Computability in Cryptography

A.K.A. Iroshan, K.H.H. Indrajee, P. Mahawithana, K.U.G.S. Dharshana, P.H.S.R. Premarathna
Department of Computer Science & Engineering, University of Moratuwa

Abstract — *Cryptography is a concept and a methodology for secret communication, which is used and improved over thousands of years. Cryptography includes a vast variety of implementation schemes, starting from simple shifting ciphers to DNA algorithms. In this paper we discuss extensively the algorithms behind those cryptographic schemes and the computability aspects of such algorithms. We outline the advantageous factors in computational complexity in such schemes. Also some interesting research being done on this area and problem areas where further research is needed, are outlined.*

Index Terms — Computability, Cryptography, Encryption

I. Introduction

Cryptography is a technique that is used to communicate securely in presence of unintended third parties, without allowing unintended parties to intercept the messages. It has a long history which runs back to 1600 BC. In the present world, where the communication plays a major role in many activities, the use of cryptography has been very important. The major applications of the cryptography in present include military communication, secure internet communication and e-commerce.

Computability is the ability to solve a computational problem. In cryptography, the various algorithms that are used to secure (encrypt) the data should be computable to the two parties who are communicating and for others the data should not be computable. The computability in cryptographic algorithms determines their strength, and the level of security. Breaking the encryption method to derive the original message is known as "Cryptanalysis". For years, several different methods have been used for the purpose of encryption and the cryptanalysis has also been developed in parallel to break these encryption methods.

II. Document Overview

The objective of this literature review is to discuss aspects of computability in cryptography and provide the reader knowledge on past present and future trends in crypto-systems. The review starts with a brief introduction to the history of cryptography and continues providing explanations on classical cryptosystems. Next the reader is introduced to modern cryptosystems. Two major modern cryptosystems are taken into consideration namely Symmetric Key Cryptography and Public Key Cryptography. Some chosen implementation schemes of these cryptosystems are extensively described. The paper next moves to describe few problems in modern cryptosystems and implementations. in certain scenarios or information systems and suggested solutions. Finally a discussion on advancements of modern cryptosystems and future research trends is included.

III. The History of Cryptography

Cryptography was invented thousands of years ago. The oldest use of the cryptography discovered is, the monuments of the old Egypt in 1900 BC [1].

The use of the ciphers has been invented about thousand years later by the Hebrew and Romans. The Caesar Cipher is one of the famous ciphers used by the Roman emperor Caesar. In early years the ciphers were mono-alphabetic and had used both substitution and transposition techniques. Later around in 15th century, the poly-alphabetic ciphers were invented and used more often because of the vulnerabilities of the mono-alphabetic ciphers [1] [2].

In the era of World War I and II, the cryptography played a major role in the military communication. Many different ways and machines were developed not only to encrypt and decrypt the messages but also to break the messages of the enemies. PURPLE machine by Japanese, ENGIMA machine by German, TypeX by British, SIGABA by Americans. were some of such machines developed in World War era [1] [2].

With the rapid development of computers and communication in the recent past, the modern cryptography was developed. Public Key or the Asymmetric key encryption, and Symmetric key encryption are the major methods of the modern cryptography. Those methods are much powerful compared to any early methods of encryption and it is proved that these are very hard to break [1].

A. Some Classical Encryption Algorithms and Their Breakability

The different encryption algorithms have different pros and cons associated with them. The computability or the ability to compute the original message from a cipher text is the most important fact about an encryption algorithm. This is also known as cryptanalysis in cryptography. The following paragraphs include a brief introduction to some commonly used ciphers and their computability.

(a) Caesar Cipher

This was used by the ancient roman emperors to send military messages securely to their fleets of armies in faraway cities. To encrypt a message with the Caesar cipher, each letter of the message was replaced with the letter that is a constant number of positions later in the alphabet. For example the letter 'A' in the original message would be replaced by 'D' when the shift is three positions later in the alphabet [3].

However, the Caesar cipher is very easy to break even by a brute-force method. The letters of the message can be shifted by the letters x positions later in the alphabet, where x is incremented while the decrypted message gives a meaning. So in the worst case by 25 tries it is breakable.

(b) Key based Mono-alphabetic Substitution Ciphers

This is similar to the Caesar cipher. But the substitutions of the letters are done in a different way.

- First a key is chosen to build the cipher alphabet.
- The cipher alphabet is built such that the letters in the key at the start and the rest of the letters are afterward avoiding repeating of the same letter.
- Then the letter of the original message is substituted by the letter at the same index of the cipher alphabet [3].

This is somewhat powerful than the Caesar cipher. But this cipher is also breakable by doing a frequency analysis of the letters in the alphabet. For example letters 'E' and. 'A' are most frequently occurred in the English alphabet. Therefore by doing a statistical analysis on the frequency of the letters of encrypted text, it is easy to find the substituted letters for 'E' and 'A'. Therefore by using statistical analysis Caesar cipher can be broken.

(c) Poly-alphabetic Substitution Ciphers

In this method, multiple cipher alphabets are used instead of one cipher alphabet. The cipher alphabet is alternated in each letter so that the cipher is unbreakable just using a frequency analysis. Vigenère square is an example for this cipher [3].

This is much powerful than the mono-alphabetic ciphers and this had been very hard to break for some time in the history. But later, the methods such as "Kasiski examination" were invented to break this cipher [3].

(d) Transposition Ciphers

The transposition cipher doesn't substitute the letters of the original message with some different letters. Instead it changes the order of the letters they appear. For example, a very simple case is writing the original message backwards from the last letter to the first letter.

This cannot be broken by a frequency analysis, because the no of the occurrences of a letter in the encrypted text is always the same as that in the original text. However according to [4] this cipher is also breakable by Genetic algorithms and Suitability Assessment.

IV. Modern Cryptography

Use Public Key or the Asymmetric key encryption, and Symmetric key encryption are the major methods of the modern cryptography. There are many sound and complete implementations of these two systems.

A. Symmetric Key Cryptography

The symmetric key cryptography was developed as a solution to the breakable classical cryptography.i.e. Caesar cipher. In a symmetric key cryptosystem, a single key is used to encrypt and decrypt data between two communicating hosts. Thus the two parties must first agree on the shared secret key prior to secure communication. The secret key must first be exchanged using some other secure manner prior to starting secure communication using the symmetric key cryptosystem. The sender of the message first encrypts the message and obtains cipher text using the symmetric key and the receiver decodes it using the same symmetric key to extract the original message. The breakability of the system depends directly on the computability of reversing the encryption without knowing the secret key.

B. Public Key Cryptography

The public key cryptography was developed to overcome issues in one-to-many communication in symmetric cryptography. This uses a one way function to encrypt the data.

A One Way Function (OWF) is a function 'f' which is feasible to compute but its inverse. 'f–1', is infeasible to compute, where the feasibility of the computation is measured against the Turing Machine. It simply means that the function 'f' can be computed in polynomial time by a Turing machine, but the inverse function cannot be measured in polynomial time [5].

In public key cryptography, the receiver first announces a function 'f' that should be used to encrypt the data, which is sent by the sender. The sender encrypts the data using that function to send it to the receiver. The receiver uses the inverse 'f-1' function of his/her encryption function to decrypt the message. However, this 'f' function may be accessible by any other unintended party also. As a result, those unintended parties can come up with the inverse function f-1 of the encryption function and they can decrypt the messages with that f-1 and see what their content is. To prevent f-1 being derived easily from the f, the f is selected as computationally intractable. That means the deriving 'f-1'from f will take extraordinary large time period or as mentioned in a previous paragraph f is an one-way function. RSA algorithm is an implementation of the public key cryptography.

The researchers have proven that, in certain public key cryptography implementations the calculation of the private key from the public key is a NP complete problem and very hard to find by brute force [6].

V. Implementations of Modern Cryptosystems

A. The RSA Algorithm: An Implementation of PKC

RSA is a widely used algorithm in ecommerce applications to encrypt and secure confidential information. RSA is a public key encryption algorithm named after Ron Rivest, Adi Shamir and Len Adleman, who invented it in 1977. RSA is used both in public key encryption as well as digital signatures. It has many benefits than symmetric key encryption. Over the years RSA system has been analyzed for vulnerability by many researchers and it is proven to be secure if properly implemented.

In their paper "Cryptography Methods Using the RSA Algorithm" [7] Cristian Lupu, Bogdan Firtat, and Claudia Enoiu gives a comprehensive explanation on the RSA algorithm.

(a) Advantages of public key cryptosystems over symmetric key cryptosystems

In symmetric key encryption the two parties who want to securely communicate over a public channel must first have the same key to decrypt the messages prior to secure communication. The key exchange should therefore happen through a secure channel before a secure communication can happen through the public channel. This is a major disadvantage of symmetric key encryption. Public key cryptosystems however overcomes this difficulty by having two related complementary keys i.e. a publicly revealed key and a secret key. A message sender would use the recipient's public key to encrypt a message but to decrypt it the secret key which is only known to the recipient is needed, thus enabling secure communication [7].

Message authentication can also be provided through public key encryption. The sender's secret key can be used to encrypt an identification message therefore signing it. The encrypted authentication message can be decrypted by the public key but to encrypt a message that can be decrypted by the public key one must have the secret key. This means only the sender who has the secret key can encrypt the message as it is proving the sender was the true originator of the message [7].

When only the integrity of a message is required and confidentiality is not a requirement encrypting the whole message is a waste of computational power and time. Cryptographic hash functions are used in such situations. A cryptographic hash function computes a message digest, and instead of encrypting the whole message now only the digest is encrypted. This is much faster than encrypting the whole message. CristianLupuet al highlights the fact that "Cryptographic hash functions typically produce hash values of 128 or more bits. This number is vastly larger than the number of different messages likely to ever be exchanged in the world" [7].

It is stated that the key length for a secure RSA transmission is typically 1024 bits. The longer the key more secure the data will be. Factoring is the act of splitting an integer in to a set of smaller integers which when multiplied together forms the original number. Prime factorization is to factorize the integer into factors which are prime numbers. The security of the RSA algorithm depends on the fact that computability of the factoring problem being extensively difficult so far.

(b) Principles of operation: RSA cryptosystem.

- RSA cryptosystem uses a public exponent e for encryption and a private exponent d for decryption.
- It uses a modulus N which is a product of two large prime numbers p and q, i.e., N = p*q.
- The exponents e and dare chosen to satisfy the condition e*d = 1 mod (p - 1) (q - 1).
- Now the public key is (N, e) and the private key is d

Example:

Select two prime numbers p = 11 and q = 3, then N = 11 * 3 = 33. Now compute,

(p - 1) (q - 1) = 10 * 2 = 20

And choose a value e relatively prime to 20, say 3. Then d has to be chosen such that,

e*d = 1 mod 20.

One possible value for d is 7 since 3 * 7 = 21 = 1 mod 20. So the public key is (N= 33, e= 3) and the corresponding private key d = 7. Now original factors p and q are discarded. Factoring breaks RSA. If an attacker can factor N into p and q, he can use the public value e to easily find the private value d. Thus security of RSA strongly depends on the computability of factorization problem.

To encrypt a plaintext message M, compute C = Me mod N, where C is the encoded message, or cipher text. To decrypt the cipher text C, compute M = Cd mod N, which yields the original message M. Continuing the previous example where the public key (N= 33, e= 3) and the private key d = 7. Let message M = 19. Encryption generates an encoded message C =Me mod N = 193 mod 33 = 28. The sender sends the encrypted message C = 28. Receiver uses the private key d and computes Cd mod N = 287 mod 33 = 19, which is the original message M. The decryption works thanks to a result from number theory known as Euler's Theorem [8].

Another use of RSA is to generate digital signatures which serve to verify the source of the message. Signing uses the private key d and is the same mathematical operation as decrypting. The receiver uses the public key e and performs an encryption operation to verify the signature.

The operation in RSA that involves the private key is thus the modular exponentiation M = Cd mod N, where N is the RSA modulus, C is the text to decrypt or sign, and d is the private key.

B. A Stream-based Implementation of XML Encryption

The paper "A stream-based implementation of XML encryption" [9] by Takeshi Imamura, Andy Clark, and Hiroshi Maruyama discusses about an efficient way of encrypting and decrypting the XML files than the presently popular methods.

(a) XML

Takeshi Imamura et al first provide an introductory explanation to what is XML. XML is a language which is used for representing tree structured data. It has been standardized by W3C in 1998. As the data represented in the XML format are in plain text, the data which needs confidentiality cannot be represented using this method. Therefore the encryption methods were introduced for the XML files. But as the current implementations are dependent on DOM (Document Object Model) API for encrypting the XML and all of them are not performing well in performance wise. we need to find a new way of encrypting XML files especially for the time critical or dependent applications. One of the methods proposed for improving the performance is to process the data as a stream.

(b) Issues in XML encryption

Takeshi Imamura et al discuss next. another problem involving drawback of performance in parsing decrypted data. As the methods in the APIs such as DOM and SAX only provide methods for parsing complete XML documents, to process the decrypted data we need to use other methods. This could happen because sometimes at the application level we need to process part of the XML file rather than the whole file. The writer has considered Xerces2 (an XML parser developed by Apache XML project) as capable of doing this task. It has mentioned in the paper that the author has been able to get a reduction of 0.27% - 26% in processing time for encryption of XML files larger than 2kB and 34% - 88% reduction for decryption.

In the content of the paper the author has given an outline of the XML encryption specification. As for the syntax, the encrypted data is shown in an "Encrypted" element. Then he has discussed the process for encryption and the decryption of data.

(c) Xerces2 a concrete implementation for XML encryption

Xerces2 is an XML parser developed for the Apache XML project. As having an extensible architecture, it can be built or configured into many usages. Then the author discusses the architecture of the Xerces2 and the Xerces Native Interface (XNI). The XNI has been compared and differed with the XAS. Next the stream-based implementation is discussed. The author has stated that they have prototyped a stream-based implementation of the XML encryption specification. The section describes its design and the development environment. As the author describes the architecture is a pipe-filter system which will use events to perform the given tasks.

When the performance evaluation of the project is compared against the DOM based implementation, it shows that the stream-based implementation in general is showing a better performance than the DOM based implementations. The graphs have been provided with the retained results. And the graphs show that for files smaller than 2KB, the performance has some issue. The author suggests that this is due to the static overhead of the process.

Finally, to conclude, the performance results which I have mentioned earlier and that the overhead problem in the encryption should be solved. And the other problem that needs to be solved is the buffering problem in decrypting.

VI. Issues and Problems in Modern Cryptosystems

It is apparent that the computability of the encryptions schemes and computability of reversing algorithms has a direct impact on the security of the encryption algorithms. In the following paragraphs such issues of breakability and security of modern cryptosystems in selected contexts are discussed.

A. Pairing Based Cryptography For Distributed and Grid Computing

(a) Introduction

Pairing based cryptography is based on the existence of efficiently computable non-degenerate bilinear maps which can be briefly described as follows. Let G_1 and G_2 be two cyclic multiplicative groups both of the same order n such that computing discrete logarithms in G_1 and G_2 is intractable. A bilinear pairing is a map ˆe: $G_1 \times G_1 \rightarrow$ G2 that satisfies the following properties:

1. Bilinearity: ˆe (a^x, b^y) = ˆe $(a, b)^{xy}$ for all a, b 2 G_1 and x, y.
2. Non-degeneracy: If g is a generator of G_1 then ˆe (g, g) is a generator of G_2.
3. Computability: The map ˆe is efficiently computable.

(b) Problems in security of distributed computing

There are two mechanisms for avoiding security in distributed computing using pairing based cryptography.

- Problem of trust delegation in distributed computing
- Achieving confidentiality over a large and distributed network.

(c) Solutions

Solution for trust delegation in distributed computing using chain signatures.

In this scenario there are j ordered distinct participants I_1, I_2 . . . I_j. The original user that initiated the request is I_1. The request m is passed from I_i to I_{i+1} along. Those divide into three categories Signing, Verification, and Security [10].

Achieving confidentiality of distributed system over distributed key agreement.

The problem of achieving confidentiality over distributed and broadcast networks is a huge problem. In most case, a message must be encrypted to a selected subset of users. To achieve scalability in this model, it must be able to dynamically share secret keys without interaction with other group members.

B. Physical Observable Cryptography

(a) Problem Definition

Attacks that surpass the information leakage via non mathematical security barriers

Some systems cannot protect against attacks that surpass the information leakage via electromagnetic fields, power consumption, etc. These attacks bypass the best mathematical security barriers. Physically observable cryptography is one of the solutions to this problem. This delivers cryptographic security against access to information which is leaked physically. This paper discusses three aspects. They are attacks which are fully adaptive to the leaked in information, invalidation of the basic and traditional cryptography for the physical observable attacks and construct pseudorandom generators which are capable of providing secure against all physical-observation attacks.

(b) Proposed Solution

Security models physically observable cryptography

The security models are based on the following axioms.

- Computation, and only computation, leaks information
- Same computation leaks different information on different computers
- Information leakage depends on the chosen measurement
- Information leakage is local
- All leaked information is efficiently computable from the computer's internal configuration.

Solution for attacks that surpass the information leakage via non mathematical security barriers is created using Computational model, Physical Security, model Assumptions, Reductions and Goals and considering the above axioms [10].

C. Attacking and Repairing the Win-zip Encryption Scheme

Win-zip is a compression tool for Microsoft Windows which also supports encryption of data. In the paper [11], it is mentioned that the "Advance Encryption – 2", encryption scheme used in Win-zip, has some points of insecurity. Also they have discussed the data and results from performing series of tests on the scheme. It emphasizes on the fact that the security of the system should be measured as a whole than component by component basis. There is also an information leakage of the encryption method where the original file name and other file data can be seen in plain text such that an adversary may use those data to guess the original data.

(a) Problems in Win-zip Encryption Scheme

- Information leakage from unencrypted metadata of the files.

Though the data is encrypted in the files, the metadata is not. Therefore an adversary may use this metadata to predict the data in the encrypted files. Therefore the information leakage should be a minimum, so that the guessing is minimal.

- An issue with the interaction between compression and encryption components of the program.

An attacker can exploit the fact that the information leakage and with some work he may be able to use the interaction between the compression methods and the AE-2 encryption scheme of the Win-zip program to gain access to the original data.

- An issue with the representation of the file names and their extensions.

As the file names can be known from the metadata, one can rename the file extensions of the files without any problem, thereby making problems of open files in Operating Systems such as Microsoft Windows where file extensions does matter.

- An issue when compressed file contains both encrypted and unencrypted files.

The compressed files can contain both encrypted and unencrypted files and as the user can't distinguish the difference between the files an attacker can control the unencrypted files of the archive without the suspicion of the user.

- An issue with the key collisions due to the fact that the maximum random files which can be achieved are 2^{32}.

As the encryption scheme can handle up to only 2^{32} keys, the keys tend to recur there by occurring collisions between keys [11].

VII. ADVANCEMENTS IN MODERN CRYPTOSYSTEMS AND RESEARCH TRENDS

Although not much advancement is happening in the already sound cryptosystems, the newly immerged cryptosystems which are still on research stage are developing and advancing fast. Two main fields of cryptosystems which have gained the attention of researches are the DNA Cryptosystems and the Elliptic Curve Cryptosystems.

A. Elliptic Curve Cryptosystems

(a) Computability Problems in Elliptic Curve Cryptosystems

When we have a deep look into the subject of computability and complexity in elliptic curves and cryptography, there are certain problems which can be considered as computability problems of cryptography.

Computability can be seen as a topic with a wide scope which can be adapted into different other topics as well. When we consider about the Cryptography, definitely there are some computability related problems in cryptography as well. The article [12] focuses on two major projects which are related computability which we will be discussing later.

(b) Symmetric vs. Public Key Cryptography and the Hybrid approach

In symmetric key schemes most of the times identical keys are used to encrypt the plain text and to decrypt the cipher text. Simply a single key is used to encrypt and decrypt. The users have to make sure that the key does not go into the hands of compromisers. If it leaks the cipher would be broken since there is no major difference between keys, sometimes a small transfer function is used to change the encryption key from decryption key. In public key encryption there are two keys per person. One is to encrypt the message and the other is to decrypt which is a public one. It ensures the identity of the person who sends the encrypted file since no one else has the private key to encrypt the message. .

TABLE 1
COMPARISON BETWEEN TRADITIONAL AND DNA CRYPTOGRAPHIC METHODS

	Security	Time Complexity	Storage Medium	Storage Capacity	Stability
Traditional Cryptography	One Fold	≥ few seconds	Computer (Silicon) Chips	1 gram of silicon chip carries 16 MB [23]	Dependant on implementation environments
DNA Cryptography	Two Fold	≥ few hours	DNA strands	1 gram of DNA carries 10^8 TB [22]	Dependant on environmental conditions

VIII. Conclusion

Cryptography is a technique used to communicate between two parties in a secret way that an observer observing the communication would not understand what is being communicated. Over the history many cryptosystems were developed. In the modern era public key encryption and symmetric key encryption are the most widely used cryptosystems. The security of un-breakability of these systems directly depends on the computational difficulties involved in reversing the encryption algorithms without knowing some key data. Also there are issues in using current cryptosystems in some information systems as distributed computing and etc. Research is carried out to provide efficient solutions to these issues as well as to come out with new cryptosystems. Two new major cryptosystems which have immerged lately are elliptic curve cryptosystems and DNA cryptosystems. It is evident that there is a huge potential of research in these newly immerged areas.

It is seen that Cryptography and the related computational techniques have evolved throughout the history and will continue to evolve into the future as well. It is one of the most interesting topics and a field of great importance to carryout intensive research and advance on.

REFERENCES

[1] Cypher Research Laboratories Pty.Ltd. (2006) Cipher Research Laborotaries.[Online].
Available: http://www.cypher.com.au/crypto_history.htm

[2] (2012, January) Wikipedia - History of Cryptography. [Online].
Available: http://en.wikipedia.org/wiki/History_of_cryptography

[3] (2011, December) Wikipedia - Classical Ciphers. [Online].
Available: http://en.wikipedia.org/wiki/Classical_cipher

[4] R. Toemeh and S. Arumugam, "Breaking Transposition Cipher with Genetic Algorithm," ELECTRONICS AND ELECTRICAL ENGINEERING, vol. 79, 2007.

[5] Jayadev Misra, "Cryptography and Secure Communication," University of Texas, 2003.

[6] Edward Ruggeri, "Cryptography In An Unconventional Model,"

[7] Cristian Lupu, Bogdan Firtat, and Claudia Enoiu, "Cryptography Methods Using The RSA Algorithm," National Institute for Research and Development in Microtechnologies, Bucharest, Romania..

[8] Beenish Anam, Kazi Sakib, Md. Alamgir Hossain, and Keshav Dahal, "Review on the Advancements of DNA Cryptography," School of Computing, The Bradford University, West Yorkshire, UK, 2010.

[9] Takeshi Imamura, Andy Clark, and Hiroshi Maruyama. (2002) A steam-based implementation of XML encryption. [Online].
Available:http://dl.acm.org/citation.cfm?id=764792.764795&coll=DL&dl=ACM&CFID=63185751&CFTOKEN=42289310

[10] Amitabh Saxena and Ben Soh. (2006, July) Pairing Based Cryptography For. [Online].
Available:http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.90.371&rep=rep1&type=pdf

[11] Tadayoshi Kohno. (2004) Attacking and repairing the winZip encryption scheme. [Online].
Availble:http://dl.acm.org/citation.cfm?id=1030083.1030095&coll=DL&dl=ACM&CFID=63185751&CFTOKEN=42289310

[12] Liljana Babinkostova. Computability and Complex in Elliptic Curves and Cryptography. [Online].
Available: http://math.boisestate.edu/reu/projects/EllipticCurves-AES.pdf

[13] Leonid Reyzin. Silvio Micali. (2003, September) Physically Observable Cryptography. [Online].
Available: http://www.cs.bu.edu/fac/reyzin/papers/physec.pdf

[14] J. Chen, "A DNA-based, biomolecular cryptography design," in IEEE International Symposium on Circuits and Systems (ISCAS)., 2003, pp. 822–825.

[15] P. Gwynne and G. Heebner, "Technologies in DNA chips and microarrays: I," Science, vol. 4, p. 949, May 2001.