International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 7 Issue 5 ǁ 2019 ǁ PP. 37-41

Role of Cryptanalysis

Prof. (Dr.) Deepak Raj
P.I.E.T. Samalkha (Panipat)

ABSTRACT
In Mathematical Science Group Based Cryptography and other forms of cryptography are present. In this paper, we discuss the insecurity factors of various schemes on which cryptographic systems are built. Various attacks are discussed which can be made on these systems.

I. Introduction

Cryptanalysis is a broad concept that refers to the study of ciphers, ciphertext or cryptosystems i.e to secret code systems with a view to tracing weakness in them that will permit getting of the plaintext or original message from the ciphertext without having the knowledge of the key or algorithm. This concept is known as breaking the cipher, ciphertext or cryptosystem. An approach related to cryptography is cryptanalysis. The cryptographer's goal is to provide security for information by developing strong cryptosystems, while the cryptanalyst's goal is to discover weakness or flaws in cryptosystems and the break the security provided by those systems. In fact, a good cryptanalyst can even determine plaintext from samples of ciphertext without even knowing the cipher that was used to produce it. When properly implemented , standard cryptography based security technologies can provide lot of protection against a wide range of attacks, including common cryptanalyst attacks. Cryptosystems come in 3 kinds:

1. Those that have been broken (most).
2. Those that have not yet been analyzed (because they are new and not yet widely used.
3. Those that have been analyzed but not broken. (RSA, Discrete log cryptosystems.

There are three most common ways to turn cipher text into plaintext:

1. Steal/purchase/bribe to get key
2. Exploit sloppy implementation/protocol problems (hacking/cracking). Examples are some- one used spouse's name as key, someone sent key along with message
3. Cryptanalysis

1. Types of Cryptanalysis

To study the cryptanalysis it can be broadly classified into the following three categories

1.1 Cipher text only attack

The enemy has intercepted cipher text but has no matching plain-text. You typically assume that the enemy has access to the cipher text. Two situations:

a) The enemy is aware of the nature of the cryptosystem, but does not have the key. True with most cryptosystems used in U.S. businesses.

b) The enemy is not aware of the nature of the cryptosystem. The proper users should neverassume that this situation will last very long. The Skipjack algorithm on the Clipper Chipis classified, for example. Often the nature of a military cryptosystem is kept secret as longas possible. RSA has tried to keep the nature of a few of its cryptosystems secret, but they were published on Cipher punks.

1.2 Known plaintext attack (KPA)

In the KPA System it has been assumed that the enemy has some matched cipher text/plaintext pairs. The enemy may well have more cipher text also. The **known-plaintext attack (KPA)** is an attack model for cryptanalysis where the attacker has access to both the plaintext(called a **crib**), and its encrypted version (ciphertext). These can be used to reveal further secret information such as secret keys and code books. Here the basic concept behind a crib is that cryptologists were looking at incomprehensible ciphertext, but if they had a clue about some word or phrase that might be expected to be in the ciphertext, they would have a "wedge," a test to break into it. If their otherwise random attacks on the cipher managed to sometimes produce those words or phrases, they would know they might be on the right track. Under such circumstances when those words or phrases appeared, they would feed the settings they had used to reveal them back into the whole encrypted message to good effect. Modern ciphers such as Advanced Encryption Standard are not currently known to be susceptible to known-plaintext attacks.

The older versions of the zip format specification have chances of this attack by using PKZIP stream cipher. Consider an example , an attacker with an encrypted ZIP file needs only one unencrypted file from the archive

which forms the "known-plaintext". After that there are some publicly available software by using them they can quickly calculate the key required to decrypt the entire archive. To obtain this unencrypted file the attacker could search the website for a suitable file, find it from another archive they can open, or manually try to reconstruct a plaintext file armed with the knowledge of the filename from the encrypted archive. However, the attack does not work on AES-encrypted zip files.

1.3 Chosen plaintext attack(CPA)

The CPA model, appears to be an unrealistic model at first instance because it is unlikely that an attacker could persuade a human cryptographer to encrypt large amounts of plaintexts of the attacker's choosing. However, modern cryptography is implemented in software or hardware and is used for a diverse range of applications; for many cases, a chosen-plaintext attack is often very feasible. Chosen-plaintext attacks become extremely important in the context of public key cryptography, where the encryption key is public and so attackers can encrypt any plaintext they choose. Here we assume that the enemy can choose the plaintext that he wants put through the cryptosystem. Though this is, in general, unrealistic, such attacks are of theoretic interest because if enough plaintext is known, then chosen plaintext attack techniques may be useable. However this is an issue with with smart cards.

The various forms of CPA are as follows:-

- **Batch chosen-plaintext attack**, where the cryptanalyst chooses all of the plaintexts before seeing any of the corresponding ciphertexts. This is often the meaning of an unqualified use of "chosen-plaintext attack".
- **Adaptive chosen-plaintext attack** (CPA2), where the cryptanalyst can request the ciphertexts of additional plaintexts after seeing the ciphertexts for some plaintexts.

CPA Relation to other attacks

A CPA is more powerful than known-plaintext attack(KPA), because the attacker can obtain many pairs of plaintexts and ciphertexts, instead of only one pair, and therefore has more data for cryptanalysis. Therefore, any cipher that prevents chosen-plaintext attacks is also secure against known-plaintext and ciphertext-only attacks. However, a chosen-plaintext attack is less powerful than a chosen-ciphertext attack, where the attacker can obtain the plaintexts of arbitrary ciphertexts. A CCA-attacker can sometimes break a CPA-secure system.

3.0 Insecurity of Group based schemes

Now, we briefly outline some techniques that have been developed to demonstrate the insecurity of group-based schemes.

3.1 Analysis of braid based schemes

To perform the analysis of braid based schemes consider the conjugacy problem in which we have to find out that whether two braids, in other words two elements of the braid group are conjugate or not. To solve the conjugacy problem Garside in 1969 given an algorithm in the braid group Bn. Algorithm given by Garside was successful at that time but in 1980, a question arises regarding the effecieny of the algorithm developed by Garside. There has been a great deal of research, motivated by cryptographic applications, into finding a polynomial time solution to the conjugacy problem. Given two braids x, y ∈ Bn, Garside worked on idea of constructing finite subsets called summit sets Ix, Iy of Bn such that x is conjugate to y if and only if Ix = Iy. The solution of conjugacy problem using this method would give an efficient solution to the conjugacy search problem and hence render the braid based protocol of Ko et al. theoretically insecure. However, given braid x, Garside's summit set Ix may be exponentially large. Thus there was a challenge to prove a polynomial bound on the size of a suitable invariant set associated with any given conjugacy class. Many refinements such as the super summit set, ultra summit set, and reduced super summit set methods have been made over the years to the summit set but a polynomial bound remains elusive. In Recent times the main focus has been on an efficient solution to each of the three types of braids: periodic, reducible or pseudo-Anasov . If someone wants to break a braid-based cryptosystem for the purposes of cryptography , then he need not require to efficiently solve the conjugacy problem. He is free to use the specifics of the protocol being employed; any algorithm only requires to work for a significant proportion of case even heuristic algorithms are quite acceptable. Indeed, Hofheinz and Steinwandt used a heuristic algorithm to solve the conjugacy search problem with very high success rates: their attack is based on the fact that representatives of conjugate braids in the super summit set are likely to be conjugate by a permutation braid (a particularly simple braid). Their attack shows an inherent weaknesses of both the Ko et al. protocol and the Anshel et al. protocol for random instances, under suggested parameters. (This has led researchers to study ways of generating keys more carefully, to try to avoid easy instances.) Around the same time, many other powerful attacks were discovered, and we now discuss some of the work that has been done.

3.2 Length-based attacks

It is an heuristic procedure for finding the Alice's private key A (B). Tannenbaum and Hughes introduced length based attacks. In certain cases, these attacks provide a neat and good probabilistic way of solving the conjugacy search problem. Suppose we are given an Hughes instance of the conjugacy search problem in Bn so we are given braids x and , y^{-1}xy and we want to find y. Suppose l : Bn → Z be a suitable length function on Bn (for example, the length of the normal form of an element). If we can write y = y′σ_i for some i, where y′ has a shorter length than y, then l(σ_i y^{-1} xyσ_i^{-1}) should be strictly smaller than l(yσ_j y^{-1} xyσ_j^{-1}) for j ≠ i. So from this i can be known by repeating the attack for a smaller instance y′ of y. This attacks will be successful or not , it depends on the specific length function used . For braid groups, there are many such suitable length functions by which this attack is possible. Before making in practice these length-based attacks ,we need to modify them to ensure that we do not get stuck in short loops; see Garber et al. [29] and Ruinskiy et al. [77]. Garber et al. [29] and Myasnikov and Ushakov [67] contain convincing attacks on both the Ko et al. and Anshel et al. protocols using a length-based approach.

3.3 Linear algebra attacks

A linear representation of the braid group is taken and then we solve the conjugacy search problem using linear algebra in a matrix group. There are two well-known representations of the braid group: the Burau representation (unfaithful for n ≥ 5) and the faithful Lawrence-Krammer representation. [42] Attacks can be made on the Anshel et al. protocol us ing the Burau representation given by Hughes and Lee . Cheon and Jun [23] provide a polynomial time algorithm to break the Ko et al. protocol using the Lawrence–Krammer representation. Under the Lawrence–Krammer representation, Budney [15] studies the relationship between conjugacy of elements in the braid group and conjugacy of their images in the unitary group.

3.4 Other directions

To improve the security of schemes based on the above protocols many protocols have been developed. Themes range from changing the underlying problem (and instead investigating problems such as the decomposition problem, the braid root problem, the shifted conjugacy problem and more) to changing the platform group (Thompson's group, polycyclic groups and others have been suggested). Moreover cryptographers have created authentication schemes and signature based on the conjugacy search problem to ensure the security of the system. But it is observed that random or generic instances of either protocol lead to particularly simplified attacks.

4.0 Stickel's scheme

Stickel's give an idea to discuss cryptanalysis of a key exchange scheme. Diff-Hellman Protocol was also discussed but this protocol is significant than that of the well-known Diffie-Hellman protocol , although formally it is not a generalization of the latter. The choice of platform adopted by Stickel makes the protocol helpless to linear algebra attacks. It appears that even such a apparently minor improvement as using non-invertible matrices instead of invertible ones would already make Stickel's protocol significantly less vulnerable, at least to linear algebra attacks. Perhaps more importantly, that to obtain the shared secret key in Stickel's scheme, the adversary does not have to solve any discrete logarithm-type problem; instead, he/she can solve the apparently easier decomposition search problem in the platform (semi)group G which is: Given a recursively presented (semi)group G, two recursively generated sub(semi)groups A, B ≤ G, and two elements u, w ∈ G, find two elements x ∈ A and y ∈ B that would satisfy x·w·y = u, provided at least one such pair of elements exists. Stickel's scheme was successfully cryptanalysed by Shpilraine include a brief description of this attack as it is particularly simple, and illustrates what can go wrong if care is not taken in protocol design. The attack works as follows. First note that an adversary need not recover any of the private exponents l, m, r, s in order to derive the key k. Instead, it suffices upon intercepting the transmitted messages u and v, to find n × n matrices x, y ∈ G such that

xa = ax, yb = by, u = xgy.

One can then compute

xvy = x a^r g b^s y = a^r xgy b^s = a^r u b^s = k.

It remains to solve these equations for x and y. The equations xa = ax and yb = by are linear, since a and b are known. The equation u = xgy is not linear, but since x is invertible we can rearrange: x^{-1} u = gy, with g and u known. Since xa = ax if and only if x^{-1} a = a x^{-1}, we write x_1 = x^{-1} and instead solve the following matrix equations involving x_1 and y:

x_1 a = a x_1 , yb = by, x_1 u = gy.

Setting x_1 = gyu^{-1} we can eliminate x1 to solve

gyu^{-1} a = agyu^{-1} , yb = by.

Here now only y is unknown and we have $2n^2$ linear equations in n^2 variables: a heavily overdetermined system of linear equations, and an invertible matrix y will be easily found. Shpilrain's attack is specific to the platform group GL(n, Fq). In particular, it uses the fact that x and u are invertible. Thus to thwart this attack, it makes sense to restrict the protocol to non-invertible matrices. However, it is unclear whether or not this actually enhances the security of the protocol.

5.0 Analysis of schemes based on logarithmic signatures

Majority of schemes based on logarithmic signatures main problem is to specify how this should be done. How can secure logarithmic signatures be generated? Magliveras et al. had the idea of restricting the logarithmic signature used in MST 1 to be totally non-transversal, that is a logarithmic signature α for a group G in which no block Ai of α is a coset of a non-trivial subgroup of G. However, this condition was shown to be insufficient by Bohli et al. who constructed instances of totally non- transversal logarithmic signatures that were insecure when used in MST 1. Key generation is also a problem for MST 2; see for a critique of this. As for MST 3, this was recently cryptanalysed by the authors. Thus it seems that a significant new idea in this area is needed to construct a secure public key cryptosystem from logarithmic signatures.

5.1 Short logarithmic signatures Let G be a finite group of order $\prod_{j=1}^{t} p_j^{a_j}$

with p_j distinct primes. Let α = [$A_1, \ldots A_S$] be a logarithmic signature for

G, with | A_i | = r_i for 1 ≤ i ≤ s. Define the length of α to be $l(\alpha) = \sum_{i=1}^{s} r_i$

The length of α is an efficiency measure: it is the number of elements that

Must be stored in order to specify a typical logarithmic signature of this kind. Since |G| = $\prod_{i=1}^{s} r_i$ we must have that l(α) ≥ $\sum_{j=1}^{t} a_j p_j$. A logarithmic signature achieving this bound is called a minimal logarithmic signature for G. An attractive open problem is: does every finite group have a minimal logarithmic signature? Now, if G has a normal subgroup N with G/N ≅ H and H and N both have minimal logarithmic signatures then G has a minimal logarithmic signature. In particular, it is clear that any soluble group has a minimal logarithmic signature. Moreover, to answer the question in the affirmative it suffices to consider simple groups only. Minimal logarithmic signatures have been found for An, PSLn(q), some sporadic groups and most simple groups of order up to 10^{10} .

Why do we attempt to propose new cryptosystems, when elliptic curve DLP systems work well? A major motivation is the worry that a good algorithm could be found for the elliptic curve DLP. This worry has increased, and the search for alternative cryptosystems has become more urgent, with the realisation that quantum computers can efficiently solve both the in- teger factorisation problem and the standard variants of the DLP. If quantum computers of a practical size can be constructed, classical public key cryptography is in trouble. Cryptosystems, including group-based ex- amples, that are not necessarily vulnerable to the rise of quantum computers have become known as post-quantum cryptosystems. A well known example, invented well before quantum computers were considered, is the McEliece cryptosystem based on the difficulty of decoding error correcting codes. Other examples include lattice-based cryptosystems (such as the GGH cryptosystem and cryptosystems based on large systems of multivariate polynomial equations (such as the HFE family of cryptosystems. Though many of these cryptosystems suffer from having large public keys, they are often computationally efficient and so we feel that these schemes are more likely than group-based cryptosystems to produce protocols that will be used in practice.

II. Conclusion

Cryptanalysts can use powerful computing equipment and a variety of procedures, processes, and techniques to launch attacks against cryptosystems. In fact, a good cryptanalyst can even determine plaintext from samples of ciphertext without even knowing the cipher that was used to produce it. Knowledgeable intruders can use cryptanalysis techniques as part of their attacks against your cryptography-based security systems. When properly implemented, standard cryptography-based security technologies can provide ample protection against a wide range of attacks, including common cryptanalysis techniques. However, to obtain highly valuable information, skilled intruders or trained espionage agents with access to powerful computing

resources might have the incentive to launch expensive and highly sophisticated cryptanalyst attacks. Stopping sophisticated cryptanalyst attacks requires highly secure systems that use strong cryptography-based security technologies. From the above discussion we are in a position to point out that group-based cryptography motivates some beautiful and natural questions for the pure group theorist. Most obviously, the cryptosystems above motivate problems in computational group theory, especially combinatorial group theory. But we would like to highlight two more problems as examples of the kind of questions that can arise. Despite ten years of strong interest in group-based cryptography, a well studied candidate for a secure, well specified and efficient cryptosystem is yet to emerge.

References

[1]. David Garber, Shmuel Kaplan, Mina Teicher, Boaz Tsaban and Uzi Vishne, Probabilistic solutions of equations in the braid group, Adv. Appl. Math. 35 (2005), 323–334.

[2]. James Hughes, A linear algebraic attack on the AAFG1 braid group cryptosystem, in Information Security and Privacy (G. Goos, J. Hart-manis and J. van Leeuwen, eds), Lecture Notes in Computer Science 2384 (Springer–Verlag, Berlin, 2002), 176–189

[3]. David Garber, Shmuel Kaplan, Mina Teicher, Boaz Tsaban and Uzi Vishne, Probabilistic solutions of equations in the braid group, Adv. Appl. Math. 35 (2005), 323–334.

[4]. Ryan D. Budney, On the image of the Lawrence–Krammer representa-tion, J. Knot Theory and its Ramifications 14 (2005), 1–17.

[5]. Mar´ıa Isabel Gonzalez Vasco, Martin Rotteler and Rainer Steinwandt, On minimal length factorizations of finite groups, J. Exp. Math. 12 (2003), 1–12.

[6]. Mar´ıa Isabel Gonz´alez Vasco and Rainer Steinwandt, Obstacles in two public-key cryptosystems based on group factorizations, Tatra Mt. Math. Pub. 25 (2002) 23–37.

[7]. P. E. Holmes, On minimal factorisations of sporadic groups, J. Exp. Math. 13 (2004) 435–440.

[8]. Aviad Kipnis and Adi Shamir, Cryptanalysis of the HFE public key cryptosystem, in Advances in Cryptology – CRYPTO ’99 (M. Weiner, ed.), Lecture Notes in Computer Science 1666 (Springer, Berlin, 1999) 19–30.

[9]. Wolfgang Lempken and Tran van Trung, On minimal logarithmic sig-natures of finite groups, J. Exp. Math. 14 (2005) 257–269.

[10]. S. S. Magliveras, Secret and public-key cryptosystems from group fac-torizations, Tatra Mt. Math. Publ. 25 (2002). 1–12.

[11]. R.J. McEliece, A public key cryptosystem based on algebraic coding theory, DSN Progress Report 42 - 44 (Jet Propulsion Lab, Pasadena, 1978) 114–116.

[12]. Sean Murphy, Kenneth Paterson, and Peter Wild, A weak cipher that generates the symmetric group, J. Cryptology 7 (1994) 61–65.