論 文 の 内 容 の 要 旨

論文題目　The Computational Complexity in Various Settings of Cryptographic Primitives
（暗号プリミティブにおける様々な状況下での計算困難性）

氏　　名　　イン ホェイ ミン ジェイソン

This thesis is broadly segmented into three main parts. The first examines the generic hardness of the generalized multiple discrete logarithm problem (GMDL), where the solver has to solve k out of n instances for various settings of the discrete logarithm problem. The second describes methods of solving certain parameters of the discrete logarithm problem with low Hamming weight product exponents. These can be applied to obtain improved attacks pertaining to the GPS identification scheme. The third relates to user authentication based on password recovery via rainbow tables. The work here demonstrates how rainbow tables can be designed to recover commonly used passwords generated in the process as well as its improvements over conventional methods.

Discrete logarithms arise in many aspects of cryptography. The hardness of the discrete logarithm problem is central in many cryptographic schemes; for instance in signatures, key exchange protocols and encryption schemes. In fact, many variants of the discrete logarithm problem have evolved over the years. Some of these include the Bilinear Diffie-Hellman Exponent Problem, the Bilinear Diffie-Hellman Inversion Problem, the Weak Diffie-Hellman Problem and the Strong Diffie-Hellman Problem.

Cryptographic constructions based on the Discrete Logarithm Problem (DLP) are applied extensively. For instance, an early application of the DLP in cryptography came in the form of the Diffie-Hellman key exchange protocol for which the security is dependent on the hardness of the DLP. Among some of the others include the ElGamal encryption and signature schemes as well as Schnorr's signature scheme and identification protocol. The multiple discrete logarithm problem mainly arises from elliptic curve cryptography. NIST recommended a small set of fixed (or standard) curves for use in cryptographic schemes to eliminate the computational cost of generating random secure elliptic curves. The implications for the security of standard elliptic curves over random elliptic curves were analysed based on the efficiency of solving multiple discrete logarithm problems.

In a generic group, no special properties which are exhibited by any specific groups or their elements are assumed. Algorithms for a generic group are termed as generic algorithms. There are a number of results pertaining to generic algorithms for DLP and k-MDL (i.e. k instances of the DLP). Shoup showed that any generic algorithm for solving the DLP must perform $\Omega(\sqrt{p})$ group operations. There are a few methods for computing discrete logarithm in approximately $\sqrt{p}$ operations. For example, Shanks Baby-Step-Giant-Step method computes the DLP in $\tilde{O}(\sqrt{p})$ operations. One other method is the Pollard's Rho Algorithm which can be achieved in $O(\sqrt{p})$ operations. Since then, further practical improvements to the Pollard's Rho Algorithm have been proposed but the computational complexity remains the same. There exist index calculus methods which solve the DLP in subexponential time. Some of these more recent works include finite fields of small characteristic as well as finite fields of medium to high characteristic. However, such index calculus methods are not relevant in our context since they are not applicable for a generic group. Hence, currently known generic algorithms are asymptotically optimal.

An extension of Pollard's Rho algorithm was proposed by Kuhn and Struik which solves k-MDL in $O(\sqrt{kp})$ group operations if $k \leq O(\sqrt[4]{p})$. It was subsequently shown that $O(\sqrt{kp})$ can in fact be achieved without the imposed condition on k. The former's method of finding discrete logarithm is sequential in the sense that they are found one after another. However in the latter's method, all the discrete logarithms can only be obtained towards the end. Finally, it was presented by Yun that any generic algorithm solving k-MDL must require at least $\Omega(\sqrt{kp})$ group operations if $k = o(p)$.

In the context of our work, suppose an adversary has knowledge or access to many instances of the discrete logarithm problem either from a generic underlying algebraic group or from a standard curve recommended by NIST. Our work investigates how difficult it is for such an adversary to solve subcollections of those instances. One of our result outcomes in this work shows that an adversary gaining access to additional instances of the DLP provides no advantage in solving some subcollection of them when k is small and for corresponding small classes of n. Our techniques are also applicable to other standard non-NIST based curves. For instance, the results in this work are relevant to Curve25519 which has garnered considerable interest in recent years. Furthermore, we also establish formal bounds for the generic hardness of solving the GMDL problem for larger k values. As a corollary, these results provide the lower bounds of solving the GMDL problem for the full possible range of inputs k. Part of this work can be viewed as a generalization of the results by Yun (EUROCRYPT '16).

More specifically, we introduce two techniques to solve such generalized multiple discrete logarithm problems. The first we refer to as the matrix method which is also shown to achieve asymptotically tight bounds when the inputs are small. We also analyse its trade-off efficiency when the sizes of matrices involved are varied. The second technique is referred as the block method which can be applied for larger inputs. We also show that the block partitioning in this method is optimized. Moreover, when n is relatively small with respect to k, the bounds that are obtained in this way are also asymptotically tight. Furthermore, we demonstrate that the block method can be adapted and applied to generalized versions of other discrete logarithm settings to also obtain generic hardness bounds for such problems. For instance, part of this work also shows that solving one out of n instances of the Discrete Logarithm Problem with Auxiliary Inputs is as hard as solving a single given instance when n is not too large. In addition, we also explain why the matrix method cannot be extended to solve these problems.

One aspect of practical cryptosystems lies in the implementation efficiency in which exponentiations can be carried out. Agnew *et al*. proposed the usage of low Hamming weight integers as secret exponents in order to achieve faster implementations. This speeds up computations since the number of multiplications for exponentiations depends on the Hamming weight of the exponent.

The GPS identification scheme is an interactive protocol between a prover and a verifier. It was introduced by Girault and later shown to be secure. This protocol is applicable for usage in low cost chips as the computational cost required by the prover is relatively low. Nevertheless, every operation incurred is still significant for low cost chips, like RFID tags. As such, Girault and Lefranc proposed that the private key exponent to be the product of two integers with low Hamming weight, thereby reducing the online computational cost. More specifically the private key was chosen such that the private key is a product of a 142-bit number with 16 random bits equal to 1 chosen among the 138 least significant ones and a 19-bit number with 5 random bits equal to 1 chosen among the 16 least significant ones (from which we henceforth refer to as GL parameters). In the same paper, it was computed that these parameters are not susceptible to a routine attack by exhaustive search.

Subsequent work by Coron, Lefranc and Poupard demonstrated a method of attack via Coppersmith's splitting system of the GL parameters with lower complexity than routine exhaustive search. As a result, they instead proposed a different set of parameters; namely that the private key be a product of a 30-bit number with 12 nonzero bits and a 130-bit number with 26 nonzero bits (from which we henceforth refer to as CLP parameters). Moreover, they show that their line of attack is not effective against the CLP parameters.

Parameterized splitting system was first introduced by Kim-Cheon and can be regarded as a generalization of Coppersmith's splitting system. Using this tool, they demonstrated an improved attack (with regards to speed) on the CLP parameters. They later further improve this attack over the previous work by applying a refinement. Thus far, this is the current fastest known attack of the GPS identification scheme with CLP parameters.

Our work highlights general methods of solving various DLP with low Hamming weight product (LHWP) exponents by providing improved results for certain settings of the parameterized splitting system. We introduce the concept of parameters dependent splitting system which served as tools to solve such problems more efficiently. Moreover, we show that the GPS identification scheme utilizing CLP parameters satisfy such settings. There are two significant results that arise from this work. The first provides an improved attack on the GPS scheme with lower time over the most recent state of the art without any increment in memory. The second result shows for the first time that the GPS scheme can be attacked in time complexity of under 2^{64} with a slight increase in memory requirement over the former.

Furthermore, we also prove that the settings required to apply our improved parameterized system are not restrictive but in fact that the set of admissible applicable values increases when the splitting sizes are further apart.

Overall, this work also serves to provide a general framework on the type of parameters suitable for consideration in future design of cryptosystems based on DLP with LHWP exponents.

One-way functions have important relevance in the context of passwords for user authentication purposes. For instance, storing user passwords in plaintext will result in a breach of security should the

password file be compromised. In particular, cryptography hash functions are employed to store the hash digests of passwords instead. The authentication process then compares the stored hash with the hash of a user's input.

There are various ways to invert such hash functions to recover the plaintext password. Some of these methods include exhaustive search, precomputations and time-memory trade-offs. In exhaustive search, every feasible password combination is searched until the correct one is obtained. In precomputations, a large table of all feasible plaintext passwords and their corresponding hashes are stored. Due to the large password space, both of these methods suffer the drawbacks of requiring long online computation time and large storage space for exhaustive search and precomputations respectively.

To alleviate such issues, cryptanalytic time-memory trade-off was introduced by Hellman. This is a method to invert generic one-way functions by utilizing a trade-off between time and memory cost. Such technique can be applied in password recovery by inverting the relevant hash function. Time-memory trade-off techniques have also been applied in practical attacks such as A5/1 and LILI-128. Rainbow tables were introduced by Oechslin which provided certain improvements over the original method by Hellman. The main difference lies in the usage of different reduction functions when generating a table during the offline phase. They have been shown to be efficient in recovering LM hash passwords as well as UNIX passwords.

Our work is motivated by the fact that distributions of passwords tend to heavily skewed. Indeed, recent surveys indicate that top frequently used user passwords constitute a significant proportion of the database. Such common passwords also tend to be identical across various databases. In this thesis, we wish to incorporate such frequently used passwords in rainbow tables generated to ensure that they can be recovered during the online phase. A canonical method of incorporating such passwords involves assigning passwords at the start points of rainbow chains. We show that it is possible to incorporate multiple passwords across a single chain instead. This is an extension of previously known results where only 3 or 4 passwords assigned in a chain were analyzed. Furthermore, we prove that this method of incorporating frequently used passwords provide a faster recovery time during the online phase as opposed to the natural way of assigning them at the start of each chain. We also include results for typical rainbow change lengths in practical settings.