AN INTRODUCTION TO CRYPTOGRAPHY

Submitted by

ABIN K G

Register No: 170021032391

RIYA JOSHY

Register No: 170021032423

SREELAKSHMI V B

Register No: 170021032435

Under the guidance of

Dr. Pramada Ramachandran

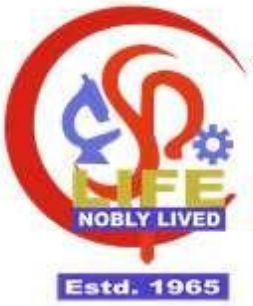
In partial fulfilment of the requirement for the award of
BACHELOR DEGREE OF SCIENCE in MATHEMATICS

2017-2020

DEPARTMENT OF MATHEMATICS

ST. PAUL'S COLLEGE, KALAMASSERY

(AFFILIATED TO M G UNIVERSITY, KOTTAYAM)



CERTIFICATE

This is to certify that the project report titled "AN INTRODUCTION TO CRYPTOGRAPHY" submitted by SREELAKSHMI V B(Reg no. 170021032435), RIYA JOSHY(Reg no. 170021032423) and ABIN K G(Reg. no:170021032391) towards partial fulfilment of the requirements for the award of Degree of Bachelor of Science in Mathematics is a bonafide work carried out by them during the academic year 2017-2020.

Supervising Guide

Dr. Pramada Ramachandran

Assistant Professor
Department of Mathematics

Head of the Department

Dr. Savitha K S

Assistant Professor
Department of Mathematics

Place: Kalamassery

Date:

Examiner:

DECLARATION

We ,SREELAKSHMI V B (Reg. no:170021032435), RIYA JOSHY(Reg. no:170021032423) and ABIN K G(Reg. no:170021032391) hereby declare that this project entitled "AN INTRODUCTION TO CRYPTOGRAPHY" is an original work done by us under the supervision and guidance of Dr. Pramada Ramachandran, faculty, Department of Mathematics in St. Paul's college Kalamassery in partial fulfilment for the award of The Degree of Bachelor of Science in Mathematics under Mahatma Gandhi University. We further declare that this project is not partly or wholly submitted for any other purpose and the data included in the project is collected from various sources and are true to the best of our knowledge.

Place: Kalamassery SREELAKSHMI V B

Date: RIYA JOSHY

ABIN K G

ACKNOWLEDGEMENT

For any accomplishment or achievement, the prime requisite is the blessing of the Almighty and it's the same that made this world possible. We bow to the lord with a grateful heart and prayerful mind.

It is with great pleasure that we express our sincere gratitude to our beloved teacher Dr. Pramada Ramachandran, Department of Mathematics, St. Paul's College, for her overwhelming support, motivation and encouragement.

We would like to acknowledge our deep sense of gratitude to Dr. Savitha K S, Head of Department of Mathematics and all the faculty members of the department and our friends who helped us directly and indirectly through their valuable suggestions and self-criticisms, which came a long way in ensuring that this project becomes a success.

We also express our sincere gratitude to Ms. Valentine D'Cruz, Principal, St. Paul's College, Kalamassery for the support and inspiration rendered to us in this project report.

ABSTRACT

Modern Cryptography relies heavily on concepts from mathematics. In this project we will be discussing several cryptographic ciphers and discovering the mathematical applications which can be found by exploring them. We begin with a review of the background material which will be needed before delving into the cryptographic ciphers. This project lends itself to be accessible to a person interested in learning about mathematics in cryptography on their own.

CONTENTS

INTRODUCTION

Cryptography is the science of using mathematics to encrypt and decrypt data. In the language of cryptography, where codes are called ciphers, the information to be concealed is called plaintext. After transformation to a secret form, a message is called ciphertext. The process of converting from plaintext to ciphertext is said to be encrypting, whereas the reverse process of changing from ciphertext back to plaintext is called decrypting.

Cryptography is an interdisciplinary subject, drawing from several fields especially mathematics. Nowadays cryptography makes extensive use of technical areas of mathematics, specifically those areas collectively known as discrete mathematics. Modern cryptography relies heavily on concepts from number theory.

Cryptography makes secure web sites and electronic safe transmissions possible. Due to the large number of commercial transactions on the internet, cryptography is very key in ensuring security of transactions. Cryptography is also used in access control to regulate access such as in cable TV and satellite. Without cryptography, hackers could get into our e-mail, listen in on our phone conversations, or break into banks/brokerage accounts. Hence in general, cryptography is an important way of achieving confidentiality, data integrity, user authentication and non-repudiation.

CHAPTER 1

SOME TOPICS IN ELEMENTARY NUMBER THEORY

In this section, we will be outlining several topics from number theory which we will need in order to explore the mathematics behind the cryptographic ciphers.

1.1 Divisibility and the Euclidean algorithm

Divisors and Divisibility: Given integers a and b with $a \neq 0$, we say that a divides b (or "b is divisible by a") and we write $a|$b if there exists an integer d such that

b= ad. In that case we call a a divisor of b. By a proper divisor of b, we mean a positive divisor not equal to b itself, and by a non-trivial divisor of b, we mean a positive divisor not equal to 1 or b.

Lemma 1.1.1. Suppose we have two integers a and b with a common divisor d≠0. That is, d|a and d|b ,then we will have d|$(ra + sb)$ for any integers r and s.

Proof. Because d is a divisor of both a and b, we can write $a = dj$ and b= dk for some integers j and k. Then

ra+sb= r(dj)+s(dk)=d(rj+sk). Since (rj+sk) is an integer then it follows that d|(ra + sb).

Proposition 1.1.1. Given two non- negative integers a and b, with a≠0, there exists a pair of unique integers q and r with 0≤ r < a such that b=aq+ r. We call q the quotient and r the remainder when b is divided by a.

Greatest common divisor: Given two integers a and b, not both zero, the greatest common divisor of a and b, denoted g.c.d.(a,b) is the largest integer d dividing both a and b. If the greatest common divisor of a and b is 1, then we say that a and b are relatively prime.

The Euclidean Algorithm.

Euclidean algorithm is useful for computing the g.c.d of two positive integers. Suppose we have two positive integers a and b, with $a >$b. The Euclidean algorithm works as follows. To find g.c.d.(a,b), we first divide b into a and write down the quotient q_1 and the remainder r_1: $a = q_1 b + r_1$. Next, we perform a second division with b playing the role of a and r_1 playing the role of b; b= $q_2 r_1 + r_2$. Next, we divide r_2 $into$ r_1: $r_1 = q_3 r_2 + r_3$. We continue in this way, each time dividing the last remainder into the second-to-last remainder, obtaining a new quotient and remainder. When we finally obtain a remainder that divides the previous remainder, we are done: that final nonzero remainder is the greatest common divisor of a and b.

Example 1. Find g.c.d (1547,560).

Solution: 1547= 2. 560 + 427

560 = 1. 427 + 133

427 = 3. 133 + 28

133 = 4. 28 + 21

28 = 1. 21 + 7

Since 7/21, we are done: g.c.d(1547,560) = 7

Definition: A prime number is an integer greater than one which has no positive divisors other than 1 and itself. A number is called composite if it has at least one non-trivial divisor.

Theorem 1.1.1–The Fundamental Theorem of Arithmetic

Given any natural number n, n can be written uniquely (except for the order of factors) as a product of prime numbers.

Example. $4200 = 2^3.3.5^2.7$

Proposition 1.1.2. Let d= g.c.d(a,b) where $a > b$. Then there exist integers u and v such that d= ua +bv. In other words, the g.c.d of two numbers can be expressed as a linear combination of the numbers with integer coefficient.

Example 2. From example 1, g.c.d(1547,560) =7. Express 7 as a linear combination of 1547 and 560.

Solution: To express 7 as a linear combination of 1547 and 560, we successively compute:

$$
\begin{aligned}
7 &= 28 - 1.\ 21 \\
&= 28 - 1.\ (133 - 4.\ 28) \\
&= 5.\ 28 - 1.\ 133 \\
&= 5.\ (427 - 3.\ 133) - 1.\ 133 \\
&= 5.\ 427 - 16.\ 133 \\
&= 5.\ 427 - 16.\ (560 - 1.427) \\
&= 21.\ 427 - 16.\ 560 \\
&= 21.\ (1547 - 2.\ 560) - 16.\ 560 \\
&= 21.\ 1547 - 58.\ 560
\end{aligned}
$$

Definition: We say that two integers a and b are relatively prime (or " a is prime to b") if g.c.d(a, b) =1, i.e., if they have no common divisor greater than 1.

1.2 Congruences

Given three integers a, b and m, we say that "a is congruent to b modulo m" and write $a \equiv$ b mod m, if the difference $a - b$ is divisible by m. The following properties are easily proved directly from the definition:

1. (i) $a \equiv a \bmod m$

   (ii) $a \equiv b \bmod m$ iff $b \equiv a \bmod m$

(iii) If $a \equiv b \bmod m$ and $b \equiv c \bmod m$, then $a \equiv c \bmod m$.

2. If $a \equiv$ b mod m and $c \equiv d \bmod m$, then $a \pm c \equiv b \pm d \bmod m$ and $ac \equiv b\,d \bmod m$. In other words, congruences with the same modulus can be added, subtracted or multiplied.
3. If $a \equiv b \bmod m$, then $a \equiv b \bmod d$ for any divisor d$|m$.
4. If $a \equiv b \bmod m$, $a \equiv b \bmod n$, and m and n are relatively prime, then $a \equiv b \bmod mn$.
5. In property 1,(i)-(iii) mean that the congruence modulo m is an equivalence relation. For fixed m, each equivalence class with respect to congruence modulo m has one and only one representative between 0 and m-1. The set of equivalence classes (called residue classes) will be denoted Z/mz. Any set of representatives for the residue classes is called a complete set of residues modulo m.

Proposition 1.2.1. The elements of Z/mz which have multiplicative inverses are those which are relatively prime to m, i.e., the numbers a for which there exists b with $ab \equiv 1 \bmod m$ are precisely those a with g.c.d$(a, m) = 1$.

Proof: First, if d=g.c.d(a,m) were greater than 1, we could not have $ab \equiv 1 \bmod m$ for any b, because that would imply that d divides ab-1 and hence divides 1.

Conversely, if g.c.d(a, m)=1, then by property 5, we may suppose that $a < m$. Then by Proposition 1.1.2, there exist integers u and v for which ua+vm=1. Choosing b=u, we see that m|1-ua=1-ab, as desired.

Example. Find 160^{-1} mod 841, i.e., the inverse of 160 modulo 841.

Solution: By Euclidean algorithm, we have

841= 5. 160 + 41

160= 3. 41 + 37

41= 1. 37 + 4

37= 9. 4 + 1

Hence, g.c.d(160,841)=1 and by Proposition 1.2.1, inverse of 160 modulo 841 exists. To find the inverse, we express 1 as a linear combination of 160 and 841 as follows:

1= 37 – 9. 4

= 37 – 9. (41 – 1. 37)

= 10. 37 – 9. 41

= 10. (160 – 3. 41) – 9. 41

= 10. 160 – 39. 41

= 10. 160 – 39. (841 – 5. 160)

= 205. 160 – 39. 841

Hence the answer is 205.

Proposition 1.2.2. (Fermat's Little Theorem).

Let p be a prime. Any integer a satisfies $a^P \equiv a \bmod p$, and any integer a not divisible by p satisfies

$a^{P-1} \equiv 1 \bmod p$.

Proof: First suppose that p$\nmid a$. We first claim that the integers $0a, 1a, 2a, 3a, \ldots (p-1)a$ are a complete set of residues modulo p. To see this, we observe that otherwise two of them say ia and ja, would have to be in the same residue class, i.e., $ia \equiv ja \bmod p$. But this would mean that p|(i - j)a, and since a is not divisible by p, we would have

P| (i – j). Since i and j are both less than p, the only way this can happen is if i=j. We conclude that the integers $a, 2a, \ldots, (p-1)a$ are simply a rearrangement of 1, 2,...,p-1 when considered modulo p. Thus, it follows that the product of the numbers in the first sequence is congruent modulo p to the product of the numbers in the second sequence, i.e.,

a^{P-1}(p-1)!$\equiv (p-1)!\, mod\ p$. Thus, p|(p-1)!$(a^{P-1}-1)$. Since (p-1)! is not divisible by p, we have p$|(a^{P-1}-1)$, as required. Finally, if we multiply both sides of the congruence $a^{p-1} \equiv 1 \bmod p$ by a, we get $a^P \equiv a \bmod p$ when a is not divisible by p. But if a is divisible by p, then this congruence $a^P \equiv a \bmod p$ is trivial. This concludes the proof of the proposition.

Proposition 1.2.3.(Chinese remainder theorem)

Suppose that we want to solve a system of congruences to different moduli; x$\equiv a_1\ mod\ m1$,

x≡ $a_2 mod\ m2$,

.....

x≡ $a_r \bmod m_r$.

Suppose that each pair of moduli is relatively prime; g.c.d(m_i, m_j)=1 for $i \neq j$. Then there exists a simultaneous solution x to all of the congruences, and any two solutions are congruent to one another modulo M=$m_1 m_2 \ldots m_r$.

Proof: First, we prove uniqueness modulo M. Suppose that x' and x'' are two solutions. Let x= x' – x''. Then x must be congruent to 0 modulo each m_i, and hence modulo M(by property 4 of congruences). We next show how to construct a solution x.

Define Mi =M/ m i to be the product of all of the moduli except for the i^{th}. Clearly g.c.d(m_i, M_i)=1, and so there is an integer Ni such that $M_i N_i \equiv 1 \bmod m_i$. Now set x= $\sum_i\ a_i M_i N_i$. Then for each i, we see that terms in the sum other than the i^{th} term are all divisible by m_i, because $m_i \mid$ Mj whenever i ≠ j. thus for each i, we have

x≡ $\ a_i M_i N_i \equiv a_i \bmod m_i$, as desired.

Definition. The Euler ϕ-function,$\phi(n)$, is defined to be the number of positive integers less than or equal to n which are relatively prime to n.

Proposition 1.2.4. the Euler ϕ-function is multiplicative, meaning that $\phi(mn) = \phi(m) \cdot \phi(n)$ whenever g.c.d(m,n)=1.

Proof: we must count the number of integers between 0 and mn-1 which have no common factor with mn. For each j in that range, let j_1 be its least non-negative residue modulo m (i.e. $0 \leq j_1 <$m and j$\equiv j_1$mod m) and let j_2 be its least non-negative residue modulo n (i.e. $0 \leq j_2 <$n and j$\equiv j_2$mod n). It follows from the Chinese reminder theorem that for each pair j_1, j_2 there is one and only one j between 0 and mn -1 for which j$\equiv j_1$mod m and j$\equiv j_2$mod n. j has no common factor with mn iff it has no common factor with m which is equivalent to j_1 having no common factor with m and it has no common factor with n which is equivalent to j_2 having no common factor with n. thus the j's which we must count are in one-to-one correspondence with the pairs j_1, j_2 for which $0 \leq j_1 <$m, g.c.d(j_1,m)=1; $0 \leq j_2 <$n and g.c.d (j_2,n)=1. The number of possible $j_1's$ is $\phi(m)$ and the number of possible j_2's is $\phi(n)$. So the number of pairs is $\phi(m) \cdot \phi(n)$. Hence the proof.

Since every n can be written as a product of prime powers, each of which has no common factors with the others, and since we know the formula $\phi(P^{\alpha}) = P^{\alpha}\left(1 - \frac{1}{p}\right)$, we can use the above proposition to conclude that for

n=$P_1^{\alpha_1} P_2^{\alpha_2} \ldots P_r^{\alpha_r}$,

$$\phi(n) = P_1^{\alpha_1}\left(1 - \frac{1}{P1}\right) P_2^{\alpha_2}\left(1 - \frac{1}{P2}\right) \ldots P_r^{\alpha_r}\left(1 - \frac{1}{Pr}\right)$$

$$= n\prod_{P/n}\left(1 - \frac{1}{p}\right)$$

Proposition 1.2.5. If g.c.d$(a, m) = 1$, then

$$a^{\phi(m)} \equiv 1 \bmod m.$$

Proof: We first prove the proposition in the case when m is a prime power: m=P^{α}.We use induction on α. The case α=1 is precisely Fermat's little theorem. Suppose that $\alpha \geq 2$, and the formula holds for the $(\alpha - 1)^{\text{th}}$ power of p. then $a^{P^{\alpha-1}-P^{\alpha-2}}$= 1 + $P^{\alpha-1}b$ for some integer b, by the induction assumption. Raising both sides of this equation to the p-th power and using the fact that the binomial coefficients in$(1 + x)^P$ are each divisible by p (except in the 1 and x^P at the ends), we see that $a^{P^{\alpha}-p^{\alpha-1}}$ is equal to 1 plus a sum with each term divisible by P^{α}. That is, $a^{\phi(P^{\alpha})}$-1 is divisible by P^{α}, as desired. This proves the proposition for prime powers.

Finally, by the multiplicativity of ϕ, it is clear that

$a^{\phi(m)} \equiv 1$ modP^{α}. Since this is true for each P^{α} which is the highest power of p dividing m, and since the different prime powers have no common factors with one another, it follows by property 4 of congruences that

$$a^{\phi(m)} \equiv 1 \bmod m.$$

Modular exponentiation by repeated squaring method

A basic computation one often encounters in modular arithmetic is finding $b^n \bmod m$ when both m and n are very large. There is a clever way of doing this that is much quicker than repeated multiplication of b by itself.

In what follows, we shall assume that b< m, and that whenever we perform a multiplication, we then immediately reduce mod m (i.e., replace the product by its least non negative residue). In that way, we never encounter any integers greater than m^2. We now describe the algorithm.

Use a to denote the partial product. When we are done, we will have a equal to the least non negative residue of $b^n \bmod m$. We start out with a=1.

Let $n_0, n_1, \ldots, n_{k-1}$ denote the binary digits of n, i.e. n=n_0+2n_1+ 4n_2+...+$2^{k-1}n_{k-1}$. Each n_j is 0 or 1. If n_0=1, change a to b. Then square b, and set

$b_1 = b^2$ mod m (ie, $b_1$is the least non negative residue of b^2 mod m). If n_1=1, multiply a by b_1 ,otherwise keep a unchanged. Next square b_1, and set

$b_2 \equiv b_1^2 \bmod m$.If n_2=1,multiply a by b_2, otherwise keep a unchanged. Continue in this way. You see that in the j^{th} step, you have computed $b_j \equiv b^{2^j} \bmod$ m. If n_j=1, i.e., if 2^j occurs in the binary expansion of n, then you include b_j in the product for a. After (k-1)-st step you'll have the desired $a \equiv b^n \bmod m$.

Chapter 2

CRYPTOGRAPHY

2.1 Some Simple Cryptosystems

Basic notions. Cryptography is the study of methods of sending messages in disguised form so that only the intended recipients can remove the disguise and read the message. The message we want to send is called the plaintext and the disguised message is called ciphertext. The plaintext and ciphertext are written in some alphabet(usually, but not always, they are written in the same alphabet) consisting of a certain number N of letters. The term “letter” (or “character”) can refer not only to the familiar A-Z, but also to numerals, blanks, punctuation marks, or any other symbols that we allow ourselves to use when writing the messages(if we don’t include a blank, for example, then all of the words are run together, and the messages are harder to read). The process of converting a plaintext to a ciphertext is called enciphering or encryption, and the reverse process is called deciphering or decryption.

The plaintext and ciphertext are broken up into ‘message units’. A message unit might be a single letter, a pair of letters(digraph), a triple of letters(trigraph) or a block of 50 letters. An enciphering transformation is a function that takes any plaintext message unit and gives us a ciphertext message unit.

In other words, it is a map f from the set P of all possible plaintext message units to the set C of all possible ciphertext message units. We shall always assume that f is a 1-to-1 correspondence, i.e., given a ciphertext message unit, there is one and only one plaintext message unit for which it is the encryption. The deciphering transformation is the map f^{-1} which goes back and recovers the plaintext from the ciphertext. We can represent the situation schematically by the diagram

$$P \overset{f}{\rightarrow} C \overset{f^{-1}}{\rightarrow} P$$

Any such set-up is called a 'cryptosystem'.

The first step in inventing a cryptosystem is to "label" all possible plaintext message units and all possible ciphertext message units by means of mathematical objects from which functions can be easily constructed. These objects are often simply the integers in some range. For example, if our plaintext and ciphertext message units are single letters from the 26-letter alphabet A-Z, then we can label the letters using the integers 0,1, 2, ...,25, which we call their "numerical equivalents". Thus, in place of A we write 0, in place of S we write 18, in place of X we write 23, and so on. As another example, if our message units are digraphs (i.e. pair of letters) in the 27-letter alphabet consisting of A-Z and a blank, we might first let the blank have numerical equivalent 26(one beyond Z), and then label the digraph

whose two letters correspond to x , y ∈ {0,1,2, … ,26} by the integer 27x + y ∈ {0,1, … ,728}.

Thus, we view the individual letters as digits to the base 27 and we view the digraph as a 2-digit integer to that base. For example, the digraph "NO" corresponds to the integer 27. 13 + 14=365. Analogously, if we were using trigraphs as our message units, we could label them by integers

729x +27y + z∈ {0,1, … ,19682}. In general, we can label blocks of k letters in an N-letter alphabet by integers between 0 and $N^k - 1$ by regarding each such block as a k-digit integer to the base N.

In some situations, one might want to label message units using other mathematical objects besides integers-for example, vectors or points on some curve. But we shall only consider integers throughout this section. Let us start with the case when we take a message unit (of plaintext or ciphertext) to be a single letter in an N-letter alphabet labeled by the integers 0, 1, 2, ...,N-1. Then by definition, an enciphering transformation is a rearrangement of these N integers.

2.1.2 Shift Transformation

Suppose we are using the 26-letter alphabet A-Z with numerical equivalents 0-25. Let the letter P∈ {0,1,2, … ,25} stand for a plaintext message unit. Define a function f from the set {0,1,2, … ,25} to itself by the rule, f(P)=$\begin{cases} P+3, if\ x < 23 \\ P-3, if\ x \geq 23 \end{cases}$

In other words, f simply adds 3 modulo 26: f(P)≡P+3 mod 26. Thus, with this system, to encipher the word "YES" we first convert to numbers: 24 4 18, then add 3 modulo 26:1 7 21, then translate back to letters: "BHV". To decipher a message, one subtracts 3 modulo 26. For example, the ciphertext "ZKB" yields the plaintext "WHY". This cryptosystem was apparently used in ancient Rome by Julius Caesar, who supposedly invented it himself.

Example given above can be generalized as follows. Suppose we are using an N-letter alphabet with numerical equivalents

0, 1, 2, ..., N-1. Let b be a fixed integer. By a shift transformation we mean the enciphering function f defined by the rule C= F(p)≡P+b mod N. Julius Caesar's cryptosystem was the case N=26, b=3. To decipher a ciphertext message unit

C∈ $\{0,1,2,\dots,N-1\}$, we simply compute

P=$f^{-1}(C)$ ≡C– b mod N.

2.1.3 Cryptanalysis

Now suppose that you are not privy to the enciphering and deciphering information, but you would nevertheless like to be able to read the coded messages. This is called breaking the code, and the science of breaking codes is called cryptanalysis.

In order to break a cryptosystem, one needs two types of information. The first is the general nature (the structure) of the system. For example, suppose we know that the cryptosystem uses a shift transformation on single letters of the 26-letter alphabet A-Z with numerical equivalents 0-25 respectively. The second type of information is knowledge of a specific choice of certain parameters connected with the given type of

cryptosystem. In our example, the second type of information one needs to know is the choice of the shift parameter b. Once one has that information, one can encipher and decipher by the formulas $C \equiv P+b \bmod N$ and $P \equiv C- b \bmod N$.

We shall always assume that the general structural information is already known. In practice, users of cryptography often have equipment for enciphering and deciphering which is constructed to implement only one type of cryptosystem. Over a period of time the information about what type of system they are using might leak out. To increase their security, therefore, they frequently change the choice of parameters used with the system. For example, suppose that two users of the shift cryptosystem are able to meet once a year. At that time, they agree on a list of 52 choices of the parameter b, one for each week of the coming year. The parameter b (more complicated cryptosystems usually have several parameters) is called a key, or more precisely, the enciphering key.

Example. So suppose that we intercept the message "FQOCUDEM", which we know was enciphered using a shift transformation on single letters of the 26-letter alphabet. It remains for us to find the b. One way to do this is by frequency analysis. This works as follows. Suppose that we have already intercepted a long string of ciphertext, say several hundred letters. We know that "E" is the most frequently occurring letter in the English language. So it is reasonable to assume that the most frequently occurring letter in the ciphertext is the encryption of E. Suppose that we find that "U" is the most frequently occurring character

in the ciphertext. That means that the shift takes “E”=4 to “U”=20, i.e., 20≡4 + b mod 26, so that b=16. To decipher the message, then, it remains for us to subtract 16 (working modulo 26) from the numerical equivalent of “FQOCUDEM”:

“FQOCUDEM” = 5 16 14 2 20 3 4 12 ⟼

15 0 24 12 4 13 14 22

= “PAYMENOW”.

2.1.4 Affine Transformations

In the case of a shift encryption of single letters of a 26-letter alphabet, it is not even necessary to have a long string of ciphertext to find the most frequently occurring letter. After all, there are only 26 possibilities for b, and one can simply run through all of them. Most likely, only one will give a message that makes any sense, and that b is the enciphering key.

Thus, this type of cryptosystem is too simple to be much good. It is too easy to break. An improvement is to use a more general type of transformation of Z/NZ, called an affine map: C≡ aP + b mod N, where a and b are fixed integers (together they form the enciphering key). For example, working again in the 26-letter alphabet, if we want to encipher our message “PAYMENOW” using the affine transformation with enciphering key a=7, b=12, we obtain:

15 0 24 12 4 13 14 22⟼ 13 12 24 18 14 25 6 10

= “NMYSOZGK”.

To decipher a message that was enciphered by means of the affine map C≡ aP + b mod N, one simply solves for P in terms of C, obtaining P≡ $a'C + b'$ mod N, where a' is the inverse of a modulo N and b' is equal to - $a^{-1}b$. Note that this works only if g.c.d(a,N)=1; otherwise we cannot solve for P in terms of C. If g.c.d(a,N)>1, then it is easy to see that more than one plaintext letter will give the same ciphertext, so that we cannot uniquely recover the plaintext from the ciphertext. For example, if we were to encipher the message "PAYBACK" by means of the affine map

C≡ aP + b mod N where a=10, b=12, again in the 26-letter alphabet. Here g.c.d((a,N)= 2 >1 and we observe that the plaintext units "P" and "C" corresponds to ciphertext unit "G". By definition, that is not an enciphering transformation: we always require that the map be 1-to-1, i.e., that the plaintext be uniquely determined from the ciphertext. To summarize, an affine cryptosystem in an N-letter alphabet with parameters $a \in$ (Z/NZ)*and

b∈ (Z/NZ) consists of the rules:

C≡ aP + b mod N, P≡ $a'C + b'$ mod N,

Where a'=a^{-1} in (Z/NZ)* , b'= - $a^{-1}b$.

As a special case of the affine cryptosystems we can set a=1, thereby obtaining the shift transformations. Another special case is when b=0: P≡ aC mod N, C≡ a^{-1}P mod N. The case b=0 is called a linear transformation, meaning that the map takes a sum to a sum, i.e., if C_1 is the encryption of P_1

and C_2 is the encryption of P_2, then C_1+C_2 is the encryption of P_1+P_2 (where, of course, we are adding modulo N).

Now suppose that we know that an intercepted message was enciphered using an affine map of single letters in an N-letter alphabet. We would like to determine the enciphering key a, b so that we can read the message. We need to know two bits of information to do this.

Example 1. Still working in our 26-letter alphabet, suppose that we know the most frequently occurring letter of ciphertext is "K", and the second most frequently occurring letter is "D". It is reasonable to assume that these are the encryptions of "E" and "T", respectively, which are the two most frequently occurring letters in the English language. Thus, replacing the letters by their numerical equivalents and substituting for P and C in the deciphering formula, we obtain:

$$10a' + b' \equiv 4 \bmod 26$$

$$3a' + b' \equiv 19 \bmod 26.$$

We have two congruences with two unknowns, a' and b'. The quickest way to solve is to subtract the two congruences to eliminate b'. We obtain $7a' \equiv 11 \bmod 26$, and $a' \equiv 7^{-1}11 \equiv 9 \bmod 26$. Finally, we obtain b' by substituting this value for a' in one of the congruences: $b' \equiv 4 - 10a' \equiv 18 \bmod 26$. So, messages can be deciphered by means of the formula P≡ 9C + 18 mod 26.

Example 2. You are trying to cryptanalyze an affine enciphering transformation of single-letter message units in a 37-letter alphabet. This alphabet includes the numerals 0-9, which are labeled by themselves (i.e., by the integers 0-9). The letters A-Z have numerical equivalents 10-35, respectively, and blank=36(indicated by an "_" for understanding). You intercept the ciphertext "OH7F86BB46R3627O266BB9" (here the O's are the letter "oh", not the numeral zero). You know that the plaintext ends with signature "007" (zero zero seven). What is the message?

From the given information, we know that "B" is the encryption for "0" (zero), and "9" is the encryption for "7".

Thus, replacing the letters by their numerical equivalents and substituting for P and C in the deciphering formula, we obtain:

$$0 \equiv 11a' + b' \bmod 37$$

$$7 \equiv 9\,a' + b' mod\ 37$$

Subtracting the two congruences to eliminate b', we obtain: $2a' \equiv -7 \bmod 37$, and $a' \equiv (-7)2^{-1} \equiv 15 \bmod 37$.

Finally, we obtain b' by substituting this value for a' in one of the congruences: $b' \equiv 20 \bmod 37$. Now, we can decipher the message by the formula P≡ 15C + 20 mod 26, which reads as follows: "AGENT_006_IS_DEAD__007.

2.1.5 Digraph Transformations

We now suppose that our plaintext and ciphertext message units are two-letter blocks, called digraphs. For example, if our plaintext is "HELP", as a digraph it is represented

"HE LP". This means that the plaintext is split up into two-letter segments. If the entire plaintext has an odd number of letters, then in order to obtain a whole number of digraphs we add on an extra letter at the end; we choose a letter which is not likely to cause confusion, such as a blank if our alphabet contains a blank, or else "X" or "Q" if we are using just the 26-letter alphabet.

Each digraph is then assigned a numerical equivalent. The simplest way to do this is to take xN+y, where x is the numerical equivalent of first letter in the digraph, y is the numerical equivalent of the second letter in the digraph, and N is the number of letters in the alphabet. Equivalently, we think of a digraph as a 2-digit base-N integer. This gives a one-to-one correspondence between the set of all digraphs in the N-letter alphabet and the set of all non-negative integers less than N^2.

Next, we decide upon an enciphering transformation, i.e., a rearrangement of the integers {0,1,2,..., $N^2 - 1$}. Among the simplest enciphering transformations are the affine ones, where we view this set of integers as Z/N^2Z and define the encryption of P to be the nonnegative integer less than N^2 satisfying the congruence C≡ aP + b mod N^2.Here, as before, a must have no common factor with N(which

means it has no common factor with N^2), in order that we have an inverse transformation telling us how to decipher:

P$\equiv a'C + b' \bmod N^2$, where $a' \equiv a^{-1} \bmod N^2$,

$b' \equiv -a^{-1}b \bmod N^2$. We translate C into a two-letter block of ciphertext by writing it in the form C=$x'N + y'$, and then looking up the letters with numerical equivalents x' and y'.

Example. Suppose we are working in the 26-letter alphabet and using the digraph enciphering transformation $C \equiv 159P + 580 \bmod 676$. Then the digraph "NO" has numerical equivalent 13. 26 + 14 = 352 and is taken to the ciphertext digraph 159. 352 + 580≡ 440 mod 676, which is "QY" (since 440=16. 26 + 24). The digraph "ON" has numerical equivalent 14. 26 + 13=377, and is taken to

159. 377 + 580≡359= "NV".

Notice that the digraphs change as a unit, and there is no relation between the encryption of one digraph and that of another one that has a letter in common with it or even consists of the same letter in the reverse order.

To break a digraphic encryption system which uses an affine transformation C≡ aP + b mod N^2, we need to know the ciphertext corresponding to two different plaintext message units. Since the message units are digraphs, a frequency analysis means counting which two letter-blocks occur most often in a long string of ciphertext, and comparing with the known frequency of digraphs in English language texts. For example, if we use the 26-letter alphabet,

statistical analyses seem to show that “TH” and “HE” are the two most frequently occurring digraphs, in that order.

Example. You know that your adversary is using a cryptosystem with a 27-letter alphabet, in which the letters A-Z have numerical equivalents 0-25, and blank=26. Each digraph then corresponds to an integer between 0 and 728=$27^2 - 1$ according to the rule that, if the two letters have numerical equivalents x and y, then the digraph has numerical equivalent 27x + y, as explained earlier. Suppose that a study of a large sample of ciphertext reveals that the most frequently occurring digraphs are “ZA”, “IA”, and “IW”. Suppose that the most common digraphs in the English language are “E_” (i.e., E blank), “S_”, “_T”. You know that the cryptosystem uses an affine enciphering transformation modulo 729. Find the deciphering key, and read the message “NDXBHO”. Also find the enciphering key.

Solution. We know that plaintexts are enciphered by means of the rule C≡ aP + b mod 729, and that ciphertexts can be deciphered by means of the rule P≡ $a'C + b'$ mod 729; here a, b form the enciphering key, and a', b' form the deciphering key. We first want to find a' and b'. We know how three digraphs are deciphered, and, after we replace the digraphs by their numerical equivalents, this gives us the three congruences:

$$675a' + b' \equiv 134 \bmod 729$$

$$216a' + b' \equiv 512 \bmod 729$$

$$238a' + b' \equiv 721 \bmod 729.$$

If we try to eliminate b' by subtracting the first two congruences, we arrive at $459a' \equiv 351 \bmod 729$, which does not have a unique solution a' mod 729(there are 27 solutions). We do better if we subtract the third congruence from the first, obtaining $437a' \equiv 142 \bmod 729$. To solve this we must find the inverse of 437 modulo 729. By way of review of the Euclidean algorithm, lets go through that in detail:

$729 = 437 + 292$

$437 = 292 + 145$

$292 = 2.\ 145 + 2$

$145 = 72.\ 2 + 1$

And then

$1 = 145 - 72.\ 2$

$= 145 - 72(292 - 2.\ 145)$

$= 145.\ 145 - 72.\ 292$

$= 145(437 - 292) - 72.\ 292$

$= 145.\ 437 - 217.\ 292$

$= 145.\ 437 - 217(729 - 437)$

$\equiv 362.\ 437 \bmod 729.$

Thus, $a' \equiv 362.\ 142 \equiv 374 \bmod 729$, and then

$b' \equiv 134 - 675.\ 374 \equiv 647 \bmod 729$. Now applying the deciphering transformation to the digraphs “ND”, “XB” and “HO” of our message-they correspond to the integers

354, 622 and 203, respectively- we obtain the integers 365, 724 and 24. Writing 365= 13. 27 + 14, 724=26. 27 + 22, 24=0. 27 + 24, we put together the plaintext digraphs into the message "NO WAY". Finally, to find the enciphering key we compute $a \equiv a'^{-1} \equiv 374^{-1} \equiv 614 \bmod 729$(again using the Euclidean algorithm) and b$\equiv -a'^{-1} b'$

$\equiv -614.647 \equiv 47 \bmod 729.$

Remark. Although affine cryptosystems with digraphs (i.e. modulo N^2) are better than the ones using single letters (i.e. modulo N), they also have drawbacks. Notice that the second letter of each ciphertext digraph depends only on the plaintext digraph. This is because that second letter depends on the mod- N value of C$\equiv aP + b \bmod N^2$, which depends only on P modulo N, i.e., only on the second letter of the plaintext digraph. Thus, one could obtain a lot of information (namely, a and b modulo N) from a frequency analysis of the even-numbered letters of the ciphertext message.

2.2 Some Examples of Secrecy Systems

In this section a number of examples of ciphers is given.

1. Simple Substitution Cipher

In this cipher each letter of the message is replaced by a fixed substitute usually also a letter. Thus, the message, M= $m_1 m_2 m_3 m_4 \ldots$, where $m_1, m_2, m_3, \ldots$ are the successive letters becomes:

E= $e_1 e_2 e_3 e_4 \ldots = f(m_1)\, f(m_2)\, f(m_3) f(m_4) \ldots$

Where the function f(m) is a function with an inverse. The key is a permutation of the alphabet (when the substitutes are letters). An example key is:

Plain alphabet:
ABCDEFGHIJKLMNOPQRSTUVWXYZ

Cipher alphabet:
PHQGIUMEAYLNOFDXJKRCVSTZWB.

An example encryption using the above key:
Plaintext:
DEFEND THE EAST WALL OF THE CASTLE
Ciphertext:
GIUIFG CEI IPRC TPNN DU CEI QPRCNI.

2. Transposition (Fixed Period d)

The message is divided into groups of length d and a permutation applied to first group, the same permutation to the second group, etc. The permutation is the key and can be represented by a permutation of the first d integers. Thus, for d=5, we might have 2 3 15 4 as the permutation. This means that:
$m_1 m_2 m_3 m_4 m_5 m_6 m_7 m_8 m_9 m_{10} \ldots$
becomes $m_2 m_3 m_1 m_5 m_4 m_7 m_8 m_6 m_{10} m_9 \ldots$
Sequential application of two or more transpositions will be called compound transposition. If the periods are $d_1, d_2, \ldots, d_n$ then the result is a transposition of period d, where d is the least common multiple of $d_1, d_2, \ldots, d_n$.

3. Vigenere Cipher

The most famous example of a polyalphabetic cipher

(In a polyalphabetic cipher, a plaintext has more than one ciphertext equivalent) was published by the French cryptographer Blaise de Vigenere (1523-1596) in his *Traicte de Chiffres* of 1586.

To implement this system, the communicating parties agree on an easily remembered word or phrase. With the standard alphabet numbered from A=00 to Z=25, the digital equivalent of the keyword is repeated as many times as necessary beneath that of the plaintext message. The message is then enciphered by adding, modulo 26, each plaintext number to one immediately beneath it. The process may be illustrated with the keyword READY; whose numerical version is 17 04 00 03 24. Repetitions of this sequence are arranged below the numerical plaintext of the message

"ATTACK AT ONCE"

to produce the array

00 19 19 00 02 10 00 19 14 13 02 04

17 04 00 03 24 17 04 00 03 24 17 04

When the columns are added modulo 26, the plaintext message is encrypted as

17 23 19 03 00 01 04 19 17 11 19 08

or, converted to letters, "RXTDAB ET RLTI".

Notice that a given letter of plaintext is represented by different letters in the ciphertext. The double "T" in the word "ATTACK" no longer appears as a double letter when ciphered, while the ciphertext "R" first corresponds to "A" and then to "O" in the original message.

In general, any sequence of n letters with numerical equivalents $b_1, b_2, \ldots, b_i$ ($00 \leq b_i \leq 25$) will serve as the keyword. The plaintext message is expressed as successive blocks $P_1 P_2 \ldots P_n$ of n two-digit integers P_i, and then converted to ciphertext blocks $C_1 C_2 \ldots C_n$ by means of the congruences $Ci \equiv P_i + bi \bmod 26$, $1 \leq i \leq n$.

Decryption is carried out by using the relations

$P_i \equiv Ci - bi \bmod 26$, $1 \leq i \leq n$.

4. Hills's Cipher

This cipher was devised in 1929 by Lester Hill, an assistant professor of mathematics at Hunter college. Hills's approach is to divide the plaintext message into blocks of n letters (possibly filling out the last block by adding "dummy" letters such as X's) and then to encrypt block by block using a system of n linear congruences in n variables. In its simplest form, when n=2, the procedure takes two successive letters and transforms their numerical equivalents $P_1 P_2$ into a block $C_1 C_2$ of ciphertext numbers via the pair of congruences

$$C_1 \equiv a\, P_1 + bP_2 \bmod 26$$

$$C_2 \equiv c\, P_1 + dP_2 \bmod 26$$

To permit decipherment, the four coefficients a, b, c, d must be selected so g.c.d(ad – bc, 26)=1.

To illustrate Hills's cipher, let us use the congruences

$$C_1 \equiv 2\,P_1 + 3P_2 \bmod 26$$

$$C_2 \equiv 5\,P_1 + 8P_2 \bmod 26$$

To encrypt the message "BUY NOW". The first block "BU" of letters is numerically equivalent to 01 20. This is replaced by

$$2(01) + 3(20) \equiv 62 \equiv 10 \bmod 26$$

$$5(01) + 8(20) \equiv 165 \equiv 9 \bmod 26$$

Continuing two letters at a time, we find that the completed ciphertext is: 10 09 09 16 16 12 which can be expressed alphabetically as: "KJJ QQM".

Decipherment requires solving the original system of congruences for P_1 and P_2 in terms of $C_1\ and\ C_2$. The plaintext block P_1P_2 can be recovered from ciphertext block C_1C_2 by means of the congruence

$$P_1 \equiv 8C_1 - 3C_2 \bmod 26$$

$$P_2 \equiv -5C_1 + 2C_2 \bmod 26$$

From block 10 09 of ciphertext, we calculate

$$P_1 \equiv 8(10) - 3(09) \equiv 53 \equiv 01 \bmod 26$$

$$P_2 \equiv -5\,(10) + 2(09) \equiv -32 \equiv 20 \bmod 26$$

Which is the same as the letter "BU". The remaining plaintext can be restored in a similar manner.

5. Autokey Cipher

A Vigenere type system in which either the message itself or the resulting cryptogram is used for the "key" is called an autokey cipher. The encipherment is started with a "priming key" (which is the entire key in our sense) and continued with the message or cryptogram displaced by the length of the priming key as indicated below, where the priming key is "COMET". The message used as key:

Message: SENDSUPPLIES...
Key: COMETSENDSUP...
Cryptogram: USZHLMTCOAYH...

The cryptogram used as key:
Message: SENDSUPPLIES...
Key: COMETUSZHLMT...
Cryptogram: USZHLOHOSTSZ...

Chapter 3
PUBLIC KEY

3.1 The idea of Public Key Cryptography

The term "cryptosystem" is more often used to refer to a whole family of 1-to-1 enciphering transformation f from a set P of all possible plaintext message units to a set C of all possible ciphertext message units. Each transformation corresponds to a choice of parameters. For example, for a fixed N-letter alphabet (with numerical equivalents also fixed once and for all), we might consider the affine cryptosystem which for $a \in$ (Z/NZ)*and b$\in$ (Z/NZ) is the map P= Z/NZ to C= Z/NZ defined by $C \equiv aP + b \bmod N$. In this example, the sets P and C are fixed, but the enciphering transformation f depends on the choice of parameters a, b. The values of the parameters are called the enciphering key k_E. In our example, k_E is the pair $(a$, b).

In practice, we shall suppose that the algorithm is publicly known, i.e., the general procedure used to encipher cannot be kept secret. However, the keys can easily be changed periodically, and if one wants, kept secret. One also needs an algorithm and a key in order to decipher, i.e., compute f^{-1}. The key is called the deciphering key k_D. In our example of the affine cryptosystem family, deciphering is also accomplished by an affine map, namely $P \equiv a^{-1}c - a^{-1}b$ mod N, and so the deciphering transformation uses the same algorithm as the enciphering transformation, except with a different key, namely, the pair $(a^{-1}, -a^{-1}b)$.

We shall always suppose that the deciphering and enciphering algorithms are publicly known, and that it is the keys k_E and k_D which can be concealed.

By definition, a public key cryptosystem has the property that someone who knows only how to encipher cannot use the enciphering key to find the deciphering key without a prohibitively lengthy computation. In other words, the enciphering function f: P⟶C is easy to compute once the enciphering key k_E is known, but it is very hard in practice to compute the inverse function f^{-1}: C⟶P. That is, from the standpoint of realistic computability, the function f is not invertible (without some additional information- the deciphering key k_D). Such a function is called trapdoor function. That is, a trapdoor function is a function f which is easy to compute but whose inverse f^{-1} is hard to compute without having some additional auxiliary information beyond what is necessary to compute.

There is a closely related concept of a one-way function. This is a function f which is easy to compute but for which f^{-1} is hard to compute and cannot be made easy to compute even by acquiring some additional information. While the notion of a trapdoor function apparently appeared for the first time in 1978 along with the invention of RSA public key cryptosystem, the notion of a one-way function is somewhat older.

With a public key system, it is possible for two parties to initiate secret communications without ever having any prior contact, without having established any prior trust for one another, without exchanging any preliminary information. All of the information necessary to send an enciphered message is public.

3.2 Classical versus Public key

By a classical cryptosystem (also called a private key cryptosystem or a symmetrical cryptosystem), we mean a cryptosystem in which, once the enciphering information is known, the deciphering transformation can be implemented in approximately the same order of magnitude time as the enciphering transformation. All of the cryptosystems in Chapter 2 are classical. Occasionally, it takes a little longer for the deciphering-because one needs to apply the Euclidean algorithm to find an inverse modulo N or one must invert a matrix- nevertheless the additional time required is not prohibitive. However, in a private key cipher both the encryption key and decryption key must be kept secret from those who are not part of the communication at hand in order to ensure cipher's security. Because of this encryption/decryption keys must be generated for pairs of people each time they wish to communicate.

In contrast, public-key cryptosystem (also known as asymmetric cryptosystem) are developed in such a way that knowledge of the encryption key gives no information as to what the decryption key is. One advantage of public key cryptography is that, there is an especially easy way to identify oneself in such a way that no one could be simply pretending to be you. Let A(Alice) and B(Bob) be two users of the system let f_A be the enciphering transformation with which any user of the system sends a message to Alice and let f_B the same for Bob. For simply sitting, we shall assume that the set P of all possible plaintext message units and the

set C of all possible ciphertext message units are equal, and are the same for all users. Let P be Alice's "signature" (perhaps the time the message was sent etc.). It would not be enough for Alice to send Bob the encoded message f_B(P), since everyone knows how to do that, so there would be no way of knowing that the signature was not forged. Rather, at the beginning (or end) of the message Alice transmits $f_B\ f_A^{-1}$(P). Then, when Bob deciphers the whole message, including this part, by applying f_B^{-1} he finds that everything has become plaintext except for a small section of jibberish, which is f_A^{-1}(P). since Bob knows that the message is claimed to be from Alice, he applies f_A(which he knows, since Alice's enciphering key is public) and obtains P. Since no one other than Alice could have applied the function f_A^{-1} which is inverted by f_A, he knows that the message was from Alice.

3.3 RSA

One of the most well-known and widely used publicly cryptosystem is the RSA cryptosystem. It is named from the last names of the inventors Rivest, Shamir, and Adleman. The success of the so called "RSA" cryptosystem, which is one of the oldest and most popular public key cryptosystems is based on the tremendous difficulty of factoring.

We now describe how RSA works. Each user first chooses two extremely large prime numbers p and q (say of about 100 decimal digits each), and sets n=pq.

Knowing the factorization of q, it is easy to compute

$\phi(n)$=(p – 1)(q – 1)= n + 1 - p - q. Next, the user randomly chooses an integer e between 1 and $\phi(n)$ which is prime to $\phi(n)$. Whenever we say "random" we mean that the number was chosen with the help of a random number generator, i.e.,, a computer program that generates a sequence of digits in a way that no one could duplicate or predict, and which is likely to have all of the statistical properties of a truly random sequence. In the RSA cryptosystem, we need a random number generator not only to choose e, but also to choose the large primes p and q.

Thus, each user A chooses two primes p_A and q_A and a random number e_A which has no common factor with

$(p_A - 1)(q_A - 1)$. Next, A computes $n_A = p_A\, q_A$, $\phi(n_A) = n_A + 1 - p_A - q_A$, and also the multiplicative inverse of e_A modulo $\phi(n_A)$: $d_A = e_A^{-1} \bmod \phi(n_A)$. She makes public the enciphering key $k_{E,A} = (n_A, e_A)$ and conceals the deciphering key $k_{D,A} = (n_A, d_A)$. The enciphering transformation is the map from $\mathbb{Z}/n_A\mathbb{Z}$ to itself given by $f(P) \equiv P^{e_A} \bmod n_A$. The deciphering transformation is the map from $\mathbb{Z}/n_A\mathbb{Z}$ to itself given by $f^{-1}(C) \equiv C^{d_A} \bmod n_A$. These two maps are inverse to one another, because of our choice of d_A.

Namely, performing f followed by f^{-1} or f^{-1} followed by f means raising to the $d_A e a^{\text{th}}$ power. But because $d_A e a$ leaves a reminder of 1 when divided by $\phi(n_A)$, this is the same as raising to the 1-st power.

In practice, we would probably want to choose P and C uniformly throughout the system. For example, suppose we are working in an N-letter alphabet. Then let k< l be suitably chosen positive integers such that for example, N^k and N^l have approximately 200 decimal digits. We take as our plaintext message units all blocks of k-letters, which we regard as k- digit base N- integers, i.e., we assign them numerical equivalents between 0 and N^k . We similarly take ciphertext message units to be blocks of l-letters in our N-letter alphabet. Then each user must choose his/her large primes p_A and q_A so that $n_A = p_A\, q_A$ satisfies $N^k < n_A < N^l$.Then any plaintext message unit, i.e., integer less than N^k,corresponds to an element in Z/ n_AZ and since $n_A < N^l$, the image

f(P) ∈ Z/ n_AZ can be uniquely written as an l-letter block.

Example. For the benefit of simplicity in computation, we shall sacrifice realism and choose most of our examples so as to involve relatively small integers. Choose N=26, k= 3, l= 4. That is, the plaintext consists of trigraphs and the ciphertext consists of four graphs in the usual 26-letter alphabet. To send the message "YES" to a user A with the enciphering key

(n_A, e_A)=(46927, 39423), we first find the numerical equivalent of "YES", namely: 24.26^2+4.26+18=16346, and then compute 16346^{39423} mod 46927, which is

21166=1. 26^3+5. 26^2+8.26+2= "BFIC". The recipient A knows the deciphering key, (n_A, d_A)=(46927,26767), and so computes 21166^{26767} mod 46927=16346= "YES".

Let us see how use A generate her keys. First, she multiplied the primes p_A=281 and q_A=167 to get n_A; then she chose e_A at random [but subject to the condition that

g.c.d(e_A,280) =g.c.d(e_A,166)=1]. Then she found

d_A=e_A^{-1} mod 280.166. The numbers p_A, q_A and d_A remain secret.

Clearly, the most time-consuming step is modular exponentiation, eg. 16346^{39423} mod 46927. But this can be done by repeated squaring method.

Example. Suppose that a message is to be sent to an individual whose listed public key is (2701,47). The key was arrived at by selecting the two primes p=37 and q=73, which in turn led to the enciphering modulus n=37.73=2701 and $\phi(n)$=36.72=2592. Because g.c.d(47,2592)=1, the integer k=47 was taken as the enciphering exponent.

The message to be encrypted and forwarded is "NO WAY TODAY". It is first translated into a digital equivalent using the previously indicated letter substitutions

M=13 14 26 22 00 24 26 19 14 3 00 24.

This plaintext number is thereafter expressed as four-digit blocks: 1314 2622 0024 2619 1403 0024. The corresponding ciphertext numbers are obtained by raising each block to the 47th power and reducing the results modulo 2701. In the first block, repeated squaring produces the value 1314^{47} ≡1241 mod 2701. The completed encryption of the message is the list

1241 1848 0873 1614 2081 0873.

For the deciphering operation, the recipient employs the Euclidean algorithm to obtain the equation

47.1103+2592(-20)=1, which is equivalent to

47.1103≡1 mod 2592. Hence, j=1103 is the recovery exponent. It follows that $1241^{1103} \equiv 1314 \bmod 2701$ and so on.

Remarks

1. In choosing p and q, user A should take care to see that certain conditions hold. The most important are: that the two primes not be too close together, and p-1 and q-1 have a fairly small g.c.d and both have at least one large prime factor.
2. While discussing authentication in a previous section, we assumed for simplicity P=C. We have slightly more complicated set-up in RSA. Here is one way to avoid the problem of different n_A's and different block sizes (k, the number of letters in a plaintext message unit, being less than l, the number of letters in a ciphertext message unit). Suppose that, Alice is sending her signature to Bob. She knows Bob's enciphering key $k_{E,B}=(n_B, e_B)$ and her own deciphering key $k_{D,A}=(n_A, e_A)$. What she does is send $f_B\ f_A^{-1}(\mathrm{P})$ is $n_A < n_B$ or else $f_A^{-1}\ f_B(\mathrm{P})$ if $n_A > n_B$. That is, in the former case she takes the least positive residue of P^{d_A} modulo n_A then regarding that number modulo n_B, she computes P^{e_A} modulo n_B and then, working modulo n_A, she raises this to the d_A-th power.

Clearly, Bob can verify the authenticity of the message in the first case by raising to the d_B-th power mod n_B and then to the e_A-th power mod n_A; in the second case he does the operation in the reverse order.

Chapter4
APPLICATIONS

4.1 Cryptography in Everyday Life

Authentication/Digital Signatures:

Authentication is any process through which one proves and verifies certain information. Sometimes one may want to verify the origin of a document, the identity of the sender, the time and date a document was sent and/or signed, the identity of a computer or user, and so on. A digital signature is a cryptographic means through which many of these may be verified. The digital signature of a document is a piece of information based on both the document and the signer's private key. It is typically created through the use of a hash function and a private signing function (algorithms that create encrypted characters containing specific information about a document and its private keys).

Time Stamping:

Time stamping is a technique that can certify that a certain electronic document or communication existed or was delivered at a certain time. Time stamping uses an encryption model called a blind signature scheme. Blind signature schemes allow the sender to get a message receipted by another party without revealing any information about the message to the other party. Possible applications include patent applications copyright archives and contracts. Time stamping is a critical application that will help make the transition to electronic legal documents possible.

Electronic Money:

The definition of electronic money (also called electronic cash or digital cash) is a term that is still evolving. It includes transactions carried out electronically with a net transfer of funds from one party to another, which may be either debit or credit and can be either anonymous or identified. Encryption is used in electronic money schemes to protect conventional transaction data like account numbers and transaction amounts, digital signature or a credit card authorization and public –key encryption can provide confidentiality.

Encryption/ Decryption in E-mail:

Email encryption is a method of securing the content of emails from anyone outside of the email conversation looking to obtain a participant's information. In its encrypted form, and email is no longer readable by a human. Only with your private email key can your emails be unlocked and decrypted back into the original message.

There are various types of email encryption, but some of the most common encryption protocols are:

Open PGP-a type PGP encryption that utilizes a decentralized, distributed trust model and integrates well with modern web email clients.

S/MIME – a type of encryption that is built into most apple devices and utilizes a centralized authority to pick the encryption algorithm and key size.

Encryption in WhatsApp:

WhatsApp uses the 'signal' protocol for encryption, which uses a combination of asymmetric and symmetric key cryptographic

algorithms. The symmetric key algorithms ensure confidentiality and integrity whereas the asymmetric key algorithms help in achieving the other security goals namely authentication and non-repudiation.

Conclusion

Cryptography and network security are the key technologies to ensure the security of the information system. As we advance towards a society where automated information resources are increased, cryptography will continue to rise in importance as a security mechanism. In this project, we have aimed to identify some of the mathematical concepts from elementary number theory behind classical and public key cryptosystems. In the case of RSA, despite years of attempts, no one has been known to crack the algorithm. Such a resistance to attack makes RSA secure in practice. Hence RSA is a strong encryption algorithm that has stood a partial test of time. Undoubtedly, such more sophisticated algorithm than RSA will continue to be developed as mathematicians discover in more in the fields of number theory and cryptanalysis.

REFERENCES

1. Neal Koblitz. A course in Number Theory and Cryptography. 2nd edition, Springer.

2. David M. Burton. Elementary Number Theory. 7th edition, McGraw-Hill.

3. C.E. Shannon, “Communication Theory of Secrecy Systems”, appeared in the report “A Mathematical Theory of Cryptography” dated Sept. 1, 1946.

4. World Wide Web