j, α^{mj}) with respect to their second coordinates, obtaining a list L_1
4) for $i \leftarrow 0$ to $m-1$, do compute $\beta\alpha^{-i}$
5) sort the m ordered pairs $(i, \beta\alpha^{-i})$ with respect to their second coordinates, obtaining a list L_2
6) Find a pair $(j, y) \in L_1$ and a pair $(i, y) \in L2$ (i.e. find the two pairs having identical second coordinates)
7) $\log_\alpha \beta \leftarrow (mj + i) \bmod n$

2.2 The Index-Calculus Algorithm

Next, we give the Index-Calculus Algorithm [5] for cyclic group G of the order n, as follows:

INPUT: A generator α of the finite cyclic group G of the order n, and an element $\beta \in G$.

OUTPUT: The discrete logarithm problem $a = \log_\alpha \beta$.

1) Select a factor base S:
Choose a subset $S = \{p_1, p_2, p_3, \ldots, p_t\}$ of G such that a "significant proportion" of all the elements in G can be efficiently expressed as the product of elements from S.

2) Collect linear relations involving logarithms of elements in S:

   a. Select a random integer k, $0 \le k \le n-1$, and compute α^k.
   b. Try to write α^k as a product of elements in S:

$$\alpha^k = \prod_{i=1}^{t} p_i^{c_i}{}_i, c_i \ge 0. \quad (1)$$

   If successfully take logarithms of both the sides of Equation (1) to obtain a linear relation:

$$k = \sum_{i=1}^{t} c_i \log_\alpha p_i (\mathrm{mod} n). \quad (2)$$

   c. Repeat Steps 2a and 2b until $t + c$ relations of the form (2) are obtained (c is a small positive integer, e.g. $c = 10$, such that the system of equations given by $t = c$ relations has a unique solutions with high probability).

3) Find the logarithms of elements in S:
Working modulo n, solve the linear system of $t + c$ relations (in t unknowns) of the form (2) collected in Step 2 to obtain the value of $\log_\alpha p_i$, $1 \le i \le t$.

4) Compute a:

   a. Select a random integer k, $0 \le k \le n-1$, and compute $\beta\alpha^k$.
   b. Try to write $\beta\alpha^k$ as a product of elements in S:

$$\beta\alpha^k = \prod_{i=1}^{t} (p_i)^{d_i}, d_i \ge 0.$$

   If the attempt is unsuccessful, then repeat Step 4a, otherwise taking logarithms of both the sides of the above equation yields $\log_\alpha \beta = (\sum_{i=1}^{t} d_i \log_\alpha p_i - k) \bmod n$. Thus compute $a = (\sum \sum_{i=1}^{t} d_i \log_\alpha p_i - k) \bmod n$, and return to a.

The Index-Calculus algorithm for computing the discrete logarithm problem in the multiplicative group of the finite field Z_p^* would be consequently as follows.

For the multiplicative group of the finite field Z_p^*, p is a prime, the factor base S can be chosen as the prime numbers. The relation (1) is generated by computing k mod p and then using trial division to check whether this integer is a product of the primes in S.

3 The ElGamal Public Key Cryptosystem

Let us now recall the ElGamal public key cryptosystem [2] before we propose our PKC: The ElGamal public key cryptosystem based on the classical discrete logarithm problem with one exponentiation in the multiplicative group of the finite field Z_p^* of the order $p-1$, is defined as follows.

Let p be a prime such that the discrete logarithm problem in Z_p^* of the order $p-1$, is infeasible and let $\alpha \in Z_p^*$ be a primitive element. Let $P = Z_p^*$, $C = Z_p^* X Z_p^*$, and define $K = \{(p, \alpha, a, \beta) : \beta \equiv \alpha^a \bmod p\}$.

The value p, α and β are the public key and a is the private key. For $K = (p, \alpha, a, \beta)$ and for a(secret) random number $k \in Z_{p-1}$, define $e_K(x, k) = (y_1, y_2)$ where $y_1 = \alpha^k \bmod p$ and $y_2 = x\beta^k \bmod p$, for $y_1, y_2 \in Z_p^*$, and $d_K(y_1, y_2) = y_2(y_1^a)^{-1} \bmod p$.

4 The Proposed Public Key Cryptosystem

Next, we give the algorithm of the key generation, the encryption and the decryption for the proposed public key cryptosystem corresponding to DLP-IV in the following subsections.

4.1 The Key Generation

Each entity creates the public key and the corresponding private key. Each entity A observe the following steps:

1) Generate a large random prime p and the two distinct primitive generators α and β, such that $\alpha \neq \beta^i$ of the multiplicative group Z_p^* of the integers modulo p of the order $p-1$.
2) Select the two random integers a and b, such that $a \neq b^i$ and $1 \leq (a, b) \leq p-2$.
3) Compute $\alpha^a \bmod p$ and $\beta^b \bmod p$.
4) A's public key is $(p, \alpha, \beta, \alpha^a, \beta^b)$ and A's private key is (a, b).

4.2 The Encryption

B encrypts the message m for A, which A decrypts. B observe the following steps:

1) Obtain A's public key $(p, \alpha, \beta, \alpha^a, \beta^b)$.
2) Represent the message as an integer m in the range $\{0, 1, 2, 3, 4, \ldots, p-1\}$.
3) Select the random integer k, such that $1 \leq k \leq p-2$.
4) Compute $\gamma \equiv \alpha^k \bmod p$, $\delta \equiv \beta^k \bmod p$, and $\eta \equiv m(\alpha^a)^k(\beta^b)^k \bmod p$.
5) Send the cipher text $c = (\gamma, \delta, \eta)$ to A.

4.3 The Decryption

To recover the plaintext m from c, A observe the following steps:

1) Use the private key (a, b) to compute $\gamma^{p-1-a} \bmod p$, and $\gamma^{p-1-b} \bmod p$ (note, $\gamma^{p-1-a} = \gamma^{-a} = \alpha^{-ak}$ and $\delta^{p-1-b} = \delta^{-b} = \beta^{-bk}$).
2) Recover m by computing $(\gamma^{-a})(\delta^{-b})(\eta) \bmod p$.

5 The Complexity of DLP-III and DLP-IV

In this section, we prove the theorems in support of the complexity of DLP-III and DLP-IV. First, we prove the theorem on the complexity of DLP-III as follows.

Theorem 1. *DLP-III involves the two distinct discrete logarithm problems in the form of DLP-I.*

Proof. We know that the mathematical structure of DLP-I in the finite cyclic group G of order n is defined as follows:

$$\alpha^a \equiv \beta.$$

Taking the logarithm of both the sides of the above equation to the base α is

$$\begin{aligned} & \log_\alpha \beta = a, \\ \text{or} \quad & a = \log_\alpha \beta. \end{aligned} \tag{3}$$

Next, the mathematical structure of DLP-III in the finite cyclic group G of order n is defined as follows:

$$\alpha^a \beta^b = \gamma. \tag{4}$$

Taking logarithm of both the sides of Equation (4) to the base α, we have

$$\begin{aligned} & \log_\alpha(\alpha^a \beta^b) = \log_\alpha \gamma, \\ \Rightarrow \quad & \log_\alpha \alpha^a + \log_\alpha \beta^b = \log_\alpha \gamma, \\ \Rightarrow \quad & a \log_\alpha \alpha + b \log_\alpha \beta = \log_\alpha \gamma, \\ \Rightarrow \quad & a + b \log_\alpha \beta = \log_\alpha \gamma, \\ \Rightarrow \quad & a = \log_\alpha \gamma - b \log_\alpha \beta, \\ \Rightarrow \quad & a = \log_\alpha \gamma - \log_\alpha \beta^b, \\ \Rightarrow \quad & a = \log_\alpha(\gamma/\beta^b). \end{aligned} \tag{5}$$

Again, taking logarithm of both the sides of Equation (4) to the base β:

$$\begin{aligned} & \log_\beta(\alpha^a \beta^b) = \log_\beta \gamma, \\ \Rightarrow \quad & \log_\beta \alpha^a + \log_\beta \beta^b = \log_\beta \gamma, \\ \Rightarrow \quad & a \log_\beta \alpha + b \log_\beta \beta = \log_\beta \gamma, \\ \Rightarrow \quad & a \log_\beta \alpha + b = \log_\beta \gamma, \\ \Rightarrow \quad & b = \log_\beta \gamma - a \log_\beta \alpha, \\ \Rightarrow \quad & b = \log_\beta \gamma - \log_\beta \alpha^a, \\ \Rightarrow \quad & b = \log_\alpha(\gamma/\alpha^a). \end{aligned} \tag{6}$$

Equation (3) represents DLP-I whereas Equations (5) and (6) represents DLP-III with two distinct discrete logarithm problems in the form of DLP-I, eventually making the computation of DLP-III more difficult. This completes the proof. □

Next, we prove the theorem on the complexity of DLP-IV as follows:

Theorem 2. *DLP-IV involves the two distinct discrete logarithm problems in the form of DLP-II.*

Proof. We know that, The mathematical structure of DLP-II in the multiplicative group of the finite field Z_p^* of order $p-1$ is defined as follows:

$$\alpha^a \equiv \beta \bmod p.$$

Taking logarithm of both the sides of the above equation to the base α:

$$\log \alpha(\beta \bmod p) \equiv a. \tag{7}$$

Now, the mathematical structure of DLP-IV in the multiplicative group of the finite field Z_p^* of order $p-1$ is defined as follows:

$$\alpha^a \beta^b \equiv \gamma \bmod p. \tag{8}$$

Taking logarithm of both the sides of Equation (8) to the base α, we have,

$$\begin{aligned}
& \log_\alpha(\alpha^a\beta^b) \equiv \log_\alpha(\gamma \bmod p),\\
\Rightarrow\ & \log_\alpha \alpha^a + \log_\alpha \beta^b = \log_\alpha(\gamma \bmod p),\\
\Rightarrow\ & a\log_\alpha \alpha + b\log_\alpha \beta = \log_\alpha(\gamma \bmod p),\\
\Rightarrow\ & a + b\log_\alpha \beta \equiv \log_\alpha(\gamma \bmod p),\\
\Rightarrow\ & a \equiv \log_\alpha(\gamma \bmod p) - b\log_\alpha \beta,\\
\Rightarrow\ & a \equiv \log_\alpha(\gamma \bmod p) - \log_\alpha \beta^b.\\
\Rightarrow\ & a \equiv \log_\alpha(\frac{\gamma}{\beta^b} \bmod p). \qquad (9)
\end{aligned}$$

Again, taking logarithm of both the sides of Equation (8) to the base β:

$$\begin{aligned}
\Rightarrow\ & \log_\beta(\alpha^a\beta^b) \equiv \log_\beta(\gamma \bmod p),\\
\Rightarrow\ & \log_\beta \alpha^a + \log_\beta \beta^b \equiv \log_\beta(\gamma \bmod p),\\
\Rightarrow\ & a\log_\beta \alpha + b\log_\beta \beta \equiv \log_\beta(\gamma \bmod p),\\
\Rightarrow\ & a\log_\beta \alpha + b \equiv \log_\beta(\gamma \bmod p),\\
\Rightarrow\ & b \equiv \log_\beta(\gamma \bmod p) - a\log_\beta \alpha,\\
\Rightarrow\ & b \equiv \log_\beta(\gamma \bmod p) - \log_\beta \alpha^a,\\
\Rightarrow\ & b \equiv \log_\beta(\frac{\gamma}{\alpha^a} \bmod p). \qquad (10)
\end{aligned}$$

Equation (7) represents DLP-II where as Equations (9) and (10) represents DLP-IV involving two distinct discrete logarithm problems in the form of DLP- II and thus making the computation of DLP-IV more difficult. This completes the proof. □

Similarly, we prove the theorem on complexity of DLP-III using the Shanks Baby-Step Giant-Step Algorithm as follows.

Theorem 3. *The Shanks Baby-Step Giant-Step Algorithm requires the double computation to compute DLP-III, i.e.* $\gamma = \alpha^a\beta^b$ *such that* $\alpha \neq \beta^i$, $a \neq b_i$ *as compare to DLP-I, i.e.* $\beta = \alpha^a$ *in the finite cyclic group* G *of the order* n.

Proof. Applying the Shanks Baby-Step Giant-Step Algorithm [2.1] for computing DLP-I. First, Steps 2.1(a) and 2.1(b) can be precomputed, if desired (this will not affect the asymptotic running time, however).

If an ordered pair $(j, y) \in L_1$(The first list) and an ordered pair $(i, y) \in L_2$(The second list) then $(\alpha)^{mj} = y = \beta(\alpha)^{-i}$. Therefore, $(\alpha)^{mj+i} = \beta$.

Taking the logarithm of both the sides of the above equation to the base α:

$$\begin{aligned}
& \log_\alpha\{(\alpha)^{mj+i}\} = \log_\alpha \beta,\\
& (mj+i)\log_\alpha \alpha = \log_\alpha \beta,\\
& mj + i = \log_\alpha \beta, \text{ where, } 0 \le j, i \le m-1. \qquad (11)
\end{aligned}$$

Since all terms in the above congruence are now known, except for $\log_\alpha \beta$, we can easily solve for $\log_\alpha \beta$.

Next, if we apply Shanks Baby-Step Giant-Step Algorithm [2.1] to DLP-III, i.e. $\delta = \alpha^a\beta^b$ such that $\alpha \neq \beta^i$, $a \neq b_i$ in the finite cyclic group G of the order n.

If $(j, y) \in L_1$ (The first list) and $(i, y) \in L_2$ (The second list), there are three cases are listed as follows.

Case 1:

$$(\alpha^a\beta^b)^{mj} = y = \gamma(\alpha^a\beta^b)^{-i}.$$

Therefore,

$$(\alpha^a\beta^b)^{mj+i} = \gamma. \tag{12}$$

Taking the logarithm of both the sides of Equation (12) to the base α:

$$\begin{aligned}
& \log_\alpha(\alpha^a\beta^b)^{mj+i} = \log_\alpha \gamma\\
\Rightarrow\ & (mj+i)\log_\alpha(\alpha^a\beta^b) = \log_\alpha \gamma\\
\Rightarrow\ & (mj+i)\{\log_\alpha \alpha^a + \log_\alpha \beta^b\} = \log_\alpha \gamma\\
\Rightarrow\ & (mj+i)\{a\log_\alpha \alpha + b\log_\alpha \beta\} = \log_\alpha \gamma\\
\Rightarrow\ & (mj+i)\{a + b\log_\alpha \beta\} = \log_\alpha \gamma\\
\Rightarrow\ & mj + i = (\log_\alpha \gamma)/(a + b\log_\alpha \beta),\\
& \qquad \text{where, } 0 \le j, i \le m-1. \qquad (13)
\end{aligned}$$

Since all terms in the above congruence are now known, except for $\log_\alpha \beta$ and $\log_\alpha \gamma$, first we can solve for $\log_\alpha \beta$ then after $\log_\alpha \gamma$, simultaneously.

Case 2:
Again taking the logarithm of both the sides of Equation (12) to the base β:

$$\begin{aligned}
& \log_\beta(\alpha^a\beta^b)^{mj+i} = \log_\beta \gamma\\
\Rightarrow\ & (mj+i)\log_\beta(\alpha^a\beta^b) = \log_\beta \gamma\\
\Rightarrow\ & (mj+i)\{\log_\alpha \alpha^a + \log_\alpha \beta^b\} = \log_\alpha \gamma
\end{aligned}$$

$$\begin{aligned}
&\Rightarrow \quad (mj+i)\{a\log_\beta \alpha + b\log_\beta \beta\} = \log_\beta \gamma \\
&\Rightarrow \quad (mj+i)\{a\log_\beta \alpha + b\} = \log_\beta \gamma \\
&\Rightarrow \quad mj+i = (\log_\beta \gamma)/(a\log_\beta \alpha + b), \\
&\qquad\qquad \text{where, } 0 \le j, i \le m-1. \qquad (14)
\end{aligned}$$

Since all terms in the above congruence are now known, except for $\log_\beta \alpha$ and $\log_\beta \gamma$, first we can solve for $\log_\beta \alpha$ then after $\log_\beta \gamma$ simultaneously.

Case 3:
Again taking the logarithm of both the sides of Equation (12) to the base γ:

$$\begin{aligned}
&\log_\gamma (\alpha^a \beta^b)^{mj+i} = \log_\gamma \gamma \\
&\Rightarrow \quad (mj+i)\log_\gamma (\alpha^a \beta^b) = 1 \\
&\Rightarrow \quad (mj+i)\{\log_\gamma \alpha^a + \log_\gamma \beta^b\} = 1 \\
&\Rightarrow \quad (mj+i)\{a\log_\gamma \alpha + b\log_\gamma \beta\} = 1 \\
&\Rightarrow \quad mj+i = 1/(a\log_\gamma \alpha + b\log_\gamma \beta), \\
&\qquad\qquad \text{where, } 0 \le j, i \le m-1. \qquad (15)
\end{aligned}$$

Since all terms in the above congruence are now known, except for $\log_\gamma \alpha$ and $\log_\gamma \beta$, first we can solve for $\log_\gamma \alpha$ then after $\log_\gamma \beta$ simultaneously.

If we compare Equation (11) from the Equations (13), (14) and (15) respectively, then we can see that the Shanks Baby-Step Giant-Step Algorithm requires the double computation to compute DLP-III, i.e. $\gamma = \alpha^a \beta^b$ such that $\alpha \neq \beta^i$, $a \neq b^i$ as compare to DLP-I, i.e. $\beta = \alpha^a$ in the finite cyclic group G of the order n, because DLP-III involves to two distinct discrete logarithm problems in the form of DLP-I in each case (By Theorem 1) whereas DLP-I has itself only one discrete logarithm problem. Therefore DLP-III definitely requires the double computation. This situation makes DLP-III more difficult than DLP-I. □

Finally, we prove the theorem on the complexity of DLP-IV using the Index-Calculus Algorithm as follows.

Theorem 4. *The Index-Calculus Algorithm requires the double computation to compute DLP-IV, i.e.* $\gamma = \alpha^a \beta^b \bmod p$ *such that* $\alpha \neq \beta^i$, $a \neq b^i$ *as compare to DLP-I, i.e.* $\beta = \alpha^a \bmod p$ *in the multiplicative group of the finite field* Z_p^* *of the order* $p-1$.

Proof. The Index-Calculus Algorithm [2.2] for computing DLP-II, i.e. $\beta \equiv \alpha^a \bmod p$ in the multiplicative group of the finite field Z_p^* of the order $p-1$ bears considerable resemblance to many of the best factoring algorithm. The index-calculus method uses a factor base, which is a set B of the small primes. Suppose $B = \{p_1, p_2, p_3, \ldots, pB\}$. The first step is to find the logarithms of the primes in the factor base.

The second step is to compute the discrete logarithm problem of the desired element γ, using the knowledge of the discrete logarithm problems of the elements in the factor base.

Let C be a bit bigger than B: say $C = B + 10$.

Now, we suppose that we have already successfully carried out the precomputation step and we compute a desired logarithm $\log_\alpha \beta$ by means of a Las-vegas type randomized algorithm [5]. Choose a random integer $s (1 \le s \le p-2)$ and compute

$$\gamma \equiv \beta\alpha^s \bmod p.$$

Now attempt to factor γ over the factor base B. If this can be done, then we obtain a congruence of the form:

$$\beta\alpha^s \equiv p_1^{c_1} p_2^{c_2} \ldots p_B^{c_B} (\bmod\ p).$$

This can be written equivalently as

$$\begin{aligned}
&\log_\alpha \beta + s \\
\equiv\ & c_1 \log_\alpha p_1 + c_2 \log_\alpha p_2 + \ldots + c_B \log_\alpha p_B (\bmod p-1). \qquad (16)
\end{aligned}$$

Since all terms in the above congruence are now known, except for $\log_\alpha \beta$, we can easily solve for $\log_\alpha \beta$.

Now, we apply the Index-Calculus Algorithm to DLP-IV, i.e. $\gamma = \alpha^a \beta^b \bmod p$ such that $\alpha \neq \beta^i$, $a \neq b^i$ in the multiplicative group of the finite field Z_p^* of the order $p-1$. Then three cases arise.

Case 1:
If we have already successfully carried out the precomputation step, then, we compute DLP-IV as follows.

Choose a random integer $s (1 \le s \le p-2)$ and compute:

$$\delta \equiv \gamma(\alpha\beta)^s \bmod p.$$

We attempt to factor γ over the factor base B. If this can be done, then we obtain a congruence of the form:

$$\gamma(\alpha\beta)^s \equiv p_1^{c_1} p_2^{c_2} \ldots p_B^{c_B} (\bmod\ p). \qquad (17)$$

Taking the logarithm of the above congruence (17) to the base α, then

$$\begin{aligned}
&\log_\alpha \gamma + s + s\log_\alpha \beta \\
\equiv\ & c_1 \log_\alpha p_1 + c_2 \log_\alpha p_2 + \ldots + c_B \log_\alpha p_B (\bmod p-1). \qquad (18)
\end{aligned}$$

Since all terms in the above congruence are now known, except for $\log_\alpha \beta$ and $log_\alpha \gamma$, first we solve for $\log_\alpha \beta$ and then for $log_\alpha \gamma$.

Case 2:
Taking the logarithm of the above congruence (17) to the base β, then

$$\begin{aligned}
&\log_\beta \gamma + s\log_\beta \alpha + s \\
\equiv\ & c_1 \log_\beta p_1 + c_2 \log_\beta p_2 + \ldots + c_B \log_\beta p_B (\bmod\ p-1). \qquad (19)
\end{aligned}$$

Since all terms in the above congruence are now known, except for $\log_\beta \alpha$ and $\log_\beta \gamma$, we solve for $\log_\beta \alpha$ and then

for $\log_\beta \gamma$.

Case 3:
Taking the logarithm of the above congruence (17) to the base γ, then

$$\begin{aligned} & 1 + s\log_\gamma \alpha + s\log_\gamma \beta \\ \equiv\ & c_1 \log_\gamma p_1 + c_2 \log_\gamma p_2 + \cdots + c_B \log_\gamma p_B (\mathrm{mod} p-1). \end{aligned} \tag{20}$$

Since all terms in the above congruence are now known, except for $\log_\gamma \alpha$ and $\log_\gamma \beta$, we solve for $\log_\gamma \alpha$ and then $\log_\gamma \beta$.

Now if we compare the congruence (16) from the congruences (18), (19), and (20) respectively, then we can clearly see that the Index-Calculus Algorithm requires the double computations to compute DLP-IV, i.e. $\gamma \equiv \alpha^a \beta^b \bmod p$ such that $\alpha \neq \beta^i$, $a \neq b^i$ as compare to DLP-II, i.e., $\beta = \alpha^a \bmod p$ in the multiplicative group of the finite field Z_p^* of the order $p-1$, because DLP-IV involves the two distinct discrete logarithm problems in the form of DLP-II in each case (By Theorem 2) whereas DLP-II has itself only one discrete logarithm problem to compute. As result, DLP-IV becomes more difficult than DLP-II. □

6 Conclusion

The problems proposed by us still hold the fact that the scope of DLP-I and DLP-II are unlimited. DLP-I and DLP-II which were initially proposed by Diffie and Hellman in the year 1976 and had numerous improvements in the past still possess tremendous potential of improvement from security point of view. This paper is not only introduced the new public key cryptosystem, but also tried to put the new concept of the discrete logarithm problem as the form of DLP-III and DLP-IV.

References

[1] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, pp. 644–654, 1976.

[2] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, pp. 469–472, 1985.

[3] S. C. Pohling and M. E. Hellman, "An improved algorithm for computing logarithms over GF(p) and its cryptographic significance," *IEEE Transactions on Information Theory*, vol. 24, pp. 106–110, 1978.

[4] R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," *Communication of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[5] D. R. Stinson, *Cryptography: Theory and Practice*, CRC Press, Boca Raton, Florida, 1995.

Sunil Kumar Kashyap has received B.Sc. and M.Sc. in Mathematics from Pandit Ravishankar Shukla University, Raipur Chhattisgarh, INDIA. At present, He is teaching as an Assistant Professor in Durga Mahavidyalaya, Raipur, Chhattisgarh, India. He is doing his research as a Research Scholor in the field of Cryptography, in Pandit Ravishanker Shukla University, Raipur Chhattisgarh, INDIA. He is life-time member of Cryptology Research Society of India, Indian Statistical Institute, Kolkota, India. Public Key cryptography is the area of research of him. He takes interest to work in Teaching and Research.

Birendra Kumar Sharma presently working Professor & Head, School of Studies in Mathematics, Pt. Ravishankar Shukla University Raipur, (C. G.) India has been workingin the field of Non Linear Operator Theorey for a long time. 15 scholars have got their Ph.D. degree under his guidance in the field of fixed point theory. Since last three years, he moved to work in the applied field such as Cryptography. He is a life member of Indian Mathematical society and the Ramanujan Mathematical Society.

Amitabh Banerjee has received M.Sc. in 1983 and Ph.D. in 1996 from Pt. Ravishankar Shukla University, Chhattisgarh, India. At present he is an Assistant Professor in Mathematicas, in Govt. D.B. Girls (Post Graduate) college, Raipur, Chhattisgarh, India. His area of research in the field of fixed point theory and now he moved to the field of cryptography.