Proceedings of the 3rd International Conference on Computing Innovation and Applied Physics
DOI: 10.54254/2753-8818/13/20240844

The application of group theory behind modern cryptography

Tianyi Wang[1, 3] and Zetong Xu[2]

[1]Shenzhen College of International Education, Shenzhen, Guangdong, 518043, China
[2]Shanghai Experimental School, Shanghai, Shanghai, 201210, China

[3]s23760.wang@stu.scie.com.cn

Abstract. The importance of cryptography and securing data has become increasingly important, and the safety of previous cryptographic methods are also being questioned. In the past few decades, with the rise of modern mathematical tools, notably group theory, cryptography has quickly advanced to more complex and safer levels. This paper will begin by recalling the definition and some terminology about groups. Then, it will summarize two of the most used cryptography systems with underlying group theory: the Diffie-Hellman Key Exchange Protocol, together with one of its variants, the Ko-Lee-Cheon-Han-Kang-Park Key Agreement, and the RSA Protocol. Also, it examines what the safety of a cryptography system means, and methods to increase the security of these protocols. When talking about the two protocols, extensive group theory is used, both as a means of operation and as a method to prove the protocol's validity. Finally, it mentions the possible directions of improvement in this field and whether these cryptographic methods are still reliable, even with the widespread use of quantum computers in the future.

Keywords: Group Theory, Cryptography, Diffie-Hellman, RSA Protocol.

1. Introduction

In the past few decades, cryptography (the study of securing information such that only the person intended can use it) has developed so much that many branches of math, such as statistics and number theory, have been extensively involved with this subject. This paper focuses on the use of group theory in two cryptographic protocols.

Cryptography means trying to communicate data safely, so from it must arise an idea: a "protocol". A protocol is an algorithm (a sequence of steps) that a computer or human can follow to transmit a message securely [1]. When talking about cryptography, there must be eavesdroppers and attackers who are trying to intercept the transmitted data, and to decipher it. This paper assumes that the "enemy" or eavesdropper will know everything used in the system and the details of the system itself. The eavesdropper will also, due to the theoretical notion about data interception, know everything that is communicated between two parties. More specifically, the IND-CCA2 specification will be considered, which determines whether a guess on the decryption result of a piece of cipher text, with the access of a decrypting machine that can decrypt any cipher text apart from the original, is significantly more often to succeed than a pure guess by chance. These are the worst-case estimates and are usually not a good representation of what happens in normal life; nevertheless, from a theoretical perspective, these requirements are still useful to compare the potential between different encryption algorithms [2].

DOI: 10.54254/2753-8818/13/20240844

A basic introduction of group theory will also be mentioned in this paper. Group theory is a branch of mathematics, mainly concerning a mathematical concept called groups. Group theory forms part of the basis of abstract algebra, and it will be introduced in part 2. They can take many forms, and some of them are very useful for encryption.

This paper will mainly talk about two encryption algorithms, the Diffie-Hellman protocol and the RSA system. The Diffie-Hellman protocol was introduced in 1976 by Whitfield Diffie and Martin Hellman and uses any suitable underlying group to generate a secret key between two parties. Many articles study the safety of a specific underlying group and proposes possible groups for this system to work well. For example, braid groups have been proposed to work securely for this system [3]. The group of points on some elliptic curves also are secure [4]. A variant of the Diffie-Hellman protocol, the Ko-Lee-Cheon-Han-Kang-Park Key Agreement Protocol is used on nonabelian (noncommutative) groups [5]. The RSA system was proposed by Ron Rivest, Adi Shamir and Leonard Adleman, and is one of the most popular examples of public key encryption algorithms, allowing anyone to send an encrypted message to a specific person, while only (s)he can decrypt these messages [6]. Group theory is used in RSA to prove Lagrange's theorem, which in turn can prove Fermat's little theorem. Fermat's little Theorem proves that RSA is valid, that after decryption the result is the same as the original text.

Soon, quantum computing will strongly affect the strength of almost all encryption algorithm, as they can give quick solutions to problems such as the factoring problem (quickly factoring a very large number into its primes) and the discrete logarithm problem (DLP). These two problems are the key to breaking many encryption algorithms: the main difficulty in attacking the Diffie-Hellman protocol is the DLP while the main difficulty for RSA is the factoring problem [7]. However, mathematicians are also proposing many possible methods to overcome quantum computing, notably WalnutDSA, a method of digital authentication (making sure a message is not changed when it is being communicated) which has been proven to be secure under quantum computing attacks [8].

2. An introduction to group theory

To understand the following two algorithms, a basic knowledge of group theory is needed. First, we will cover the definition of a group, followed by some commonly used notation, and finally we will mention some terminology which will be helpful for later understanding.

2.1. Definition of a group

A group G comprises of a set of elements S and an operation $*$ that is valid on any two members of S such that the conditions $\forall a, b \in G, a * b \in G$ (closure), $\forall a, b, c \in G, (a * b) * c = a * (b * c)$ (associativity), $\exists e \in G$ s.t. $\forall a \in G, a * e = e * a = a$ (the existence of a group identity) and $\forall a \in G$, $\exists b \in G$ s.t. $a * b = b * a = e$ where e is the identity element (the existence of inverses for every element) are satisfied, and G is written as $G = \langle S, * \rangle$. (An operation $*$ is a function that takes two inputs and return its result; an operation on a and b is written $a * b$ or ab when the context is clear.)

2.2. Group notation and terminology used in this paper

The identity element will always be referred to as e in this paper. The inverse of a in a group is written as a^{-1}, and $a * a * \ldots * a$ (n a's being operated together for $n \geq 0$) is written as a^n; this can be called a power. The order of a group $G = \langle S, * \rangle$ is the number of elements in S, written $|G|$. The order of an element a in a group is the minimum $n \geq 1$ such that $a^n = e$, denoted $\text{ord}(a) = n$. When there are two groups $G = \langle S, * \rangle$ and $H = \langle T, * \rangle$ and $T \subseteq S$, H is called a subgroup of G, which can be written as $H \leq G$. An abelian group (commutative group) is a group for which $\forall a, b \in G, a * b = b * a$. A cyclic group is a group for which $\exists g \in G$ s.t. $\forall a \in G, g^n = a$ ($n \in \mathbb{Z}$), where g is called the generator of the group (it can generate all the elements in the group by operating by itself). The conjugate of an element a by another element b in G is $b^{-1} * a * b$. Finally, when all elements in a subgroup are operated by another element, the set formed is called a coset (the coset may not be a group, as it may not contain the identity). When the subgroup $H = \{h_1, h_2, \ldots, h_n\}$, the left coset $aH = \{ah_1, ah_2, \ldots, ah_n\}$ while the right coset

DOI: 10.54254/2753-8818/13/20240844

$aH = \{h_1a, h_2a, \dots, h_na\}$. For abelian groups, the left cosets and right cosets are identical, so can be simply referred to as the coset.

3. The Diffie-Hellman key exchange protocol

The Diffie-Hellman Key Agreement Protocol is a method of secretly conceiving a key over a public channel, published in 1976 by Whitfield Diffie and Martin Hellman. It is one of the most widely used encryption algorithms and forms a basis for many other schemes. This set of schemes is not designed to communicate information by themselves, but rather to establish a key that can then be used in further communication. A key cannot be directly communicated between two parties due to the theoretical idea of data interception in every communication: if the interceptor acquires both the cipher text and the key, the message can be easily decrypted. As a result, methods need to be invented for a key to be indirectly determined by two parties, without any information that could easily cause the key to be leaked.

3.1. The protocol

The Diffie-Hellman protocol applies to two people wishing to communicate a secret key, so let one be called Alice and the other Bob.

First, Alice and Bob agree on a cyclic group G and one of its generators g. This information can be made public. Then, Alice chooses any integer a from 2 to $|G| - 1$, and Bob does the same, getting b. Alice calculates g^a and gives the result to Bob, while Bob calculates g^b and gives the result to Alice. Alice then raises Bob's result to its ath power, getting $(g^a)^b = g^{ab}$, and Bob raises Alice's result to its bth power, getting $(g^b)^a = g^{ba} = g^{ab}$. Thus, they get the same result [9].

In this process, the information that is communicated include G, g, g^a and g^b. The key of the protocol is for an interceptor not to be able to calculate x given g^x, and this is the main difficulty of breaking the Diffie-Hellman protocol, called the Discrete Logarithm Problem (DLP). If the naive approach is taken, by finding g, g^2, etc. and comparing each of them to g^x, then the time taken would be computationally infeasible ($|G|$ can get as large as, or even larger than 10^{200}, which is near impossible for normal computers to calculate within a reasonable amount of time).

However, the difficulty of the DLP varies with the group used. For instance, if the additive group of integers modulo n is used ($a * b \stackrel{\text{def}}{=} a + b \bmod n$), then the DLP in this case is essentially finding a suitable x such that $g \times x \equiv k \bmod n$, given g, k and n. This problem is identical to finding $g^{-1}k$ modulo n, which can be easily done in a feasible amount of time using methods such as the Extended Euclidian algorithm. As a result, we can see that the additive group modulo n is not a suitable group that can be used for the Diffie-Hellman problem.

As can be seen, the difficulty of the DLP depends both on the size of the group and on the nature of the group. Many groups have been suggested for which it is difficult to solve the DLP in, and a few will be mentioned in part 3.4.

3.2. Calculating a power in logarithmic time

Even though the DLP is very difficult to solve, this does not mean calculating g^x will be nearly as difficult. One method is to operate g by itself for x time, but it is possible to calculate g^x in $\log_2 x$ instead of x iterations.

One method is to use the binary representation of x. x's binary representation has $\lfloor \log_2 x \rfloor$ digits. As we can calculate $g^2 = g * g$, $g^4 = g^2 * g^2$, etc. we can calculate all g^{2^k}, for $2^k \leq x$, taking $\lfloor \log_2 x \rfloor$ operations. Then, we choose the powers of g whose digits are 1 in the binary representation of x, and operate them together. This can be alternatively written as $g^x = r_0 g^{2^0} * r_1 g^{2^1} * \dots * r_{\lfloor \log_2 x \rfloor} g^{2^{\lfloor \log_2 x \rfloor}}$ when calculating large powers of an integer, where r_n stands for the $(n+1)$th digit from the right in the binary representation of x. Calculating large powers of elements in other groups have a similar process, relies on the same principal and takes the same amount of (negligible) computation time.

DOI: 10.54254/2753-8818/13/20240844

3.3. Variants of the system: using nonabelian platform groups

The Diffie-Hellman protocol can be used on many abelian groups. For nonabelian (and thus noncyclic) groups, though, a variant of the protocol can be used. The Ko-Lee-Cheon-Han-Kang-Park Key Agreement Protocol is proposed by Ko et al. and uses the conjugates of g instead of powers of g for key exchanging.

The Ko Protocol's algorithm is described in the following way. Alice and Bob agree on a group G, one of its elements g and two of its commuting subgroups A and B, meaning that for any elements $x \in A$ and $y \in B$, $xy = yx$. Alice chooses any element $a \in G$ and Bob chooses $b \in G$. Alice calculates $a^{-1}ga$ and Bob calculates $b^{-1}gb$. (Note that in this protocol, $a^{-1}ga \neq g$, as the group is nonabelian.) Alice then gives her result to Bob and Bob gives his to Alice. Alice calculates the conjugate of $b^{-1}gb$ by a, which is $a^{-1}b^{-1}gba$, and Bob calculates the conjugate of $a^{-1}ga$ by b, getting $b^{-1}a^{-1}gab$. Because A and B are commutative, $a^{-1}b^{-1}gba = b^{-1}a^{-1}gab$ (the inverse of a group element must be in the group by definition), and as a result they have the same key.

This procedure is very similar to the original Diffie-Hellman, as nothing changes besides the power of g. It is easy to see that when the group used is nonabelian, conjugation serves as a good alternative to exponentiation. Finding an x such that $x^{-1}gx = k$ where g and k are given is called the conjugacy search problem and is the equivalent of the DLP but in a nonabelian group.

There are many other variants of the Diffie-Hellman protocol for nonabelian groups. One is the Anshel-Anshel-Goldfeld Key Agreement Protocol, which does not need two commuting subgroups, and another is the Stickel Key Agreement Protocol, which uses a two-sided exponentiation $a^x gb^y$ as its trapdoor function.

3.4. Platform groups that can be used

In the previous discussion, the group G was not specified. There are many groups that can be used as G, but to qualify as a possible group, solving the DLP on G must be computationally difficult. These functions are known as a "trapdoor function", or a "one-way function", meaning that the function is easy to compute given an input but hard to find a valid input given an output.

One candidate that can be used for G is the multiplicative group modulo n, denoted $\mathbb{Z}_n$. This is the original implementation that Diffie and Hellman used, and there is no algorithm yet which can solve the DLP on this group in a reasonable time. It has been proven that for certain primes (more specifically, when $\varphi(n)$ can be factorized into relatively small prime numbers), breaking the Diffie-Hellman problem on $\mathbb{Z}_n$ is as computationally difficult as solving the DLP on it [9].

Another group that can be used is the set of points on an elliptic curve (usually over $\mathbb{R}$), with the operation of point addition. An elliptic curve is defined as the graph of an equation of the form $y^2 = x^3 + Ax + B$, where A and B are constants. Point addition on elliptic curves is defined as follows: take two points on an elliptic curve P and Q, and connect them using a straight line. If the line intersects another point on the elliptic curve (if the line is tangent to a point, the line is defined as intersecting that point twice), then $R = P * Q$, where R is the other point of intersection. Otherwise, $R \stackrel{\text{def}}{=} (\infty, \infty)$, implying that the line PQ is vertical. Even though there is an infinity at $y = \infty$ and another at $y = -\infty$, these two cases are treated the same when using this group; in other words, the "top infinity" and the "bottom infinity" is treated as the same point, which is in fact the identity of the group [4]. (Also, the inverse of an element in the group is its reflection across the x-axis, because they form a vertical line, which means that they operate to get $e = (\infty, \infty)$.) When operating in this group, the operation $*$ is also sometimes written as $+$, hence $R = P + Q$. In some cases, attacks can be made to simplify the DLP on an elliptic curve to that of $\mathbb{Z}_n$, using pairings such as the Weil pairing and the Tate-Lichtenbaum pairing, and as a result somewhat decreasing its strength.

Lastly, a more recently proposed group which claims to be difficult on the conjugacy search problem are n-braid groups (a non-abelian group). An n-braid group consists of n strands of string, each of which connect to a point on the left-hand side and another on the right. However, the strings must go from left to right and never backwards, so knots are not allowed. Two braids with the same n are operated by

DOI: 10.54254/2753-8818/13/20240844

"sticking" the n right-hand ends of the first braid to the n left-hand ends of the second, so clearly braid groups are nonabelian. The identity braid is the one which connects all left-hand points straight to their respective right-hand points, while the inverse of a braid is the graphical reflection of it horizontally. No algorithm which solves the conjugacy search problem for braid groups quickly enough (in polynomial time complexity) has yet been found [5].

There are also quite many other viable platform groups, but the most important criteria for whether a group is useful is the difficulty of solving the DLP or the conjugacy search problem on it, and whether operating two elements and finding the inverse of any element is easy enough.

4. The RSA cryptosystem

The Rivest-Shamir-Adleman (RSA) cryptosystem is a public key cryptosystem developed by Ron Rivest, Adi Shamir, and Leonard Adleman, who publicly described the system in 1977. RSA makes the encryption key public so that anyone can send encrypted messages to a specific user but keeps the decryption key private so that only the user himself/herself can decrypt any encrypted message.

4.1. The system

Now, the RSA system's details and procedure will be described. First, Bob chooses two large primes p and q, both being around 200 digits. Then, N is calculated by $N = pq$. p and q are kept private, but N is released to the public.

Next, the numbers d and e are chosen such that $de \equiv 1 \bmod \varphi(N)$, where $\varphi(N)$ is the Euler totient function (the number of positive integers from 2 to $N-1$ that are relatively prime to N). e, the encryption key, becomes public and d, the decryption key, stays private. e is relatively prime to $\varphi(N)$ and is usually not a big number ($2^{16}+1 = 65537$ is commonly used), and it is guaranteed we can find a possible d in these conditions: $d = e^{-1} \bmod \varphi(N)$. This is a special case of cyclic groups, $\mathbb{Z}_n$, which we will talk about in the proof of the theorem.

If Alice wants to send a message to Bob, she converts the string of letters to numbers (by using base 26 or similar methods), the result being the integer M. Alice makes sure that $M < N$; otherwise, the message can be sent in parts, making M smaller for every part. Alice sends the number $X = M^e \bmod \varphi(N)$ to Bob.

To decrypt the message, Bob simply calculates $X^d \bmod \varphi(N)$, which will be equal to the M Alice sent. Note that M must be relatively prime to N for the system to work; however, this is trivial in practical usage, as N has only two prime factors, so it is extremely unlikely that N shares a factor with M [10].

As to why this system is secure: if the message is intercepted, the interceptor will know X (the intercepted message), N (the public information), and e (which is also public). If (s)he wants to find M, his/her main goal is to find a d such that $de \equiv 1 \bmod \varphi(N)$ (after that it will be easy, as only $M = X^d \bmod \varphi(N)$ needs to be calculated and every variable in the equation is known). However, this task is very difficult, as $\varphi(N)$ will be very hard to calculate for very large N (there is no quick algorithm for calculating $\varphi(N)$ if the prime factors of N are not known and factoring a number much larger than 10^{200} is also as difficult without the help of quantum computing). As a result, there have not yet been methods to use intercepted data to immediately acquire M [11].

4.2. Proving Lagrange's theorem

Lagrange's Theorem states that if $H \leq G$, then $|H|$ divides $|G|$. To prove this, we need to prove two lemmas first. We prove that if $H \leq G$, then for all $a \in H$, we have $aH = H$; we also prove that if $H \leq G$ and $a, b \in G$, then either $aH = bH$ or $aH \cap bH = \emptyset$.

To prove the first lemma, we prove that $aH \subseteq H$ and $H \subseteq aH$. To show that $aH \subseteq H$, note that H is a group which must be closed by definition, so for every $h \in H$, $ah \in H$. Thus $aH \subseteq H$.To show that $H \subseteq aH$, we let h be any element in H. $a^{-1}h \in H$ due to the properties of group H, so $a(a^{-1}h) = h \in aH$. Since $aH \subseteq H$ and $H \subseteq aH$, $aH = H$.

DOI: 10.54254/2753-8818/13/20240844

For the second lemma, we assume that $aH \cap bH \neq \emptyset$, and show that $aH = bH$. Let an element $x \in aH \cap bH$, so there must exist h_1 and h_2 such that $ah_1 = bh_2 = x$. H is a group, so ${h_1}^{-1} \in H$. We operate the two sides by ${h_1}^{-1}$ on the right: $ah_1{h_1}^{-1} = a = bh_2{h_1}^{-1}$, so $aH = bh_2{h_1}^{-1}H$. Since $h_2{h_1}^{-1}H = H$ (H is closed), $aH = bH$.

Having proven those two lemmas, we can now prove Lagrange's Theorem. We realize that any element in a group G, a, must be inside at least one of the cosets of G, because $a \in aH$ (H contains e and $ae = a$). Next, as two cosets are either equal or contain no common elements, we can choose one coset from each set of identical cosets, totaling n cosets. Since $|aH| = |H|$, $|G| = |a_1H| + |a_2H| + \cdots + |a_nH| = n|H|$, where a_k is the kth coset that is chosen to be included, so $|H|$ divides $|G|$. Thus, Lagrange's theorem has been proven.

4.3. Proving the validity of RSA using Lagrange's theorem

Proving the validity of RSA means showing that $X^d \bmod \varphi(N) = m$; that is, the decrypted message is the same as the original. Before proving this, we must first prove Fermat's Little Theorem using Lagrange's Theorem.

Fermat's Little Theorem states that $a^{n-1} \equiv 1 \bmod n$, where $a \in \mathbb{Z}$ and n is prime. We prove this by using the properties of the group $\mathbb{Z}_n$. Let a be any element in $\mathbb{Z}_n$, and let $k = \text{ord}(a)$. The elements generated by a, $K = \{1, a, a^2, \ldots, a^{k-1}\}$ (note that $1 = a^0$), form a subgroup of G, because the set of K is a subset of the integers from 1 to $n-1$ (subset), $a^i \times a^j = a^{ij \bmod k}$ (closure), G is associative so the group of its subset must also be associative (associativity), the element $a^0 = 1$ is in the group (identity), and $a^i \times a^{k-i} \equiv 1 \bmod n$ (existence of inverses). $|K| = k$, because it contains all elements from a^0 to a^{k-1}, and according to Lagrange's Theorem, $|K|$ divides $|G|$, which means that k divides $n-1$. We can write $n - 1 = rk$, where $r \in \mathbb{Z}^+$. Then, $a^{n-1} \equiv a^{rk} \equiv (a^k)^r \equiv 1^r \equiv 1 \bmod n$. Thus, Fermat's Little Theorem is true.

When proving the validity of RSA, we assume that M and N are relatively prime to each other. We want to show that $(M^e)^d \equiv M \bmod N$ for all M between 1 and $N-1$. Since $ed \equiv 1 \bmod \varphi(N)$, $ed = s\varphi(N) + 1 = s(p-1)(q-1)$ for some $s \in \mathbb{Z}^+$ ($\varphi(N) = (p-1)(q-1)$ because p and q are different primes). Because $p - 1 | \varphi(N)$, $p - 1 | k\varphi(N)$. $M^{ed} \equiv M^{s\varphi(N)+1} \equiv M \times M^{s(p-1)(q-1)} \equiv M \times (M^{p-1})^{s(q-1)} \equiv M \times 1^{s(q-1)} \equiv M \bmod p$ ($M^{p-1} = 1$ due to Fermat's little theorem), so $p | M^{ed} - M$. Similarly, $q | M^{ed} - M$, thus $pq | M^{ed} - M$ (because p and q are both prime). Rearranging, we get $M^{ed} \equiv M \bmod pq$, leading to $M^{ed} \equiv M \bmod N$, validating the RSA system.

5. Conclusion

This paper mainly covers two commonly used cryptographic protocols, the Diffie-Hellman protocol and the RSA protocol. First, the important cryptographic terms and notions were introduced, such as the definition of a cryptographic protocol and its security. Next, the underlying theory of the Diffie-Hellman protocol was explained using group theory, and relevant concepts were explained earlier in the paper. The security of certain platform groups of the Diffie-Hellman protocol was also mentioned, and for now these algorithms stay quite safe, if the correct techniques and choice of platform groups are used. For the RSA protocol, the algorithm and the proof of validity is mentioned, by using group theory (Lagrange's theorem) to prove the result. In the future, though, the emergence of quantum computers may make some protocols prone to attack by using completely new computing methods. There has been hints that even the DLP can be broken soon by quantum computers.

This paper provides an overview of the main algorithms proposed before for the purposes of cryptography and some basic ideas about group theory for readers new to cryptography and to group theory.

However, there are many limitations on this paper. Many other useful and common protocols have not been covered in this paper, and the attacks have also not been well described. Also, some of the protocols and methods (such as the Ko protocol) have not been fully described due to time reasons.

DOI: 10.54254/2753-8818/13/20240844

Authors Contribution

All the authors contributed equally and their names were listed in alphabetical order.

References

[1] Anshel I, Anshel M and Goldfeld D 1999 An algebraic method for public-key cryptography. *Mathematical Research Letters* 6, 287–91.
[2] Fujisaki E and Okamoto T 1999 How to enhance the security of public-key encryption at minimum cost. *Lecture Notes in Computer Science* 1560, 53-68.
[3] Dehornoy P 2004 Braid-based cryptography. *Contemporary Mathematics*, 360.
[4] Washington L 2008 Elliptic curves: number theory and cryptography. *New York: Chapman & Hall*, 1-18.
[5] Ko K, Lee S, Cheon J, Han J, Kang J and Park C 2000 New public-key cryptosystem using braid groups. *Lecture Notes in Computer Science*, 1880, 166-183.
[6] Seetha R and Mythili N 2020 Modern cryptography – a review. *International Journal of Scientific & Technology Research*, 9, 1679-1685.
[7] Kahrobaei D, Flores R and Noce M 2023 Group-based cryptography in the quantum era. *Notices of the American Mathematical Society*, 70, 752-763.
[8] Anshel I, Atkins D, Goldfeld D and Gunnells P 2017 WalnutDSATM: a quantum-resistant digital signature algorithm. International *Journal of Computer Mathematics: Computer Systems Theory*, 6, 260-284.
[9] Boer B 2000 Diffie-Hellman is as strong as discrete log for certain primes. *Theory and Application of Cryptography*, 403, 530-539.
[10] Lee G T 2018 Abstract algebra. *New York: Springer International Publishing AG*, 233-239.
[11] Miller S and Takloo-Bighash R 2006 An invitation to modern number theory. *Princeton: Princeton University Press*, 3-28.