AN EXPOSITION OF THE MATHEMATICS EXPECTED OF CRYPTOGRAPHIC CHIPS

Kenneth K. Nwabueze

Department of Mathematics,
University of Brunei Darussalam,
BE 1410, Gadong, BRUNEI

Daphne Lai Teck Ching,

Department of Computer Science,
University of Brunei Darussalam
BE 1410, Gadong, BRUNEI

Abstract

Most computer implementations of public key cryptography use specially constructed cryptographic chips that can execute algorithms with binary numbers of length 2048 bits or even more. In this paper, an elementary exploration of the extent of mathematics taken into account in the design of such cryptographic chips is discussed in very simple terms.

Mathematics Subject Classification: 94A05, 94A60

Keywords: Cryptography, Cryptosystems, Algebra and Number Theory

1 Introduction

In recent times, the perpetual increase in information transmitted electronically has led to an increase reliance in cryptography. Cryptography is an algorithm that converts input data into something unrecognizable and vice versa. Cryptography is concerned with four objectives, namely, *confidentiality*; meaning that the information cannot be understood by an unintended recipient; *integrity*, meaning that the information cannot be altered or damaged in transit between sender and intended recipient without detection; *non-repudiation*, meaning that the generator of the information cannot deny at a later stage his or her intentions in the transmission of the information; *authentication*, meaning that both the sender and receiver can easily get back to

each other. Cryptography is used in internet security, phones, televisions, and remote access. It was developed in an effort to help ensure information and communication security. Without cryptography, hackers can get into unauthorized e-mails, bank accounts or acquire free cable service, as well as listen to people's phone conversations.

A cryptosystem or cryptographic system consists of the package of all procedures, protocols, cryptographic algorithms and instructions used for encrypting and decrypting messages by means of cryptography. In general, it contains an integrated assembly of cryptographic primitives, such as encryption algorithms, hash algorithms, etcetera. It also contains protocols for their use, operational procedures, and auxiliary elements, such as documentation, user training materials, and so on. All these taken together make effective security possible. The ability to attack a digital system depends solely on the cryptographic size and protocols involved. Therefore, to ensure absolute security, large numbers are utilized in the manipulation of modulo arithmetic.

The subject is usually classified under the domain of information technology or management, but the major components of the working process hang mainly on the domain of pure mathematics, namely, algebra and number theory. Good cryptography requires strong algorithms, keys of adequate length, and secure key management by means of elementary rudiments of algebra and number theory. The specific objective of this paper is to discuss, in very simple terms, those elements of mathematics utilized in enhancing adequate security.

2 Preliminary Notes

We begin by recalling some important definitions and notions in basic cryptography. By encryption, we mean the process of transforming data or a message into some unintelligible or unrecognizable form. The process of encryption usually involves a set of instructions that enables a step-by-step procedure for transforming the data or message. This is called an encryption algorithm. An encryption algorithm is always associated with a reverse procedure that restores the original message. The reverse process is called decryption. Taken together, the techniques of encryption and decryption are called cryptography.

In general, the protocol of cryptography goes as follows: suppose M is a message to be sent. Let $E(M)$, and $D(M)$ be the encryption, and decryption algorithms or functions, respectively; that is, if $E(M)$ is an encrypted message, then a decryption of this encrypted message would be displayed as $D[E(M)]$. Observe that the latter symbol is just the original message, that is $D[E(M)] = M$. Therefore, if a message is encrypted and decrypted using the symmetric key k , (that is, the message is encrypted and decrypted with same key) and if $E_k(M)$ and $D_k(M)$, are the encryption and decryption functions, respectively,

then $D_k[E_k(M)] = M$. In the same way, if a public-key system (that is, the message is encrypted with a secret key, and decrypted with a public key) is used, and if p_k represents the public key and s_k is the private (secret) key, then $D_{s_k}[E_{p_k}(M)] = M$. Similarly, we have $D_{p_k}[E_{s_k}(M)] = M$.

A simple example of a cryptographic function is the *XOR* function used in binary arithmetic. The *XOR* function is defined with the rules:

$$0 + 0 = 0, \quad 0 + 1 = 1, \quad 1 + 0 = 1, \quad 1 + 1 = 0.$$

Since these define addition rules in the binary system, we have used the notation “+” to denote *XOR* addition above. By the use of the *XOR* addition, we can use any string of zeros and ones as a key to encrypt and decrypt data or messages. For example, suppose we have a message which is a string of four binary digits $M = 1011$, and we are given a four digit key $k = 1010$. We can encrypt the message M , with the key k by addition; that is

$$E = M \text{ “+” } k : 1011 + 1010 = 0001 \text{ (with no carry).}$$

So we have that the encrypted message E is 0001. Note that anyone seeing this value will be unable to interpret the original message M , because it could have been any one of sixteen possible combinations of four zeros or ones. However, if you know the key, it is easy to recover the original message. All that needs to be done to achieve this is to add the key k to the encrypted message E , following the *XOR* rules of addition defined above. So the original message M is recovered thus:

$$M = (M + k) + k = 0001 + 1010 = 1011.$$

In other words, performing an *XOR* addition with the same key twice restores the original message. Note that the security of this system depends on the length of the key k . In this case, there are 16 possible keys of length 4; that is $16 = 2^4$ possibilities. So in general, for a key of length n , there are 2^n possible keys.

Most commercial cryptosystems use keys of length 56, that is, the key space has 2^{56} possibilities. Given current techniques of cryptanalysis (the art of breaking cryptosystems), this key space is no longer large enough for security. With the current state of technology, it is possible to decrypt the message in this key space by employing a *brute force* attack on a message, where by a “brute force” attack we mean the process of trying out every possible key. If we increase the key space to 2^{128} then the system is quite secure; at least with respect to a brute force attack. Note that a key space of 2^{128} is 2^{72} times as large as the common key space of 2^{56} .

There are many well known cryptographic algorithms; two of which are the RSA and the Diffie-Hellman key exchange, which we shall discuss towards the end of this paper. The RSA has been widely studied and it has the advantage that any weaknesses in an algorithm can easily be dictated by practitioners.

3 Essential rudiments

What follows in this section constitutes the rudiments of algebra and number theory required in the implementation of most cryptosystems. Specially designed cryptographic chips must be able to dance around the techniques discussed here using bits length of over 2000.

Modulo Arithmetic

Let p denote a prime number. By $(\text{mod } p)$ we mean *arithmetic done modulo p* ; that is, *divide by p , and keep the remainder r* , where $0 < r < p$. As an example, by $(\text{mod } 5)$, we mean the remainders 0, 1, 2, 3, or 4, obtained by dividing any positive whole number by 5. We shall use the notation $a \equiv b(\text{mod } n)$ (read as a is congruent to b modulo n) for the expression: n divides a , with a left over b . The number n is called the “modulus”. Note that $a \equiv b(\text{mod } n)$ also implies that $a(\text{mod } n) \equiv b(\text{mod } n)$. Therefore, there exist some integers k_1 and k_2 , such that

$$a = k_1 \cdot n + r, \quad b = k_2 \cdot n + r, \quad (0 < r < n).$$

This means $a - b = (k_1 - k_2) \cdot n$; that is, n divides $a - b$, written mathematically as $n \mid (a - b)$.

*The Group Z_p^**

The set $Z_p^* := \{1, 2, \dots, p-2, p-1\}$ is very important in the implementation of public key cryptography because multiplication and exponentiation (taking powers) take place within this set. The following facts are obvious about this set:

- if we multiply any two numbers *modulo p* in the set, the result is a number in the set; meaning that the set is closed under multiplication.
- for any number n in the set, there exists another number n^{-1} in the set such that $n \cdot n^{-1} = 1(\text{mod } p)$; meaning that any number in the set has a multiplicative inverse.

The above two properties imply that the set Z_p^* is a group under multiplication *modulo p* . We also have that Z_p^* is a group under exponentiation, because the k th power of a number is simply the multiplication of that number by itself k times.

For particular examples of the above properties, consider the set Z_7^* . Multiplying 5 and 6 in this set, yields $5 \cdot 6 = 30 \equiv 2(\text{mod } 7)$, and 2 is an element in the set. Also we have $5 \cdot 3 = 15 \equiv 1(\text{mod } 7)$; and this means that 3 is the multiplicative inverse of 5. Similarly, 5 is the multiplicative inverse of 3. It is easy to check that all inverses exists for all the element in the set. Note that it follows that exponentiating a number in the set yields another number in the set. For instance 6 to the fourth power yields $6^4 \equiv 1(\text{mod } 7)$, which

again is an element in the set. Since the set is closed under multiplication and exponentiation *modulo* 7, and every element has an inverse, we have that Z_7^* is a group. Note that because p is a prime, each element has a multiplicative inverse in Z_p^* . Moreover, the only common divisor of p and each of the numbers in the set Z_p^* is 1. In other words, the greatest common divisor (*gcd*) of p and any number in this set is 1.

The following remarks are worthwhile.

- The difference between Z_p and Z_p^* is that in Z_p^* , we omit 0 in the list of elements because it does not have a multiplicative inverse. On the other hand, if we add 0 to the set Z_p^* , we get the set Z_p , which consists precisely of 0 and all remainders *modulo* p .
- The modulo arithmetic considered above for p is not the same for composite numbers. For instance, the set Z_{15}^* is not a multiplicative group. For example, it is an easy exercise to see that if we perform multiplication in *modulo* 15, then 6 has no inverse in the set. In addition, the set is by no means closed under multiplications, for instance 6 and 5 are in the set, but $6 \cdot 5 \pmod{15} \equiv 0 \pmod{15}$ is not in the set.

An element a in Z_p^* is said to be a generator *modulo* p if the set consists of elements made up of powers of a , that is, the set

$$\{a^1 \pmod{p}, a^2 \pmod{p}, \dots, a^{(p-1)} \pmod{p}\},$$

contains, in some order, all the members of Z_p^* . This means that the set Z_p^* represents a rearrangement of $\{a, a^2, \dots, a^{(p-1)}\}$ after all calculations are done *modulo* p . For example, 3 is a generator of Z_7^* , since $3^1 \equiv 3 \pmod{7}, 3^2 \equiv 2 \pmod{7}, 3^3 \equiv 6 \pmod{7}, 3^4 \equiv 4 \pmod{7}, 3^5 \equiv 5 \pmod{7}, 3^6 \equiv 1 \pmod{7}$; that is $\{3, 3^2, 3^3, 3^4, 3^5, 3^6\} = \{1, 2, 3, 4, 5, 6\}$ after calculations *modulo* 7. In other words, the powers of the generator 3 give rise to a permutation of Z_7^* . In general, a generator-tuple *modulo* p is a set of different k generators. This means, $\{a_1, \dots, a_k\}$ is a generator-tuple if each a_i is a generator *modulo* p , where a_i is not equal to a_j , if i is not equal to j . For example, the set $\{3, 5\}$ is a generator-tuple of Z_7^* , since 3 and 5 are generators of Z_7^* .

Note that sometimes an element may generate only a subgroup of a group. For example, the number 2 is not a generator of Z_7^* , because the powers of 2 only yield 1, 2, or 4, (*modulo* 7); that is, $\{2, 2^2, 2^3\} = \{1, 2, 4\}$. Therefore 2 is said to *generate the subgroup* $G(3)$ *modulo* 7, where the symbol $G(3)$ denote a group consisting of three elements. Note also that the number 4 is a generator of $G(3)$, since $\{4, 4^2, 4^3\} \pmod{7} \equiv \{1, 2, 4\} \pmod{7}$.

*Order of an element modulo Z_p^**

A group generated by an element a is said to have order q *modulo* p if q is the lowest power such that $a^q \equiv 1 \pmod{p}$. The two generators of Z_7^* , namely 3

and 5, considered previously, are of order 6 *modulo* 7, because 6 is the smallest power of 3 or 5 that gives us 1 *modulo* 7. That is, $1 \equiv 3^6 \equiv 5^6 \pmod{7}$, and no smaller power satisfies this property. Similarly, the generators of $G(3)$ *modulo* 7, namely 2 and 4, have order 3 *modulo* 7, since $2^3 \equiv 4^3 \equiv 1 \pmod{7}$, and no lower power of 2 or 4 is congruent to 1. Generally speaking, for a prime q , with $1 < q < p$, we define $G(q)$ as the group (or subgroup) of prime order q *modulo* p , if for some generator a ($1 < a < p$), we have that the set $\{a, a^2, a^3, \dots, a^q\}$ is a subset of Z_p^* . This means that the powers of a yield each of the elements in the subgroup. Recall that, by definition, q is the lowest power of a that gives 1 *modulo* p ; therefore, $a^q \equiv 1 \pmod{p}$. So powers larger than q simply start over and run through the same set of numbers. If $a^q \equiv 1 \pmod{p}$, then $a^{(q+1)} \equiv a \pmod{p}$, $a^{(q+2)} \equiv a^2 \pmod{p}$, etcetera. Finally, note that if a is an element of the group Z_p^* , then a is a generator of Z_p^* if a is an element of order $p-1$. This means, if $a^{p-1} \equiv 1 \pmod{p}$, and no lower power gives 1 *modulo* p , because it necessarily follows that the powers of a *mod* p , namely, $a^1, a^2, \dots, a^{p-1}$, runs through all the numbers $1, 2, \dots, p-1$.

4 Essential results

Fermat's Little Theorem

Fermat's theorem tells us that *for any prime p , and number n not divisible by p , we have $n^{p-1} \equiv 1 \pmod{p}$.*

Recall that p does not divide the integers $1, 2, 3, \dots, p-1$, and therefore, by Fermat's theorem, any of these integers raised to the power $p-1$ equals 1 *modulo* p . So for an element n of Z_p^* , we have $n^{p-1} \equiv 1 \pmod{p}$. For example, in Z_7^* , we have $p-1 = 6$, and

$$1^6 \equiv 2^6 \equiv 3^6 \equiv 4^6 \equiv 5^6 \equiv 6^6 \equiv 1 \pmod{7}.$$

It is worthwhile stating that we are not saying here that any number $n < p$ has order $p-1$ *modulo* p , because the order of n may be smaller than $p-1$. For example, 2 has the order 3 *modulo* 7, since $2^3 \equiv 1 \pmod{7}$. Of course, it is also true that $2^{7-1} = 2^6 \equiv 1 \pmod{7}$, as asserted by Fermat's theorem.

As an easy consequence of Fermat's theorem, we have that *the order q of any element of a multiplicative group modulo p must divide $p-1$* . This result also follows from the Lagrange's theorem. For instance, if we take the case where $p = 7$, then $p-1 = 6$, and so the order of any element divides 6. For example, we had previously obtained that 3 and 5 have order 6 in Z_7^* , and 6 divides $(7-1)$. Also, we had that 2 and 4 have order 3 *modulo* 7, and 3 divides $(7-1)$. All these can be seen to be true because if an element a is of order q *modulo* p , then $a^q \equiv 1 \pmod{p}$. However, it also follows by Fermat's theorem

that $a^{p-1} \equiv 1 \pmod{p}$. Therefore if q does not divide $p-1$ then there exist some number n , such that $p-1 = n \cdot q + r$, for $0 < r < q$. Thus we would have that

$$1 \equiv a^{p-1} \equiv a^{n \cdot q + r} \equiv (a^q)^n \cdot a^r \equiv 1 \cdot a^r \pmod{p}.$$

This implies $a^r = 1$; which in turn implies a has order r modulo p which is less than q , and this is contradiction.

Euler's function and theorem

Euler's function for a positive integer n , denoted by $\Phi(n)$, is the set of all numbers less than n that are relatively prime to n . That is, the number of positive integers k ($0 < k < n$), with $\gcd(k, n) = 1$. For a prime p we have that all positive numbers less than p are relatively prime to p , and so $\Phi(p) = p-1$. For example, for $p = 7$, we have $\Phi(7) = 6$. We can show that for a given number n and any number k relatively prime to n , we have that $k^{\Phi(n)} \equiv 1 \pmod{n}$. This is the statement of the Euler's theorem. Note that this theorem applies to any composite numbers n , as well as prime numbers. The Euler's theorem is usually utilized in RSA cryptosystem. RSA cryptosystem uses large numbers which are composite. It is easy to see that the Euler's theorem is a generalization of the Fermat's theorem, since by Fermat's theorem $k^{\Phi(p)} = k^{p-1} \equiv 1 \pmod{p}$.

We can say outright what the number of generators modulo p are, using a result which states that for a prime p , the number of generators modulo p is $\Phi(p-1)$. For example, the number of generators modulo 7 is $\Phi(6) = 2$; indeed $\Phi(6)$ are 1 and 5. So there are a total of two generators modulo 7. Recall that the two generators are 3 and 5. The significance of the numbers 1 and 5 above is that if we have a generator a , then both a^1 and a^5 will be generators. Therefore 3 and $3^5 \equiv 5 \pmod{7}$ are generators. Alternatively, we can also consider 5 as a generator, and so both 5 and $5^5 \equiv 3 \pmod{7}$ are generators.

Now take the group $G(q)$ of prime order q modulo p , where both p and q are distinct primes. Since the order of any element modulo p must divide $p-1$, it follows that q must divide $p-1$. The number of generators of $G(q)$ are precisely $\Phi(m)$, for which m divides $p-1$. This means that there are $q-1$ generators of the subgroup $G(q)$ of prime order q modulo p . This fact guarantees that for large q we have a lot of generators to choose from. For instance, since 3 divides $7-1 = 6$, there are $\Phi(3) = 2$ generators of order 3 modulo 7. This means, there are two generators of the subgroup $G(3) := \{1, 2, 4\}$. Similarly, since 2 divides $7-1 = 6$, there are $\Phi(2) = 1$ generator of order 2 modulo 7. It is also easily checked that 6 is a generator of order 2 modulo 7, giving the subgroup $G(2) = \{1, 6\}$.

5 The RSA Cryptography

A very well known cryptography is the RSA algorithm invented in 1978 by Ron Rivest, Adi Shamir, and Leonard Adleman. We shall go through the protocol of this system in order to see that all the mathematics needed for its implementation have been taken care of in sections 3 and 4. The protocol of this algorithm is as follows:

Protocol

Step 1: Find two large, (say of about 1024 bit) prime numbers p and q . Set $n = pq$.

Step 2: Choose a number e which is greater than 1, but less than n and such that $\gcd((p-1)(q-1), e) = 1$. Note that e does not have to be prime, but it must be odd, and $(p-1)(q-1)$ can not be a prime since it is an even number.

Step 3: Find a number d such that $(de - 1)$ is divisible by $(p-1)(q-1)$. In the expression of section 3, this means finding a number d such that $de \equiv 1 \pmod{(p-1)(q-1)}$, in other words, d is the multiplicative inverse of e .

Step 4: The encryption function is $E(P) = C \equiv P^e \pmod{n}$, where C is the ciphertext (a positive integer), and P is the plaintext (a positive integer). The message P being encrypted must be less than the modulus n .

Step 5: The decryption function is $D(C) = P \equiv C^d \pmod{n}$, where C is the ciphertext (a positive integer), and P is the plaintext (a positive integer). The public key is the pair (n, e) (to be made public) and your private key is the number d (to be kept secret). The product n is called the encryption modulus, e is called the encryption exponent, and d is called the decryption exponent.

End of the game: The public key can be freely published because there are no known easy methods of calculating d , p , or q given only the public key (n, e) . If p and q are each 1024 bits long, then it will take millions of years for the most powerful computers presently in existence to factor n into p and q .

An illustration

We give a concrete example of the RSA algorithm. For convenience sake we shall choose p and q to be small. Note that in practice p and q chosen in this illustration would be large enough that the factorization of $n = pq$ is not feasible. So our choice of p and q here is unrealistically very small.

Now choose the first prime $p = 61$ and the second prime $q = 53$. Our encrypting modulus is $n = pq = 3233$ and since $\Phi(n) = 52 \cdot 60 = 3120$, we can choose our encrypting exponent to be $e = 17$, since $\gcd(17, 3120) = 1$. The decryption exponent is the unique integer d satisfying $17d \equiv 1 \pmod{3120}$. Solving this gives $d = 2753$. Therefore, the public key is $(e, n) = (17, 3233)$

and your private key is $d = 2753$. For example, to encrypt the plaintext value 123, we have the following:

$$\begin{aligned} E(123) &= 123^{17} \pmod{3233} \\ &= 337587917446653715596592958817679803 \pmod{3233} \\ &= 855, \end{aligned}$$

and to decrypt the ciphertext value 855, we proceed as follows:

$$D(855) = 855^{2753} \pmod{3233} = 123.$$

6 Diffie-Hellman Key Exchange

Another popular example of cryptosystems is the Diffie-Hellman key exchange, (also called exponential key agreement) proposed by Diffie and Hellman in 1976. The following is the protocol involved.

Protocol

Step 1: Fix a large prime number p and an integer n of order $p-1 \pmod{p}$; that is $n^{(p-1)} \equiv 1 \pmod{p}$.

Step 2: Randomly choose two integers k and l .

Step 3: Generate two numbers using k and l as follows: $n^k \pmod{p}$ and $n^l \pmod{p}$.

Step 4: With the knowledge of the random numbers chosen in step 2, and the two numbers generated in step 3, compute a secret key

$$s \equiv n^{kl} \equiv (n^l)^k \pmod{p} \equiv (n^k)^l \pmod{p}.$$

End of the game: Two people can send a message to each other by first agreeing on the values in step 1, namely, p and n . Each person chooses a random number, that is, one person chooses, say k and the other chooses, say l ; and generates $n^k \pmod{p}$ and $n^l \pmod{p}$, respectively. One party can encrypt a message and the other can decrypt the message with s . Note that each party does not know each others chosen number k and l in steps 2.

Finally, note that with the knowledge of $n^k \pmod{p}$ and $n^l \pmod{p}$, one cannot deduce k , l or $n^{kl} \pmod{p}$.

An Illustration

As a simple illustration of the Diffie Hellman algorithm, we again, use very unrealistically small values of p , k and l for demonstration purposes.

Fix $p = 7$, and find an element n in Z_7^* of order $7 - 1 = 6$; in fact we have that 3 is a candidate. So $n = 3$. Now let $k = 13$ and $l = 5$. Step 3 yields

$$n^k \pmod{p} \equiv 3^{13} \pmod{7} \equiv 3 \pmod{7} \text{ and } n^l \pmod{p} \equiv 3^5 \pmod{7} \equiv 5 \pmod{7}.$$

The secret keys of any two communicating parties are:

$$s_1 = (n^l)^k(\text{mod } p) \equiv (3^5)^{13}(\text{mod } 7) \equiv 5(\text{mod } 7),$$

and

$$s_2 = (n^k)^l(\text{mod } p) \equiv (3^{13})^5(\text{mod } 7) \equiv 5(\text{mod } 7).$$

References

- [1] A. Menezes, P. van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.
- [2] K. K. Nwabueze, *Basic number theory: a first course*, Educational Technology Centre, Universiti Brunei Darussallam, 2003.
- [3] K. K. Nwabueze, Function language in digital logics, *Int. J. Pure Appl. Math.*, **1** (2005), 131-138.
- [4] S. Y. Yan, *Number Theory for Computing*, Springer, 2002.

Received: April 21, 2005