MDPI

Article

On the Investigation of Variants for Discrete Logarithm Problems in Abelian Groups: An Algebraic Structure Approach

Denis C.K. Wong [1,*] and Lik-How Low [2]

[1] Affiliation 1; deniswong@utar.edu.my
[2] Affiliation 2: lowlikhow@1utar.my
* Correspondence: deniswong@utar.edu.my

Abstract

In this work, we investigate variants of the discrete logarithm problem (DLP) based on different algebraic group properties. We demonstrate that within abelian groups—particularly cyclic groups—the problems DLP^2 and, more generally, DLP^n are polynomially reducible to solving multiple instances of the classical DLP. This result confirms that multi-generator variants in abelian settings inherit the vulnerabilities of the conventional DLP and therefore provide no additional resistance against quantum attacks such as Shor's algorithm. Motivated by this limitation, we propose that DLP^n should instead be formulated in non-abelian groups. Specifically, let G be a non-abelian group. Given generators $g_1, g_2, \ldots, g_n \in G$ of cyclic subgroups $H_1, H_2, \ldots, H_n$ of G, respectively, and an element $A \in G$, the problem is to find $(\alpha_1, \alpha_2, \ldots, \alpha_n)$ such that $A = \prod_{i=1}^{n} g_i^{\alpha_i}$, where $0 \leq \alpha_i < |H_i|$. This formulation leverages the non-commutative structure of G, which may obstruct reductions that trivialize multi-generator variants in abelian groups. Our findings strongly motivate the exploration of quantum-resistant cryptographic primitives in non-commutative algebraic structures, as these settings may evade known quantum attacks and provide new hardness assumptions for post-quantum security.

Keywords: Discrete Logarithm Problem; abelian groups; cyclic groups

1. Introduction

Discrete Logarithm Problem (DLP) is the foundational difficulty assumption underlying much of modern public-key cryptography. Historically, the DLP first appeared in [4], the work of Diffie and Hellman (1976), who introduced the notion of public-key cryptography and demonstrated a method, now known as the Diffie-Hellman Key Exchange, for two parties to agree on a shared secret over an insecure channel. Building on this foundation, ElGamal (1985) [5] was the first to formalize the problem into practical cryptographic schemes, pioneering the use of DLP to underpin both encryption and digital signature schemes.

Recent researchers have extensively analyzed the computational hardness of the DLP, refining the boundaries of its security in abelian settings. On the practical front, Tanujaya et al. [9] conducted comparative efficiency analyses on elliptic curve DLP (ECDLP) solvers, confirming that while Shanks' Baby-Step-Giant-Step offers speed for smaller orders, Pollard's Rho remains the most efficient method for higher-order curves, requiring larger key sizes to maintain security. Simultaneously, the theoretical hardness of the problem is under constant pressure from algorithmic breakthroughs. For example, Lido [8] recently presented a provably quasi-polynomial algorithm for computing discrete logarithms in finite fields of

Received:
Revised:
Accepted:
Published:

Copyright: © 2026 by the authors. Submitted to *Journal Not Specified* for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license.

small characteristic, utilizing elliptic presentations to significantly lower the complexity estimates for these specific fields. Furthermore, generic discrete logarithm algorithms are being continuously optimized. Wu et al. [13] investigated the non-uniform birthday problem to refine the Gaudry-Schost algorithm, demonstrating that precise parameter selection can further reduce the time complexity for solving a DLP in an interval.

In response to these advances in algorithmic attacks, researchers have proposed modifying the underlying algebraic structures or problem definitions to improve security. Tripathi et al. [10] introduced the "Dependent RSA Discrete Logarithm Problem", a public-key cryptosystem that integrates the RSA and DLP assumptions to achieve IND-CCA2 security, suggesting that hybrid hardness assumptions might offer more resilience. Similarly, the complexity of solving multiple instances simultaneously has been extensively analyzed; Wu and Zhuang [14] derived tighter lower bounds for the Generalized Multiple Discrete Logarithm Problem (solving k out of n instances), employing matrix techniques and Number Theoretic Transforms to analyze the reduction complexity. Despite these innovations, such variants largely remain within abelian groups. This limitation raises the critical question of whether such modifications are sufficient to evade quantum cryptanalysis, or if they ultimately reduce to the standard DLP vulnerabilities inherent to abelian groups.

Beyond improvements in concrete algorithms, a substantial body of work demonstrates that the difficulty of the DLP is fundamentally constrained by the algebraic structure of the underlying group. In the generic group model, where algorithms are limited to black-box group operations and cannot exploit representation-specific features—these limitations become explicit. Shoup [17] established tight lower bounds for generic discrete logarithm algorithms, proving that any solver in a cyclic group whose order is divisible by a large prime p must perform $\Omega(\sqrt{p})$ group operations. This bound matches classical algorithms such as Baby-Step–Giant-Step and Pollard's Rho, indicating that their performance is not a consequence of algorithmic inefficiency, but rather reflects an intrinsic barrier imposed by abelian group structure itself. This perspective was extended to settings with multiple instances. Yun [16] proved that solving n discrete logarithms in parallel in a generic abelian group requires $\Omega(\sqrt{np})$ group operations, ruling out any substantial asymptotic advantage from parallelism alone.

Ying and Kunihiro [15] further generalized this phenomenon to the (k, n)-Generalized Multiple Discrete Logarithm Problem, showing that even when the solver is permitted to recover only a subset of instances, the same structural square-root barrier persists across a wide range of parameters. Taken together, these results indicate that within abelian groups, the hardness of the DLP is governed less by the absence of efficient algorithms than by the linear constraints inherent to the group operation. In particular, access to multiple instances, auxiliary information, or generalized formulations does not alter the underlying algebraic limitations.

This work adopts a structural perspective: rather than pursuing faster algorithms, we examine the abelian group structure itself, which governs the behavior of discrete logarithms and inherently constrains the potential for long-term cryptographic hardness. While much of the existing literature on the DLP emphasizes algorithmic efficiency, our approach is structural — we investigate which aspects of the problem arise intrinsically from abelian group properties, independent of specific solution techniques.

In this paper, we revisit the structure of the DLP through the lens of group theory. To make the discussion accessible, we begin with a concise and intuitive overview of the relevant algebraic objects: groups, cyclic groups, abelian groups, normal subgroups, and the natural homomorphisms that connect them. These concepts form the underlying "machinery" that dictates how discrete logarithms behave.

1.1. Aim of paper.

To construct public-key cryptographic primitives that surpass the hardness of the classical Discrete Logarithm Problem (DLP), one must look beyond cyclic groups—and even beyond the broader class of abelian groups. Shor's algorithm [6] demonstrates that the DLP can be solved in polynomial time on a quantum computer, underscoring the urgency of identifying quantum-resistant alternatives. Since every problem in a cyclic group ultimately reduces to the classical DLP, this motivates the exploration of discrete logarithm-type problems in richer algebraic settings, such as non-cyclic abelian groups. In this paper, we investigate whether multi-generator variants, including the DLP^2 proposed by KAZ [2], succeed in offering increased complexity. We clarify the structural limitations of these formulations, showing that even in non-cyclic abelian groups—where reductions are less immediate—the multi-generator problems remain reducible to the classical DLP. As a result, these variants inherit the same quantum vulnerability and do not provide stronger resistance than the standard DLP.

1.2. Organization of paper.

The remainder of this paper is structured as follows. Section 2 reviews the necessary background in group theory and provides illustrative examples of the Discrete Logarithm Problem across various abelian contexts. Section 3 analyzes the security of multi-generator variants in cyclic groups. Section 4 extends this analysis to non-cyclic abelian groups and presents a formal proof that the DLP^n reduces to classical DLP instances. Section 5 concludes the paper and outlines directions for future research.

2. Definitions and Preliminary Results

2.1. Variants of DLP.

The motivation for this work arises from the DLP^2 introduced in the study of KAZ [2]. In their original formulation, the definition of DLP^2 was presented ambiguously, which rendered the proposed hard problem — and the associated cryptographic schemes — vulnerable to several types of attacks. In this paper, we provide a more rigorous and extensive formulation of DLP^2, grounded in fundamental concepts of group theory. Our approach allows us to derive a variety of results in both cyclic and abelian groups. Unless stated otherwise, all groups considered throughout this paper are assumed to be multiplicative.

Conventionally, the DLP is defined over a cyclic group of prime order, as formalized below.

Definition 1. *Let $G = \langle g \rangle$ be a cyclic group of prime order p with generator g. Given g and a target element $A \in G$, find the exponent $\alpha \in \mathbf{Z}_p$ such that $g^\alpha = A$.*

We denote the discrete logarithm of A with base g as $dlog_g(A) = \alpha$. The concept of the DLP can be generalized to finite groups of arbitrary order, but more careful treatment is required. The following Double Discrete Logarithm Problem (DDLP) was proposed by [3] and is based on two cyclic groups.

Definition 2. *Suppose p and q are distinct primes. Let $\mathbf{Z}_p$ be the ring of integers modulo p, let $G_{q,p} \subset \mathbf{Z}_p$ be a multiplicative cyclic group of order q, and let G_p be a cyclic group of order p. Let g and h be the generators of G_p and $G_{q,p}$, respectively. The double discrete logarithm problem is to find $x \in \mathbf{Z}_q$ such that $A = g^{h^x}$, given $(G_p, G_{q,p}, g, h, A)$.*

To solve the DDLP, we first need to solve the DLP in G_p to obtain $y = dlog_g(A) \in \mathbf{Z}_p$, then we solve the DLP in $G_{q,p}$ to obtain $x = dlog_h(y) \in \mathbf{Z}_q$. In short, we need to find $x = dlog_h(dlog_g(A))$.

https://doi.org/10.3390/1010000

Next, we highlight the double discrete logarithm problem (DLP^2) proposed by KAZ and reformulate it over an abelian group, as stated below.

Definition 3. *Let G be a multiplicative abelian group and $g_1, g_2 \in G$ such that $\langle g_1 \rangle = G_1 \leq G, o(g_1) = n_1$; and $\langle g_2 \rangle = G_2 \leq G, o(g_2) = n_2$. Given g_1, g_2, n_1, n_2 and A, we need to find the 2-tuple (α, β) such that $g_1^{\alpha} g_2^{\beta} = A$, where $0 \leq \alpha < n_1$ and $0 \leq \beta < n_2$.*

We will also define an instance of a DLP^i solver.

Definition 4. *Let G be a group (Not necessary abelian). A DLP^i can be expressed as: $g_1^{\alpha_1} g_2^{\alpha_2} \dots g_i^{\alpha_i} = A$ where $g_1, g_2, \dots g_i, A \in G$. A DLP^i solver is an algorithm upon receiving $\{g_1, g_2, \dots g_i, A, G\}$ as an input and outputs $\{\alpha_1, \alpha_2, \dots, \alpha_i\}$, which is one possible solution that satisfies the equation if and only if such a solution exists. Otherwise, the solver aborts.*

Using Shor's algorithm [6] with quantum computers, we will be able to obtain a DLP solver (a DLP^i solver with $i = 1$) for computation. Hence, we are able to find $\alpha = dlog_g A$ for any g and A that are elements of a cyclic group, provided there is a possible solution.

In this paper, we provide such treatments by performing detailed investigations. Some preliminary concepts of group theory together with simple illustrations will be discussed in the next section.

2.2. Groups.

A **group** is a set G together with only one binary operation, typically denoted by $*$, that "behaves nicely", satisfying the following four axioms:

1. Closure: The operation never leads you outside the set. Combining any two elements in G must result in an element that is also in G, that is, $\forall a, b \in G, a * b \in G$.
2. Associativity: The way elements are grouped for the operation doesn't change the final result, that is, $\forall a, b, c \in G, (a * b) * c = a * (b * c)$.
3. Existence of Identity: There is a unique identity element denoted as e that leaves any other element unchanged when combined with it, that is, $\forall a \in G, \exists e \in G$ such that $a * e = e * a = a$.
4. Existence of Inverse: Every element a has a unique inverse element denoted a^{-1}, which, when combined with a, yields the identity element e. In mathematical notation, we have $\forall a \in G, \exists a^{-1} \in G$ such that $a * a^{-1} = a^{-1} * a = e$.

Throughout this work, we focus on multiplicative groups where the operation $*$ is multiplication, denoted as $a \cdot b$, or simply as ab; and the multiplicative identity is 1. The structure of groups provides the controlled algebraic environment necessary for many cryptographic constructions. The number of elements in a group G is called the order of G, and is denoted as $|G|$. When $|G|$ is finite, G is called a finite group. Since the elements of G play an important role, we also have an analogous definition for the order of an element in G. The order of an element $g \in G$, denoted as $o(g)$ or $|g|$, is the smallest positive integer n such that $g^n = 1$.

When all elements in G satisfy the commutative law, that is, $\forall a, b \in G, ab = ba$, then G is called an abelian group. In abelian groups, elements interact predictably, making them convenient for designing protocols, but this also makes many problems easier to attack. In this paper, we mainly deal with abelian groups. A well-known family of abelian groups is the class of cyclic groups, each of which is generated by a single element g. Notationally, if G is cyclic of order p, then $G = \langle g \mid g^p = 1 \rangle$. This is the natural home of the discrete logarithm problem. Even when we work inside a larger abelian group, discrete logarithms always "live" inside some cyclic subgroup generated by g. A subset $H \subseteq G$ is a subgroup,

denoted $H \leq G$, if H itself forms a group under the operation inherited from G. An important subgroup is the normal subgroup. A subgroup H is a normal subgroup, denoted $H \trianglelefteq G$, if it is invariant under conjugation, that is, $gHg^{-1} = H \quad$ for all $g \in G$.

We note that in abelian groups, every subgroup is normal. Normal subgroups are essential because they allow us to form quotient groups.

To make our discussions more effective, we define the following concepts of independence.

Definition 5. *Let G be a group (not necessarily abelian). Suppose G_1 and G_2 are two cyclic subgroups of G. If the generator $g_1 \in G_1$ can be written in the form $g_1 = g_2^k$ for some $k \in \mathbf{Z}$, where g_2 is a generator of G_2, then we say that G_1 and G_2 are dependent. Otherwise, we say that G_1 and G_2 are independent.*

Remark 1. *We take note that from Definition 2.2, if $G_p \leq G_{q,p}$, then we will show in the next section that the result is a very easy consequence. Thus, to ensure the hardness of DDLP in Definition 2, we need to ensure that G_p and $G_{q,p}$ are independent.*

Illustration 1. We follow the conventional approach to finding discrete logarithms in number systems that make use of primitive elements. Our illustration uses a primitive element to form a group, illustrating the concept of the DLP. Let $p = 7$. Clearly, 3 and 5 are primitive elements modulo 7. Note that we use the equality and congruence symbols interchangeably. The intended meaning of the symbols should be clear from context. Suppose that we consider the primitive element $g = 3$ modulo 7. The following computations verify that 3 is a primitive element.

Table 1. Powers of a Primitive Element $g = 3$ and Non-primitive Element $g = 2$ modulo 7.

α	$3^\alpha = A \pmod p$	$2^\alpha = A \pmod p$
0	$3^0 = 1 \pmod 7$	$2^0 = 1 \pmod 7$
1	$3^1 = 3 \pmod 7$	$2^1 = 2 \pmod 7$
2	$3^2 = 2 \pmod 7$	$2^2 = 4 \pmod 7$
3	$3^3 = 6 \pmod 7$	$2^3 = 1 \pmod 7$
4	$3^4 = 4 \pmod 7$	$2^4 = 2 \pmod 7$
5	$3^5 = 5 \pmod 7$	$2^5 = 4 \pmod 7$
6	$3^6 = 1 \pmod 7$	$2^6 = 1 \pmod 7$

From Column 2 of Table 1, we see that the choice of primitive element ensures that all powers of 3 modulo 7 are distinct and generate all elements of $\mathbf{Z}_7^*$. Based on this fact, we can take $G = \langle 3 \rangle = \{3^0, 3^1, 3^2, 3^3, 3^4, 3^5\} = \{1, 2, 3, 4, 5, 6\}$, which is a cyclic group of order 6. Next, we consider $U(7) = \{x \in \mathbf{Z}_7 : gcd(x, 7) = 1\} = \{1, 2, 3, 4, 5, 6\}$. Thus, we see that $G \cong U(7)$ under multiplication. In contrast, Column 3 illustrates that 2 is not a primitive element modulo 7 because 2 will not generate the group G, but instead generates a subgroup of G of order 3.

First, we illustrate how the DLP in a subgroup H of G is related to the DLP in G.

Let us consider the subgroup $H = \langle 2 \rangle = \{2, 2^2 = 4, 2^3 = 1\}$ modulo 7, that is, $H = \{1, 2, 4\} \leq G$. Suppose $2^2 = 4 \pmod 7$, that is, $2 = dlog_2(4) \pmod 7$, this is equivalent to the existence of DLP in G. Since $H = \langle 2 \rangle \leq G = \langle 3 \rangle$, then $2 \in \langle 3 \rangle$, and so $2 = 3^t$ for some t. Therefore, from $2^2 = 4 \pmod 7$, we have $(3^t)^2 = 4 \pmod 7$, that is, $3^{2t} = 4 \pmod 7$, that is, $2t = dlog_3(4) \pmod 7$. Or more generally, we see that $dlog_2(4)t = dlog_3(4)$. Thus, from Table 1, we can easily compute $t = 2$. These computations give us an idea that obtaining a

solution for $dlog$ in G is equivalent to finding the relationship between the generator of H and the generator of G.

2.3. Quotient groups and homomorphisms.

Suppose G_1 and G_2 are two groups (not necessarily abelian). A group homomorphism is a map $\phi : G_1 \to G_2$ between two groups that preserves the group operation, that is, $\phi(ab) = \phi(a)\phi(b)$ for all $a, b \in G_1$. Without loss of generality, we assume both G_1 and G_2 are multiplicative groups in the previous equation. The concept of homomorphisms provides a way to relate the algebraic structure of different groups.

The most important homomorphism for this work is the natural homomorphism $\theta : G \to G/N$, which makes use of the quotient group G/N. If $N \trianglelefteq G$, then the quotient group, G/N, is the set of all **cosets** of N in G, that is, $G/N = \{g + N \mid g \in G\}$. Note that the operation in G/N is defined by combining coset representatives as $(a + N) \cdot (b + N) = ab + N$. Under this operation, G/N will form a group known as the quotient group. The quotient group essentially captures the structure of G after factoring out, collapsing, or projecting onto the subgroup N. This tool allows us to simplify multi-generator DLPs by eliminating a certain number of generators.

Illustration 2. Consider another subgroup $K = \langle 6 \rangle = \{1, 6\} \leq G$ from Illustration 1. Then, the quotient group is $G/K = \{K, 2K, 3K\}$, where $2K = \{2, 5\}$ and $3K = \{3, 4\}$. Suppose $\theta : G \to G/K$ is the natural homomorphism. Then $\theta(1) = \theta(6) = K$, $\theta(2) = \theta(5) = 2K$, and $\theta(3) = \theta(4) = 3K$.

Let us consider $3^5 = 5 \pmod 7$, that is, $5 = dlog_3(5)$. Apply θ to obtain $\theta(3^5) = \theta(5)$, that is, $\theta(3)^5 = \theta(5)$, that is, $(3K)^5 = 2K$, that is, $5 = dlog_{3K}(2K)$ in G/K. Similarly, we see that $\theta(4)^5 = \theta(2)$ which gives $5 = dlog_{3K}(2K)$ in G/K. From this example, we see that a $dlog$ in the quotient group G/K corresponds to $|K|$ distinct $dlog$ in G. This is related to the fact that θ is not injective. Therefore, when working over natural homomorphisms, we need to carefully take care of the non one-to-one property.

Illustration 3. To illustrate the DLP^2 in a group that must be generated by multiple elements, we use the multiplicative group $G = \mathbf{Z}_{35}^*$, which has order 24. We select two generators, $g_1 = 16$ and $g_2 = 13$. The subgroup $\langle 16 \rangle$ has order 3, and $\langle 13 \rangle$ has order 4. The smallest subgroup containing both is $\langle 16, 13 \rangle$, which has order $\text{lcm}(3, 4) = 12$. Note that this subgroup is not the entire group $U(35)$. Let the target element be $A = 16^1 \cdot 13^2 \pmod{35}$. The DLP^2 problem is to find (α_1, α_2) such that $9 = 16^{\alpha_1} \cdot 13^{\alpha_2} \pmod{35}$, where $0 \leq \alpha_1 < 3$ and $0 \leq \alpha_2 < 4$. Despite having two unknowns and a seemingly complex search space of $3 \times 4 = 12$ pairs, the abelian property of $\mathbf{Z}_{35}^*$ allows this problem to be reduced to two independent standard DLPs modulo 5 and modulo 7, as will be discussed in the main body of this paper. This ease of reduction highlights the vulnerability of multi-generator problems in abelian settings.

3. Cyclic Groups

In this section, we begin by exploring some results for the Discrete Logarithm Problem (DLP) and the DLP^2 within the context of cyclic groups. Recall that for DLP, given a group G (not necessarily abelian) and two elements $g, A \in G$, the DLP is the problem of finding an exponent α such that $g^\alpha = A$. For a solution α to exist, the target element A must belong to the cyclic subgroup generated by g. Since $A \in \langle g \rangle$, we can immediately restrict our focus to this subgroup, $\langle g \rangle$. By its very definition, $\langle g \rangle$ is a cyclic group where g is the natural generator. Therefore, when tackling the DLP, we always implicitly assume that the problem is being solved within the cyclic group $\langle g \rangle$. However, if $A \notin \langle g \rangle$, then we can just declare

that no solution exists. This is closely linked to a more general form of DLP, called the generalized DLP, which is given as follows:

Definition 6. *Given $g, h \in G$, compute $dlog_g(h)$ if $h \in \langle g \rangle$. Otherwise, output $h \notin \langle g \rangle$.*

This extended form of the DLP confirms that the identification of subgroups in a given group is linked to the DLP in a very natural way. We derive a simple illustration for this extension.

Lemma 1. *Suppose $H \leq G$, where G is a cyclic group of order n. Solving the DLP in H is equivalent to solving the DLP in G.*

Proof. If n is prime, then the only subgroups of G are either the trivial subgroup $H = \{1\}$ or the proper subgroup $H = G$. Thus, we may assume that G is a cyclic group of composite order.

Suppose $H \leq G$ and $|H| = p$, where p is a prime such that $p \mid n$. Then, H is a prime order cyclic subgroup of G. Without loss of generality, we write $H = \langle h \mid h^p = 1 \rangle$. Assume that we can solve DLP in G. Then, given the instance $g^\alpha = A, g, A \in G, \alpha \in \mathbf{Z}_n$, we can find $\alpha = dlog_g(A)$. Now, given the instance of DLP in H, $h^\beta = B, h, B \in H, \beta \in \mathbf{Z}_p$. Since $h \in H \leq G$ and G is cyclic, then $h = g^k$ for some $k \in \mathbf{Z}$. Thus, $(g^k)^\beta = B$ implies $g^{k\beta} = B$. Compare B with A as A is publicly known to obtain $g^{k\beta} = g^\alpha$ implies $k\beta = \alpha$, and so $dlog_h(B) = \beta = \frac{\alpha}{k} = \frac{dlog_g(A)}{k}$. Therefore, we can find $dlog_h(B)$ since k is invertible modulo p, and solve the DLP in H. Take note that, if no such equality, we abort, but since G is finite and closed, we definitely can find such an equality.

Conversely, suppose we can solve DLPs in H. Then, given the instance $h^\beta = B, h, B \in H, \beta \in \mathbf{Z}_p$, we can find $\beta = dlog_h(B)$. Next, given the instance of DLP in G, $g^\alpha = A, g, A \in G, \alpha \in \mathbf{Z}_n$. If $g \in H$, then $g = h^t$, and so $h^{t\alpha} = A$. Thus, we proceed as previous paragraph to obtain $dlog_g(A) = \alpha = \frac{\beta}{t} = \frac{dlog_h(B)}{t}$, which solves the DLP in G. On the other hand, if $g \notin H$, then $g \in G - H$, and so $g \in bH$, a coset of H in G with coset leader $b \in G$. Thus, $g = bh'$ for some $h' \in H$ and since H is cyclic, we can write $h' = h^\theta, \theta \in \mathbf{Z}_p$. Since $g^\alpha = A$, then we have $h^{\theta\alpha} = b^\alpha A, h \in H \leq G, b^\alpha A \in G$. Similarly, we compare $b^\alpha A$ with B to get $dlog_g(A) = \alpha = \frac{\beta}{\theta} = \frac{dlog_h(B)}{\theta}$, and so we can solve DLP in G. □

The group G used in the conventional DLP is a cyclic group of order p, where p is an odd prime. Thus, the flexibility of solving DLP via group and subgroups will not give any impact to the conventional DLP defined over cyclic groups of prime order. However, if we generalize the DLP to a cyclic group of non prime order, that is, of composite order, then the lemma above is relevant. Next, we look at the case when the order of G is aa product of distinct primes.

Theorem 1. *Suppose G is a cyclic group of order $p_1 p_2 \dots p_r$, where $p_1, p_2, \dots, p_r$ are distinct primes. Solving the DLP in G is equivalent to solving the DLP in a subgroup H of G.*

Proof. Suppose we can feasibly solve the DLP using a DLP solver for a cyclic group $G = \langle g \rangle$ for any generator $g \in G$ of order $n = p_1 p_2 \dots p_r$. To solve the DLP in a subgroup H of G, given that $h^\beta = B$ for some $h \in H$, we need to find β. Since $h \in G$, then we can write $h = g^\alpha$ for some α. Thus, we can substitute h into the equation, $(g^\alpha)^\beta = B$ to obtain $g^{\alpha\beta} = B$ and so $\alpha\beta = dlog_g B$. Let $d = gcd(\alpha, n)$, then we can recover $\beta = (\frac{\alpha}{d})^{-1} \frac{dlog_g B}{d}$, where the inverse of $(\frac{\alpha}{d})$ exists modulo n/d.

Conversely, suppose we can feasibly solve the DLP using a DLP solver in every subgroup of $G = \langle g \rangle$. If we can feasibly factorize n, the Pohlig-Hellman algorithm can be used to break down the DLP in G into smaller DLPs in subgroups of G as shown below:

Let $n = p_1^{e_1} p_2^{e_2} \cdots p_r^{e_r}$, where $p_1, p_2, \ldots, p_r$ are distinct odd primes. We note that Pohlig-Hellman is efficient if all the prime factors are small, even if the DLP is hard. For each $i \in \{1, \ldots, r\}$, we set $g_i := g^{\frac{n}{p_i^{e_i}}}$. By Lagrange's Theorem, the element $g_i \in G$ has order $p_i^{e_i}$. Next, we set $A_i := A^{n/p_i^{e_i}}$ for some $A \in G$, and for $\alpha_i \in \{0, \ldots, p_i^{e_i} - 1\}$, we compute $g_i^{\alpha_i} = A_i$ in the cyclic subgroup $\langle g_i \rangle$. In the end, we obtain a set of linear congruences

$$\begin{aligned} \alpha &\equiv \alpha_1 \pmod{p_1^{e_1}} \\ \alpha &\equiv \alpha_2 \pmod{p_2^{e_2}} \\ &\vdots \\ \alpha &\equiv \alpha_r \pmod{p_r^{e_r}}. \end{aligned}$$

We can use the Chinese Remainder Theorem to combine the congruences for a unique solution $\alpha \pmod{n}$ which is a solution of $g^\alpha = A$. Therefore, the converse of the theorem will follow directly by setting $e_1 = e_2 = \cdots = e_r = 1$. □

If we assume that we can feasibly solve the DLP in at least one but not all subgroups of G, and also if we can feasibly factorize n, then the ability to solve the DLP in any single subgroup does not allow us to solve the DLP in group G, but it is able to narrow down the search for α due to knowledge of $\alpha \equiv \alpha_i \pmod{d_i}$ for some $i \in \{1, \ldots, \tau(n)\}$, where $d_i \mid n$ and $\tau(n)$ denotes the number of divisors of n.

Next, let us investigate the behavior of the DLP^2 in a cyclic group G. The DLP^2 is defined as finding a pair of exponents (α, β) such that: $g_1^\alpha g_2^\beta = A$, where $g_1, g_2, A \in G$. If a solution exists, the target element A must belong to the subgroup H generated by g_1 and g_2, i.e., $H = \langle g_1, g_2 \rangle$. Since H is a subgroup of the cyclic group G, it follows that H must also be a cyclic subgroup. The fact that H is cyclic allows us to show that the DLP^2 can be efficiently solved by transforming it into multiple DLPs.

Theorem 2. *A DLP^2 in a cyclic group G can be reduced to and solved by solving multiple DLPs in G.*

Proof. We want to find integers α, β such that $g_1^\alpha g_2^\beta = A$. Let $n_1 = o(g_1)$ and $n_2 = o(g_2)$ be the orders of g_1 and g_2, respectively. Without loss of generality, we might assume the order of the cyclic subgroup G is $n_1 n_2$. We construct a generator of G to be $g = g_1^{n_2} \cdot g_2^{n_1}$, and we can express all base and target elements in terms of g as follows:

$$g^a = g_1, \qquad g^b = g_2, \qquad g^\gamma = A.$$

This involves solving two or three standard DLPs, we can use a DLP solver to solve for a, b, γ:

$$a = dlog_g g_1, \qquad b = dlog_g g_2, \qquad \gamma = dlog_g A,$$

Upon substituting back into the original DLP^2 equation, we have $g_1^\alpha g_2^\beta = A$, and so $(g^a)^\alpha (g^b)^\beta = g^\gamma$ which implies $g^{a\alpha + b\beta} = g^\gamma$. Since g has order $|G|$, we equate the exponents modulo $|G|$ to obtain a linear congruence for the unknowns α and β,

$$a\alpha + b\beta \equiv \gamma \pmod{|G|}.$$

A solution for the linear congruence is guaranteed to exist if and only if $\gcd(a, b, |G|)$ divides γ, which is satisfied if a solution (α, β) exists. By finding all pairs (α, β) satisfying this congruence, we solve the DLP^2. □

In conclusion, the DLP^2 in a cyclic group is reduced to solving multiple DLPs followed by solving a Linear Diophantine Equation (the congruence). This means that a DLP^2 in a cyclic group is computationally equivalent to the standard DLP, and it is not a fundamentally harder problem.

Remark 2. *(Case where g_1 or g_2 is a generator of G): If one of the bases, say g_1, is already a generator of $G = \langle g_1, g_2 \rangle$, the general method still holds, but simplifies significantly. The order of the group is simply $o(G) = o(g_1)$. The constructed generator g will be g_1 itself, making $a = dlog_g g_1$ equal to 1. The final congruence simplifies to:*

$$\alpha + b\beta \equiv \gamma \pmod{o(g_1)}$$

where $b = dlog_{g_1} g_2$ and $\gamma = dlog_{g_1} A$. Then, we can simply find any possible solution by choosing any $\beta \in \mathbf{Z}_{o(G)}$ and obtain the corresponding $\alpha \equiv \gamma - b\beta \pmod{o(G)}$.

The difficulty of the Discrete Logarithm Problem is inherently tied to the group's structure and is preserved under fundamental group-theoretic mappings.

Theorem 3. *Suppose G_1 and G_2 are two finite multiplicative cyclic groups. Then, the DLP is preserved under any group isomorphism.*

Proof. Let $\phi : G_1 \to G_2$ be an isomorphism from G_1 onto G_2, and let $g_1^\alpha = A_1$ be a DLP instance in G_1. Applying the isomorphism ϕ to both sides yields $\phi(g_1^\alpha) = \phi(A_1)$. Using the property of isomorphism, we have $\phi(g_1)^\alpha = \phi(A_1)$. Defining $g_2 = \phi(g_1)$ and $A_2 = \phi(A_1)$, we obtain the equivalent DLP instance $g_2^\alpha = A_2$ in G_2. Thus, any solution α in G_1 is also a solution in G_2. The reverse direction follows by applying the inverse isomorphism ϕ^{-1}. □

The following theorem shows the relationship between a group and its quotient subgroup.

Theorem 4. *Suppose G is a cyclic group and N is a subgroup of G. If the DLP is solvable in G, then the DLP is also solvable in the quotient group G/N.*

Proof. We take note that the quotient group G/N is defined by the set of all cosets $\{g + N \mid g \in G\}$, with operation $(a + N)(b + N) = ab + N$. Then we can define the natural homomorphism $\theta : G \to G/N$ as $\theta(g) = g + N$.

If we have a solvable DLP instance $g^\alpha = A$ in G, applying the homomorphism θ produces an equivalent instance in G/N as shown below:

$$\theta(g^\alpha) = \theta(A)$$
$$\theta(g)^\alpha = \theta(A)$$
$$(g + N)^\alpha = A + N$$

This is a DLP in G/N with base $g + N$ and target element $A + N$, which is equivalent to show that $\alpha = dlog_{g+N}(A + N)$. Since G is cyclic, the exponent α obtained by solving the DLP in G also solves the corresponding DLP in G/N. □

4. Abelian Groups

In this section, we investigate the discrete logarithm problem in general finite abelian groups. Unlike cyclic groups, which have a single generator, abelian groups may require several generators and therefore look "multi-dimensional." However, their internal structure is highly rigid, and this rigidity turns out to collapse every multi–generator discrete logarithm problem back into the ordinary DLP. We begin with the structural decomposition of finite abelian groups, followed by an intuitive picture of what happens with two generators, and then the formal reductions showing that DLP^2 is not harder than DLP.

4.1. Decomposition of finite abelian groups

Every finite abelian group has a very rigid internal structure. Although elements may look complicated when written in product form, e.g. $(g_1^{a_1} g_2^{a_2} \cdots)$ using several generators, the group itself always breaks apart into simpler cyclic building blocks. This fact is central to understanding how "multi-generator discrete logarithm problems" behave.

The most convenient form for our purposes is the *invariant factor* decomposition. Suppose G is an abelian group. Then, $G \cong \mathbf{Z}_{n_1} \times \mathbf{Z}_{n_2} \times \cdots \times \mathbf{Z}_{n_k}$ such that $n_1 \mid n_2 \mid \cdots \mid n_k$. Each $\mathbf{Z}_{n_i}$ is cyclic, and the divisibility condition makes this decomposition unique. This expresses G as a direct product of k independent cyclic "axes," and all exponent arithmetic inside G splits accordingly. When we study discrete logarithms in G, we will see that the problem always separates into k independent ordinary DLPs, one in each cyclic factor.

Equivalently, for any abelian group G, there is another equally valid but differently organized decomposition,

$$G \cong G_{p_1} \times G_{p_2} \times \cdots \times G_{p_m},$$

where each G_{p_i} is a p_i-group. Each component itself decomposes as

$$G_{p_i} \cong \mathbf{Z}_{p_i^{e_{i1}}} \times \mathbf{Z}_{p_i^{e_{i2}}} \times \cdots .$$

This version is particularly useful for classical DLP algorithms such as the Pohlig–Hellman method, which reduce DLP to its prime-power parts. However, for analyzing "multi-generator" claims such as the DLP^2 proposed by KAZ, the invariant factor form is conceptually cleaner.

Suppose that there is an element of G which can be written as $x = (x_1, x_2, \ldots, x_k)$ such that $x_i \in \mathbf{Z}_{n_i}$, and a base generator is written as $g = (g_1, g_2, \ldots, g_k)$, then the discrete logarithm equation $g^a = x$ is equivalent to the system $g_i^a = x_i$ in $\mathbf{Z}_{n_i}$ for each $i = 1, \ldots, k$. In other words, a single "multi-generator" DLP in a finite abelian group is always just k ordinary DLPs in the cyclic factors. This structural fact underlies many reductions later in the paper. It also explains why attempts to generalize the classical DLP to several generators (such as the DLP^2 in KAZ) must be handled with care, since the group structure itself forces the problem to decompose.

4.2. Reduction of DLP^2 to DLP

This reduction is critical for evaluating the security of schemes, such as the one proposed by KAZ, that attempt to introduce computational hardness by using multiple generators, leading to the introduction of the DLP^2. The following theorem demonstrates that this attempt to increase complexity is negated by the algebraic properties of the group.

Theorem 5. *Let G be a finite abelian group of order pq, where p, q are distinct primes. If the DLP is solvable in a cyclic subgroup of G, then the DLP^2 is also solvable in G.*

Proof. Suppose G_1 and G_2 are two subgroups of G. Without loss of generality, we may let $G_1 = \langle g_1 \rangle$ and $G_2 = \langle g_2 \rangle$. Given the DLP^2 instance in G, that is, $g_1^\alpha g_2^\beta = A$, everything depends on whether G_1 and G_2 are *dependent* (one is contained in the other) or *independent* (they contribute different "directions" inside G).

(Case 1) Dependent generators. If $G_2 \leq G_1$ or equivalently, $g_2 = g_1^k$ for some k, then

$$g_1^\alpha g_2^\beta = g_1^\alpha (g_1^k)^\beta = g_1^{\alpha + k\beta}.$$

This shows that the DLP^2 exists entirely in the cyclic group G_1. Thus the whole DLP^2 in G instance collapses into a single ordinary DLP in G_1, that is, $g_1^{\alpha+k\beta} = A$. Solving the DLP in G_1 recovers $\alpha + k\beta$, and the usual one-dimensional freedom describes all (α, β) solutions.

(Case 2) Independent generators. If G_1 and G_2 are independent, i.e. neither subgroup is contained in the other, then they span a rank-2 abelian subgroup of G. Quotienting by G_1 kills the g_1-component:

$$\phi : G \to G/G_1, \qquad \phi(x) = x + G_1.$$

Applying ϕ to the equation $g_1^\alpha g_2^\beta = A$ gives

$$(g_2 + G_1)^\beta = A + G_1,$$

where $g_2 + G_1$ is a coset of G_1. Since G has order pq with p, q distinct primes and G_1 is a nontrivial proper subgroup, the quotient group G/G_1 has prime order and is therefore cyclic (generated by $g_2 + G_1$). This is a *one-dimensional* DLP in the quotient group, so solving it yields β. Once β is known, substitute back into the original equation $g_1^\alpha = A g_2^{-\beta}$, which is a DLP inside the cyclic group G_1.

In both dependent and independent situations, the DLP^2 instance reduces entirely to ordinary DLPs in certain cyclic subgroups. □

Remark 3. *We would like to point out that the group G in Theorem 5 can be easily generalized to arbitrary order $p_1 p_2 \dots p_n$, where p_i are all distinct primes.*

On the other hand, if G is of arbitrary order with two independent subgroups G_1, G_2, such that $G_1 \cap G_2$ contains nontrivial elements, i.e. $G_1 \cap G_2 > \{1\}$, then the pair (α, β) in DLP^2 is also not unique. There is more than one set of solutions, because there exists a nontrivial element $g_0 \in G_1 \cap G_2$ such that $g_0 \neq 1$. This means there must exist non-zero integers x_1 and x_2 such that:

$$g_0 = g_1^{x_1} = g_2^{x_2}$$

and we can find both x_1 and x_2 by solving two smaller DLPs: $g_1^{x_1} = g_0$ and $g_2^{x_2} = g_0$. From this, we see that $g_1^{x_1} g_2^{-x_2} = g_0 g_0^{-1} = 1$, where 1 is the identity element. We next multiply this identity as many times into the first solution to obtain the other solutions. Assume (α_0, β_0) is one possible solution of the DLP^2, which means $g_1^{\alpha_0} g_2^{\beta_0} = A$. We multiply the equation by 1 for t times to obtain

$$g_1^{\alpha_0} g_2^{\beta_0} = A$$
$$g_1^{\alpha_0} g_2^{\beta_0} (1)^t = A(1)^t$$
$$(g_1^{\alpha_0} g_2^{\beta_0})(g_1^{x_1} g_2^{-x_2})^t = A$$

and hence we obtain

$$g_1^{\alpha_0 + x_1 t} g_2^{\beta_0 - x_2 t} = A, \quad t \in \mathbf{Z}.$$

Hence, $(\alpha, \beta) = (\alpha_0 + x_1 t, \beta_0 - x_2 t)$.

Remark 4. ***Remark on Groups of Square-Free Order*** *The following was the condition stated in KAZ for their DLP*^{2}*. They used the cyclic group* $\mathbf{Z}_n$*, the additive group modulo n, where* $n = p_1 p_2 \dots p_k$*,* p_i *are distinct primes, and n is square-free.*

The group structure can be simplified by the Chinese Remainder Theorem to form the product group: $G \cong \mathbf{Z}_{p_1} \oplus \mathbf{Z}_{p_2} \oplus \cdots \oplus \mathbf{Z}_{p_k}$.

In this specific case, the reduction to the quotient group $H = G/\langle g_1 \rangle$ *corresponds to solving a DLP in a group whose order is* $n/|\langle g_1 \rangle|$*, a divisor of N. If* $o(g_1) = p_i$*, then the reduction is to a group of order* n/p_i*. This highlights that a significant portion of the group structure is bypassed, simplifying the problem considerably.*

4.3. Reducing DLP to DLP^2

Theorem 6. *If DLP*^{2} *is solvable in G and we may choose any* g_2*, then DLP is solvable in G.*

Proof. Given $g^\alpha = A$, set $g_1 = g$ and $g_2 = 1$. Then the DLP^2 instance

$$g_1^\alpha g_2^\beta = A$$

is equivalent to the original DLP, and the solver returns α.

On the other hand, if the DLP^2 solver requires g_1 and g_2 to be independent, choose $g_2 = A$ and query the solver for

$$g_1^\alpha A^\beta = g_1.$$

This produces the congruence

$$\alpha + \beta dlog_{g_1}(A) \equiv 1 \pmod{o(g_1)},$$

from which $dlog_{g_1}(A)$ is recovered, provided $\beta \not\equiv 0$. □

Remark 5. *Thus the DLP*^{2} *and the DLP are computationally equivalent in all finite abelian groups.*

4.4. Reduction of Multi-Generator DLP^n *to Conventional DLP*

The following theorem demonstrates that due to the structure of finite abelian groups, the generalized DLP^n problem is computationally equivalent to solving multiple instances of the conventional DLP and a linear congruence.

Theorem 7. *Suppose G is a finite abelian group. Solving a DLP*^{n} *in G is not harder than solving DLPs in G, for any integer* $n \geq 1$.

Proof. Without loss of generality, we may assume that G is a multiplicative group. Given $g_1, g_2, \dots, g_n, A \in G$, the DLP^n asks for the exponents $\alpha_1, \alpha_2, \dots, \alpha_n \in \mathbf{Z}$ that satisfy the equation

$$g_1^{\alpha_1} g_2^{\alpha_2} \cdots g_n^{\alpha_n} = A \text{ in } G. \tag{1}$$

Our reduction involves 4 steps, as illustrated below.

Step 1: Group Decomposition

By the Fundamental Theorem of Finite Abelian Groups, G is isomorphic to a direct product of cyclic groups as shown below.

$$G \cong C_{m_1} \times C_{m_2} \times \cdots \times C_{m_k},$$

where C_{m_i} denotes cyclic groups of orders m_i for $i = 1, 2, \dots, k$.

Step 2: Applying the Isomorphism and Separation

Let $\phi : G \to C_{m_1} \times \cdots \times C_{m_k}$ be an isomorphism. The isomorphism ϕ distributes over group multiplication. We define the following component representations:

$$\phi(g_j) = (x_{j,1}, x_{j,2}, \ldots, x_{j,k}), \quad \text{for } j = 1, \ldots, n, \text{ and}$$
$$\phi(A) = (y_1, y_2, \ldots, y_k).$$

Applying ϕ to Equation (1) yields k independent equations, one for each cyclic component:

$$x_{1,i}^{\alpha_1} x_{2,i}^{\alpha_2} \cdots x_{n,i}^{\alpha_n} = y_i \quad \text{in } C_{m_i}, \text{ for } i = 1, \ldots, k.$$

Step 3: Reduction to Linear Congruences using DLP

Since each component C_{m_i} is a cyclic group, we select a generator h_i for C_{m_i}. We then use the DLP solver to find the discrete logarithms of all component elements with respect to the generator h_i as follows:

- Find $a_{j,i} = dlog_{h_i}(x_{j,i})$, such that $h_i^{a_{j,i}} = x_{j,i}$,
- Find $b_i = dlog_{h_i}(y_i)$, such that $h_i^{b_i} = y_i$.

Substituting the exponents into the component equation gives a linear congruence modulo the order m_i:

$$a_{1,i}\alpha_1 + a_{2,i}\alpha_2 + \cdots + a_{n,i}\alpha_n \equiv b_i \pmod{m_i}.$$

Step 4: Solving the System

Combining the k independent linear congruences results in a solvable system of equations over the exponent ring:

$$\begin{cases} a_{1,1}\alpha_1 + a_{2,1}\alpha_2 + \cdots + a_{n,1}\alpha_n \equiv b_1 \pmod{m_1} \\ a_{1,2}\alpha_1 + a_{2,2}\alpha_2 + \cdots + a_{n,2}\alpha_n \equiv b_2 \pmod{m_2} \\ \quad \vdots \\ a_{1,k}\alpha_1 + a_{2,k}\alpha_2 + \cdots + a_{n,k}\alpha_n \equiv b_k \pmod{m_k} \end{cases}$$

This system is efficiently solved using standard linear algebra techniques (e.g., Gaussian elimination) over the appropriate rings $\mathbf{Z}_{m_i}$. Since the DLP solver (Step 3) and the linear system solution (Step 4) are both assumed to be efficient, the DLP^n is efficiently solved. □

The introduction of a DLP^n is often motivated by the belief that it can inherently offer more security than the conventional DLP or a DLP^2. However, the security of these generalized variants also depends heavily on the chosen group structure and the method of attack. For example, some proposed attacks use known mathematical techniques, such as the MDS matrix approach utilized in the method by Wu and Zhuang [14] for deriving tighter bounds on the Generalized Multiple Discrete Logarithm problem. This specific technique, when applied to solve a DLP^n, may lead to better time complexities than expected, compromising the intended security gain of the DLP^n. As our reduction demonstrates, the reliance of these multi-generator discrete logarithm problems on abelian group properties ensures that the true security ceiling remains fixed by the computational cost of the single-instance DLP.

We highlighted the following remark showing that the properties off DLP in cyclic group can be easily generalized to abelian group.

Remark 6.

- *By using Theorem 3 and Step 1 in Theorem 7, we see that the result in Theorem 3 can be easily generalized to any finite abelian groups.*
- *From Theorem 4, if G is an arbitrary group (not necessary abelian), then for the result of the theorem to hold, we need to find a normal subgroup N such that G/N is cyclic.*

5. Conclusion and Future Research Direction

The foundational challenge of securing public-key cryptography against quantum computers necessitates a critical reassessment of the algebraic problems upon which current systems are built. Our work establishes that multi-generator variants of the Discrete Logarithm Problem (DLP), such as DLP^2 and the more general DLP^n, offer no enhanced security complexity when formulated within abelian groups. Regardless of the number of generators, these problems are polynomially reducible to solving multiple instances of the conventional DLP. Given that Shor's algorithm provides a polynomial-time solution for the conventional DLP, it follows that any of its variants confined to the commutative structure of abelian groups are similarly vulnerable and therefore not quantum-resistant. Continued efforts to construct quantum-resistant cryptographic primitives must move beyond the commutative structures of abelian groups.

Consequently, a promising direction for future research lies in exploring non-abelian groups, where the lack of commutativity fundamentally changes the nature of the discrete logarithm operation and potentially disrupts known quantum attack mechanisms. We have proposed a DLP^n formulation in non-abelian groups as a candidate for this exploration. An immediate next step is to analyze the hardness of this specific non-abelian problem, particularly investigating whether existing classical and quantum algorithms designed for commutative groups can be efficiently adapted. Another possible direction is to investigate the analog form of the DLP, such as the exhaustion number problem [12] or the complete decomposition problem [11], which generalize the DLP by working with sets of elements rather than just one single element. Beyond groups, algebra also offers other avenues for investigation, including hard problems defined over general semigroups or monoids. These structures, which lack properties such as inverses, could introduce interesting computational complexity suitable for new quantum-resistant cryptographic schemes.

Recent cryptographic proposals highlight the risks of neglecting the algebraic structure underlying discrete-logarithm-based assumptions. In the KAZ construction, the claimed quantum resistance of the Double Discrete Logarithm Problem appears difficult to justify, as the underlying problem remains structurally reducible to the conventional Discrete Logarithm Problem. Consequently, the security of the associated key encapsulation mechanism, key agreement, and digital signature schemes ultimately inherits the vulnerabilities of the abelian-group conventional DLP.

More broadly, cryptographic constructions seeking stronger or more robust hardness assumptions may need to completely abandon the classical finite abelian group setting altogether. Non-abelian groups such as dihedral or permutation groups can be researched in the future and explored further. However, non-abelian groups alone do not guaranty increased security. The hardness of the resulting discrete logarithm problem depends critically on small group-theoretic choices, such as the structure and size of subgroups and quotient groups, and whether they possess prime or composite-like behavior. In many cases, large cyclic subgroups remain efficiently identifiable. These structural details can substantially influence the availability of reductions and the overall security of the scheme. Another possible direction is the use of carefully constructed infinite or finitely generated groups, in which the discrete logarithm problem can be made arbitrarily hard by

design [18]. Exploring whether such constructions can allow for practical and genuinely quantum-resilient cryptographic primitives remains an important open problem.

Abbreviations

The following abbreviations are used in this manuscript:

MDPI	Multidisciplinary Digital Publishing Institute
DLP	Discrete Logarithm Problem
DDLP	Double Discrete Logarithm Problem
KAZ	Kriptografi Atasi Zarah
IND-CCA2	Indistinguishability under Adaptive Chosen-Ciphertext Attack
RSA	Rivest–Shamir–Adleman
MDS	Maximum Distance Separable
gcd	Greatest Common Divisor

References

1. Stinson, D. R.; Paterson, M. B. *Cryptography: Theory and Practice*, 4th ed.; CRC Press: Boca Raton, FL, USA, 2019. https://doi.org/10.1201/9781315282497.
2. KAZ-SIGN PQC Digital Signature Scheme. Available online: https://www.antrapol.com/KAZ-SIGN (accessed on 23 December 2025).
3. Krishnan, D.; Fu, X. Towards Zero Knowledge Argument for Double Discrete Logarithm with Constant Cost. *Theor. Comput. Sci.* **2024**, *1018*, 114799. https://doi.org/10.1016/j.tcs.2023.114799.
4. Diffie, W.; Hellman, M. E. New Directions in Cryptography. *IEEE Trans. Inf. Theory* **1976**, *22* (6), 644–654. https://doi.org/10.1109/TIT.1976.1055638.
5. ElGamal, T. A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. *IEEE Trans. Inf. Theory* **1985**, *31* (4), 469–472. https://doi.org/10.1109/TIT.1985.1057074.
6. Shor, P. W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM J. Comput.* **1997**, *26* (5), 1484–1509. https://doi.org/10.1137/S0097539795293172.
7. Pohlig, S. C.; Hellman, M. E. An Improved Algorithm for Computing Logarithms over GF(p) and Its Cryptographic Significance. *IEEE Trans. Inf. Theory* **1978**, *24* (1), 106–110. https://doi.org/10.1109/TIT.1978.1055817.
8. Lido, G. A Provably Quasi-Polynomial Algorithm for the Discrete Logarithm Problem in Finite Fields of Small Characteristic. *Finite Fields Their Appl.* **2025**, 102753. https://doi.org/10.1016/j.ffa.2025.102753.
9. Tanujaya, N. A.; Thendito, J. G.; Faisal, N.; Yusol, Y. Comparative Efficiency of Algorithms for Solving the Elliptic Curve Discrete Logarithm Problem. *Procedia Comput. Sci.* **2025**, *269*, 1348–1357. https://doi.org/10.1016/j.procs.2025.09.076.
10. Tripathi, S. K.; Gupta, B.; Pandian, K. K. S. An Alternative Practical Public-Key Cryptosystem Based on the Dependent RSA Discrete Logarithm Problem. *Expert Syst. Appl.* **2021**, 114047. https://doi.org/10.1016/j.eswa.2020.114047.
11. Wong, D. C. K. Complete Decomposition of Dihedral Groups with Application to Key Exchange Protocol. *Asian-Eur. J. Math.* **2023**, *16* (5), 2350074. https://doi.org/10.1142/S1793557123500742.
12. Wong, D. C. K.; Wong, K. W.; Yap, W. S. Exhaustion 2-Subsets in Dihedral Groups of Order 2p. *Asian-Eur. J. Math.* **2018**, *11* (3), 1850047. https://doi.org/10.1142/S179355711850047X.
13. Wu, H.; Zhuang, J.; Duan, Q.; Zhu, Y. Non-Uniform Birthday Problem Revisited: Refined Analysis and Applications to Discrete Logarithms. *Inf. Process. Lett.* **2021**, *175*, 106225. https://doi.org/10.1016/j.ipl.2021.106225.
14. Wu, H.; Zhuang, J. Tighter Bound for Generalized Multiple Discrete Logarithm Problem via MDS Matrix Method. *Inf. Process. Lett.* **2024**, *188*, 106535. https://doi.org/10.1016/j.ipl.2024.106535.
15. Ying, J. H. M.; Kunihiro, N. Bounds in Various Generalized Settings of the Discrete Logarithm Problem. In *Applied Cryptography and Network Security (ACNS 2017)*; Gollmann, D., Miyaji, A., Kikuchi, H., Eds.; Lecture Notes in Computer Science, Vol. 10355; Springer: Cham, Switzerland, 2017; pp. xxx–xxx. https://doi.org/10.1007/978-3-319-61204-1_25.
16. Yun, A. Generic Hardness of the Multiple Discrete Logarithm Problem. In *Advances in Cryptology – EUROCRYPT 2015*; Oswald, E., Fischlin, M., Eds.; Lecture Notes in Computer Science, Vol. 9057; Springer: Berlin, Heidelberg, Germany, 2015; pp. xxx–xxx. https://doi.org/10.1007/978-3-662-46803-6_27.
17. Shoup, V. Lower Bounds for Discrete Logarithms and Related Problems. In *Advances in Cryptology – EUROCRYPT '97*; Fumy, W., Ed.; Lecture Notes in Computer Science, Vol. 1233; Springer: Berlin, Heidelberg, Germany, 1997; pp. xxx–xxx. https://doi.org/10.1007/3-540-69053-0_18.

18. Battarbee, C.; Darbinyan, A.; Kahrobaei, D. Hard Instances of Discrete Logarithm Problem and Cryptographic Applications. *Cryptology ePrint Archive*, Paper 2025/1461. Available online: https://eprint.iacr.org/2025/1461.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.