

DATA ENCRYPTION: MIXING UP THE MESSAGE IN THE NAME OF SECURITY



by Ed DeHart

Introduction

For thousands of years mankind has desired and sought out secret and secure means of communication. This was oftentimes necessitated by war or other political emergencies. Today, military and national security concerns are still of utmost importance. Internet-based commerce, however, has brought communications security to the forefront in a broader sense. Modern digital-based encryption methods form the basis for the security that we often take for granted when making purchases or tending to banking tasks online. Although we currently enjoy very secure systems, there are many incidents of intrusion and malicious hacking that make us think twice when sending sensitive personal data across the Internet. Crimes such as identity theft are common and can cause significant suffering to the victim. The future may indeed hold better and more secure methods, maybe even the ultimate security: a truly unbreakable method of encryption.

When information is transmitted, there always exists the possibility of interception by a party outside of the intended sender-receiver domain. Although current networking technology provides means of intrusion detection, it can never be known for sure if information has been obtained and distributed to unintended receivers. Encryption, a strategy to lessen this risk, seeks to make any intercepted or inappropriately obtained information unintelligible to the intercepting party. The two classes of encryption are codes and ciphers. In this article we will examine how encryption works to accomplish the greater goal of information and communications security. In an overview fashion, the classifications and basic mechanics of encryption will be examined. Brief historical references will be made to demonstrate how encryption has played a role in communications through the years. Finally, a discussion of the possibilities that the future may hold with regards to data encryption will be presented.

As with any pursuit of advancement in technology, there are trade-offs that must be made, and data encryption is no exception. One of the major tasks is to ensure that the algorithms or keys for encryption and decryption remain unspoiled and secret to outsiders, yet known to the intended users. Another task is to devise a scheme that meets encryption needs and is practical. The system must not be burdensome with regards to resources to the point of being inefficient. We will look at some basic methods of encryption that have been employed for millennia, as well as modern means to meet the current needs for advanced data security.

Means of Accomplishing the Task

Encryption methods are built upon the ideas of replacement and scrambling, or mixing up a message so as to be unintelligible to any intercepting party. This is by no means done in a haphazard manner. The sender and receiver utilize methods and keys for encryption that are known to both, but unknown to outside parties. The more complex the encryption process, the more difficult it is to break the scheme. However, added complexity usually involves more lengthy encryption and decryption processes.

Terms describing encryption are often misused and misunderstood. The following list contains some of the most common terms with definitions [4, 10].

cipher a manner of encryption carried out at the symbol level (i.e., letter or number) by transposition or substitution

ciphertext data or message encrypted with a cipher

cleartext unencrypted data or message

code a manner of encryption carried out on a group or block of letters, numbers, or symbols

codeword a word, group, or block of symbols encrypted by a coding scheme

cryptography the study or science of data encryption

encipher to encrypt a message with a cipher

encode process of encryption with the use of codes

encode a message that has been encrypted by both coding and cipher methods for additional security

encryption the process of making data secret by enciphering or encoding or both

missive an unencrypted message

plaintext unencrypted data or message

The code and cipher are the two methods of encrypting data. Encryption must be thought of as an umbrella term that covers both. Likewise, decryption encompasses the processes of deciphering and decoding. Furthermore, there are methods that involve both codes and ciphers, and the terms encryption and decryption may apply to these methods as well [10]. A term that frequently emerges in formal writings of the history of encryption is missive. This simply refers to an unencrypted message, and is synonymous with "plaintext," or

“cleartext” [10]. The terms will be used interchangeably throughout this article.

The encryption of data or a message is accomplished by one, or both, of the methods of encoding or enciphering. Each involves distinct methodologies and the two are differentiated by the level at which they are carried out. Encoding is performed at the word or block level and deals with the manipulation of groups of characters. Enciphering works at the character level. This includes scrambling individual characters in a message, referred to as transposition, and substitution, or replacing characters with others. Let us examine the basics of these methods, and consider examples of the underlying concepts.

Codes

Codes generally are designed to replace entire words or blocks of data in a message with other words or blocks of data. Languages can be considered codes, since words and phrases represent ideas, objects, and actions. There are codes that substitute entire phrases or groups of numbers or symbols with others. A single system may employ both levels of encoding. For example, consider a code encryption scheme as follows: *the = jam, man = barn, is = fly, dangerous = rest*. Then the message, *the man is dangerous*, would read in encrypted form, *jam barn fly rest*. Although overly-simplistic, this example illustrates the basis of codes.

Some of the earliest code forms date back to the 1300s in areas surrounding the Mediterranean Sea. Codebooks were required to accompany the encoded messages, as memorization of a long list of plain words and code equivalents would be too difficult. Many times in these early days, codebooks would be recorded on large folded parchment papers. Senders and receivers would update each other on an as-needed basis, many times via courier. The inherent danger in these arrangements is obvious [10].

With the passage of time, codes began to be employed in the military and political arenas. Since the possession of a code book is necessary in such systems, it became apparent that the possibility of lost or stolen code books carried a high risk. For example, any captured soldier on a battlefield carrying the codebook would pose a disastrous scenario. All secret communications could then be exposed. Any considerable amount of time that passed before discovery of the capture would worsen the situation. The coding methods offered no surefire breach-detection mechanism. However, methods of combating the potential risks were employed, including the use of common words in reference books and novels for codewords, as well as the breaking-up of codebooks. That is, in a military setting, one set of code books could be employed for field and battle operations, while another set could be used for administrative groups. In such a distributed system, the compromise of one group would not necessarily endanger the other, and thus the potential damage could be isolated and/or reduced [10].

With the advent of electrical-based communications, codes became more sophisticated in answer to the needs of the systems. For example, the inventions of Morse code and the telegraph dictated a need for secure transmission that was more sophisticated. In the 1800s, the “additive” was brought into code-based communication. This was simply a secret number that was added to the enciphered code to add another layer of protection [10].

Before digital computing technology, codes relied on list-arrangement schemes to facilitate the use of codebooks. Without the bene-

fit of relational database and indexing technologies, the order and arrangement of the codebook entries were of the utmost importance. Generally, historical codes can be classified as one-part or two-part codes. One-part codes are sometimes referred to as alphabetical or numerical codes, and two-part codes as randomized codes. One-part codes utilize the same codebook for encoding and decoding. A separate codebook is used for encoding and decoding with the two-part code system. The one-part codebook lists plainwords alphabetically with their codeword translations. Some implementations use a root-stem word structure in the creation of the codeword. In this way, the plainwords and codewords may remain in alphabetical order, as they have been linked semantically. This order facilitates the use of one codebook, but allows for easier breaking by analysis since frequency patterns in the usage of codewords will emerge [10].

The two-part code is randomized. It consists of codewords that do not follow any order with respect to their plainword equivalents. It was common to use a block of digits for codewords. For practicality in large code dictionaries, two-part systems require lists for encoding and decoding. Although more difficult to construct, the two-part system is much less susceptible to breaking through analysis [10].

Codes are very susceptible to breaking and possess a large exposure surface with regard to interception and decryption via analysis. Also, there are no easily-implemented means by which to detect breaches in the system. Codes can be made stronger with a layered approach: techniques of both coding and enciphering utilized within a single system. The cipher-based methods of transposition and/or substitution may be employed in concert with coding to realize a more robust system. An example of such a system will be presented in the next section.

Ciphers

The other method of encryption is the cipher. Instead of replacing words or blocks of numbers or symbols with others, as does the code, the cipher replaces individual or smaller sets of letters, numbers, or characters with others, based on a certain algorithm and key. Ciphers are traced back to Arabic, Greek, and Roman civilizations. The use of ciphers diminished after the fall of the Roman Empire, but regained popularity in the Middle Ages by officers of the Catholic church who desired a strong method of secret communications [10]. Presumably, this stemmed from a perceived need to maintain secrecy with regards to church policy and to provide papal privacy.

Like codes, the use of ciphers grew with time, especially during the 19th century. This era saw the development of more sophisticated communication systems. The fact that ciphers worked at a lower level, the letter or symbol, made them very adaptable to the electrical-based communication systems that were being developed and implemented. Where codes were vulnerable to breaking by interception of the codebook, ciphers proved to be a more secure means of sending encrypted messages by relying on keys that could travel with the message. A key, in this case, is a number that is algorithmically applied to plaintext or ciphertext to produce the other.

Cipher-based encryption methods can be broken down into two major classes based on the method of manipulation: substitution and transposition. In substitution, one symbol is simply replaced by another. An example would be to replace letters of the alphabet with another letter. Probably some of the most well-known examples of pure substitution ciphers are the “cryptoquote” games, published

daily in many newspapers across the country. In these puzzles, the player begins by analyzing visually-based linguistic clues, and then, by deduction, begins guessing which letter replaces the ciphertext letter in a trial-and-error manner. Examples of clues include the commonness of three-letter words such as *the* and *and*. Also, words with an apostrophe offer strong clues: the letter following the apostrophe could be an *s*, *t*, or *m*.

Substitution ciphers can be broken down into two subclasses: monoalphabetic and polyalphabetic. The above-mentioned cryptquote puzzle is an example of a pure, monoalphabetic scheme. Here, each occurrence of the letter in the plaintext form is always replaced by the same letter in the ciphertext, in a one-to-one relationship [4]. There are certain inherent weaknesses in monoalphabetic schemes. One of these is that recurring letters make deciphering the encryption easier. For example, the word *LADDER*, with the two adjacent *D*s in the middle, would yield ciphertext with two like, adjacent letters in the middle. This reduces the mathematical complexity of deciphering the message [4]. Substitution ciphers require some sort of mapping key in their implementation. Such mapping keys may be rendered in different forms that will be examined shortly.

Monoalphabetic ciphers are sometimes rendered in a “shift” fashion. That is, the message is encrypted by shifting backward or forward within the alphabet for the cipher-match by a given quantitative difference in alphabetical position. For example, let us say that the shift key is +3. Then in this system, $A = D$, $B = E$, etc. To decipher, you simply move ahead (in a positive shift) the number of the shift along the alphabet. One of the more well-known renditions of this system from history is the Caesar-Shift [10]. Obvious weaknesses here include the ease of breaking the code, the need to have knowledge of the key at both ends, and transportation of the shift key. Another way to render a monoalphabetic cipher is to utilize a random set of character/symbol matches. Again, both sender and receiver need tables of matches, and transporting the map could present security problems for such a system. However, a random match cipher is stronger than a shift-cipher since a simple shift cipher can be broken with relative ease by repetitive effort until an intelligible result emerges. These types of ciphers can be thought of as code encryption at the character level.

A more sophisticated and robust means of cipher-based encryption is found in the polyalphabetic scheme. Here, letters or symbols are replaced by different symbols in the ciphertext by using more than one replacement alphabet. For example, if the name *ABRAHAM* were to be encrypted with a polyalphabetic system, the result could be something like *BFGHURF*. Notice that each occurrence of the letter *A* is replaced by a different letter. Some of the earliest known polyalphabetic schemes utilized hand-held, rotating disks and printed tables, including the famous Vigenère’s table, developed by the 16th century French cryptographer Blaise de Vigenère [10]. Figure 1 shows one of the more popular versions of Alberti’s disk, a rotating cipher disk from history [10].

The cipher disk consists of two disks with a common hub. There is an inner disk that rotates and a fixed outer disk. Characters on the two disks will match when aligned. Both sending and receiving parties utilize the same disk, and the setting of the initial alignment is agreed upon before communicating. This starting base is considered the key. A rotating cipher disk may be employed in either a mono- or polyalphabetic encryption scheme. In a monoalphabetic implementation, the disk simply maps one letter to another. A polyalphabetic scheme

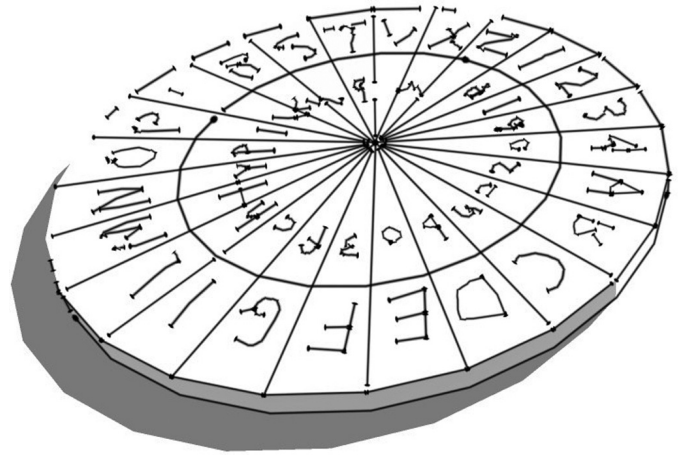


Figure 1: Example of a cipher disk, Alberti’s disk.

may also be employed with the cipher disk. Instead of using the same alignment throughout the entire message, the polyalphabetic method calls for shifts in disk alignment at predetermined intervals. In this way, more than one alphabet is used in the process. There are as many mapped matches to a character as there are shifts. These ciphers are much harder to break than are the monoalphabetic versions.

The other major class of cipher is the transposition cipher. In this method, symbols are not replaced by others, but instead are altered in position within a block or word of plaintext. In a simple transposition cipher, the plaintext *FIGHT* could be rendered in ciphertext as *TIFHG*. Transposition ciphers utilize a mapping key that serves as a pointer to the cipher symbol. The first step is to set the message into blocks for mapping, then perform the encryption or decryption process on them [4]. Let us look at a simple mapping scheme:

A B C D
maps to
C D B A

The encryption algorithm, utilizing this map, or key, would transpose the plaintext *BACD* as *DCBA* and in turn, the decryption system would map in reverse to expose the plaintext.

As mentioned in the discussion of codes, there are encryption methods that utilize both code and cipher methods to create a layered encryption that is stronger and more difficult to break. Let us consider a simple example of a code-and-cipher method using the missive, “move to the west.” First, we encode the plaintext according to codebook entries.

plainword(s)	codeword(s)
<i>move</i>	<i>gads</i>
<i>to the south</i>	<i>aeilthun</i>
<i>to the west</i>	<i>aeilbret</i>

The encrypted message is *gads aeilbret*. Notice also that this code accomplishes some compression in the process by reducing the number of characters to be transported. Many times a compression facility is built into coding schemes. Now this encoded message can be

enciphered. For the sake of simplicity, let us use the following monoalphabetic cipher key with a shift of +3:

$a = d, b = e, c = f, d = g, e = h, f = i, g = j, h = k, i = l, j = m, k = n,$
 $l = o, m = p, n = q, o = r, p = s, q = t, r = u, s = v, t = w, u = x, v = y,$
 $w = z, x = a, y = b, z = c$

Assuming that spaces are unencrypted, the resulting encrypted message is *jdgv dhloeuhw*. On the receiving side, the inverse of the encryption algorithm is carried out by first deciphering with the key. Then, the result is decoded to produce the original plaintext.

Encryption of Digital Data for Security

Modern ciphers are built by component ciphers, including the rotation cipher, the XOR cipher, expansion permutation, as well as others [4]. In computing, encryption is carried out at the bit level, and the digital techniques used to manipulate bits form the basis of current secure communications systems.

The goals in digital encryption are no different than those of historical encryption schemes. The difference is found in the methods, not the objectives. Secrecy of the message and keys are of paramount importance in any system, whether they are on parchment paper or in an electronic or optical format. Modern communication systems utilize software protocols to carry out encryption within a hierarchical protocol suite, such as TCP/IP. The process remains practically transparent to the sender and receiver.

Thus far, for illustrative purposes, we have discussed manipulation at the character, letter, or symbol-level. Computing technology takes encryption to a much lower, and thus, more powerful level. Digital electronic and optical systems employ cipher-based encryption methods at a basal level. These systems lend themselves very well to enciphering schemes, as bit manipulation affords an excellent facility of positional alteration, as well as the important component of parity checking.

Digital data and information, including video, audio, and text, can be separated into groups, or blocks, of bits, and then manipulated for encryption by such methods as XOR (exclusive OR), encoding-decoding, and rotation [4]. As an example, let us examine the basics of the XOR method. Here, a group of bits (e.g., a byte) of the data is compared to a digital key, and the exclusive-or operation is performed on the two to produce an encrypted result. Figure 2 illustrates the process. When the exclusive-or operation is performed on the plaintext and key, the ciphertext emerges and is sent. The receiver performs the exclusive-or operation on the ciphertext and the same key, and the original plaintext is reproduced [4].

Simple manipulation techniques such as these illustrate the potential power of digital encryption. Over the decades, discoveries in number theory and computer science have broadened the scope of implementations of ciphers in computing. The basis of computer data security is the re-

lationship between ciphers, keys, transposition, and substitution [10]. Concealment of the key is of primary importance in data encryption.

Modern digital communications systems employ encryption methods based on numeric key systems. In digital systems the encrypting and decrypting algorithms are generally referred to as ciphers. The key is a number that is used in an algorithmic transformation of the plaintext or ciphertext. As seen in the exclusive-or example, both encryption and decryption use the key on the working matter (plaintext or ciphertext). Current key cryptography is broken down into two categories based on usage scope. These are symmetric-key and asymmetric-key cryptography. In symmetric-key the sender and receiver utilize the same algorithm and key for encryption and decryption. In asymmetric-key encryption the concepts of public and private keys come into play. This arrangement provides a method for key security. The designated receiver holds a private key, while the public key is available to all. The two keys are quantitatively distinct [4].

There are several examples of modern encryption systems based on symmetric-key methods. Let us begin by discussing the Data Encryption Standard (DES). This is a symmetric-key system that has formed the basis for certain methodologies, including Kerberos 4.0 [1]. Though the DES process commences with a 64-bit input key, the effective key length is 56 bits, since for every byte of input data, the least significant bit is used for parity check. The value of the parity bit is determined so as to make an odd number of 1's in each byte of information [7]. DES consists of 16 rounds of encryption. A round of encryption is a process performed upon the data, such as XOR or permutation. DES begins by splitting the 64 bits of data into two 32-bit sections. Then, the 16 rounds of encryption processes are performed on the data before it is eventually reassembled into an output of 64 bits. At the core of the DES process is the "DES function." This is a process that uses a 48-bit key on each 32-bit section, and carries out various encryption/manipulation operations on the data [4].

In response to criticism that DES's key of 64 bits was too short and therefore not secure enough, Triple DES (3DES) was developed. Triple DES seeks to allay the short-key concern of DES by increasing the key length. The process "stacks" three DES processes on top of each other in an encrypt-decrypt-encrypt fashion on the encryption side, and in a decrypt-encrypt-decrypt fashion on the decryption side. Because the process of 3DES was shown to be time-consuming and burdensome with regards to resources, the Advanced Encryption Standard (AES) was developed. This standard is rendered in three basic versions that vary in key-length (128-, 192-, or 256-bit), as well as the number of encryption rounds. AES is one of the more popular methods employed today. A given key in this process is basically an altered version of the prior key in the encryption stack [4].

Asymmetric-key cryptography uses a set of two distinct keys in the encryption and decryption processes. Encompassing number

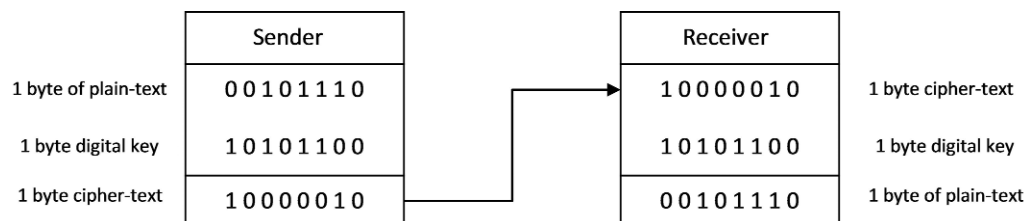


Figure 2: The XOR process for encryption.

theory and modular arithmetic, much of the mathematical concept of public and private key encryption is based on "Fermat's little theorem." Fermat's little theorem states that if p is a prime number and not a factor of a , then when a is raised to the power of $p-1$ and that result divided by p , there will be a remainder of 1

[9]. In equation form, the theorem may be rendered as:

$$a^{p-1} = 1 \pmod{p}$$

The theorem has been mathematically extended to form the basis for public-private key cryptography, and is exemplified in the RSA public key algorithm. RSA is named for Ron Rivest, Adi Shamir, and Leonard Adleman, who invented the method in 1978 [4]. In this method two very large prime numbers, p and q , are established, and their product is referred to as the quantity n . A number e is chosen that is greater than 1 and less than n ($1 < e < n$). Furthermore, e and $(p-1)(q-1)$ must have no common prime factors, and as such, will be relatively prime. The number e does not have to be prime, but will be odd. Once these quantities have been determined, the basis for the process is established. The steps of the algorithm in list form are as follows [3, 4]:

1. Choose two large prime numbers p and q .
2. Multiply p by q to obtain n .
3. Multiply $p-1$ by $q-1$ to obtain ϕ .
4. Select a random integer e such that e and ϕ are relatively prime.
5. Calculate d such that $d \cdot e = 1 \pmod{\phi}$.

The receiver makes the numbers e and n public while keeping ϕ and d private [4].

The equations for encryption and decryption are [4]:

$$C = P^e \pmod{n}$$

The ciphertext, C , is modulus- n of the value representing the plaintext, P , raised to the power of e .

$$P = C^d \pmod{n}$$

The plaintext, P , is modulus- n of the value representing the plaintext, C , raised to the power of d .

Other methods exist in digital encryption that utilize the public and private key concept. One of the more well-known of these is the Diffie-Hellman system. This method was proposed by Whitfield Diffie and Martin Hellman in a 1976 paper they authored titled, "New Directions in Cryptography." The process facilitates secure key exchange by using public keys to exchange private keys [2, 4]. These mathematical processes are at the heart of security mechanisms in current communications systems. I will discuss the concepts of some of these mechanisms shortly.

A definite parallel can be drawn when considering the encryption of modern digital keys and the historic method of concealed codebooks, cipher tables, and keys, and the efforts made to keep them out of the wrong hands. Concealment, control, and containment of keys (and codebooks in the olden days) were, and still are, major concerns of any system of data encryption. The public and private key

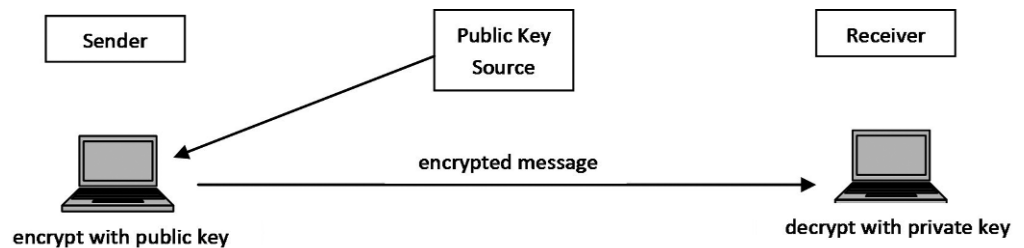


Figure 3: Asymmetric-key encryption.

systems such as RSA and Diffie-Hellman, form the basis for current communications security.

The public-private key system provides a solution to key distribution security pitfalls that exist in symmetric-key systems. The fact that two distinct keys, held by separate parties, are used for the encryption and decryption processes adds more built-in security to the arrangement [1]. Let us now examine the asymmetric-key process. As shown in Figure 3, the sender obtains the receiver's public key, available to all. This key may be obtained from any accessible source including the Internet. After encrypting the message with this key, the data or message is sent. The intended receiver is capable of decrypting the message with a second key that is not accessible in a public manner.

Mathematical "safety" is provided in a system like RSA by forcing any would-be cipher-breaker to factor a very large number. The larger the number, the more unlikely it is that the factoring can be accomplished in any reasonable amount of time [3].

Although in this arrangement, an eavesdropper can intercept the encrypted message, they will not be able to read it without the private key. However, there does exist the possibility that an eavesdropper can "digitally pose" as the original sender, by encrypting it with the public key and sending the information to the receiver. While asymmetric-key encryption can provide secrecy of communication, when used alone it cannot guarantee the identity of the sender. This is where methods of authentication, the process that seeks to validate the identity of the communicating parties, come into play.

Authentication may be established by methods such as digital signatures and certificates. Digital signatures are concerned with ensuring that a message has originated from a known and trusted source. This method is basically the asymmetric-key process in reverse. With a digital signature, the sender encrypts the data to be sent with his or her private key and sends the message. The receiver decrypts the message using the public key of the sender. In this way, since only the sender holds the private, secure key, it can be presumed with a certain level of confidence that the source of the message is indeed valid [1].

Digital certificates seek to reduce some of the potential hazards of digital signatures. These could include for example, decrypted signatures

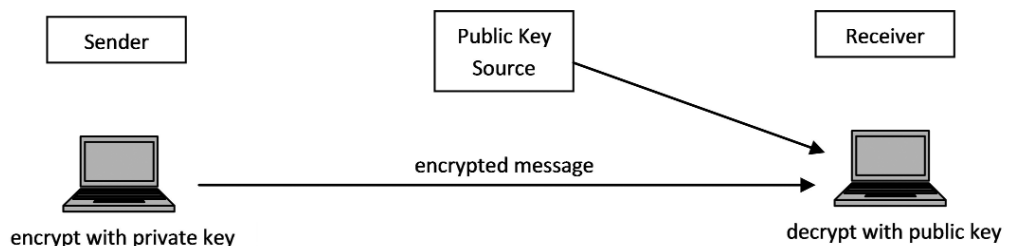


Figure 4: The digital signature, a reverse asymmetric-key encryption.

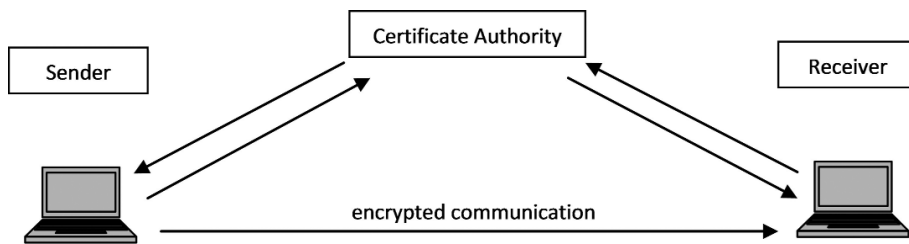


Figure 5: The digital certificate system.

or passwords. Either of these could result in a breach of the system. The digital certificate system provides a secure method of acquiring a public key. The public key given from a digital certificate source (a third-party certificate server) is basically an encrypted version of the key. The issuers of digital certificates are often referred to as certificate authorities. Figure 5 illustrates the process [1].

Here the sender receives a certificate, an encrypted public key, from the certificate source. The receiver gets a copy of the sender's public key and requests a copy of the certificate. The receiver obtains the encryption key from the certificate source that was used for the sender's certificate. The receiver then decrypts the certificate with the key and thereby obtains the sender's public key. The receiver also checks the digital signature to verify authenticity [1].

The concepts of asymmetric-key methods provide for services such as secure sockets layer (SSL). SSL is a protocol that ensures authentication, message integrity, and confidentiality for Internet-based transactions in e-commerce [1]. Web browsers support the protocol, and when the web address begins with https (http-secure) as opposed to http, SSL is invoked. A closed padlock icon may appear on the browser. SSL works at the transport layer and supports application layer protocols above it such as HTTP and FTP [1,4]. The web browser encrypts the data, using either 40 or 128-bit encryption [6]. SSL technology employs a certificate method to ensure authenticity of the communicating parties. For the utmost security, a trusted vendor should be the source of the certificate. While SSL certificates may be generated by any online commercial entity, it is safer to utilize those provided by validated third-party vendors, or certificate authorities [6].

Information Security in the Future: Truly Unbreakable Systems?

Currently implemented computing and communication systems utilize binary methods and manipulations that are based upon prime numbers, modular arithmetic, and the factorization of very large numbers. Computation in this paradigm is carried out via discrete, two-state techniques. While this has served and continues to serve us well in current systems, there are more powerful and promising models on the horizon. One of the more well-known of these is termed quantum computing and its derivative technology, quantum cryptography.

Quantum cryptography is based on Heisenberg's theory of uncertainty, which is derived from the principles of quantum physics. Quantum physics is concerned with elementary particles on an extremely small scale, and how the behavior of such particles tends to "buck" the classical principles of physics. Matter, for example, can be broken down

to the atomic level, and further into subatomic particles. Light, the basis for quantum computing and cryptography, may be broken down to a basal element called the photon. Photons, and other extremely small-scale particles, exhibit some interesting characteristics. Among these are the uncertainties of state [8].

Quantum cryptography could be thought of as an object-oriented method of encryption at a basal and physical level. An object, such as a photon of light, possesses a set of attributes such as polarization or intensity, and these may be altered and manipulated. Quantum cryptography methods seek to capitalize upon those attributes that have a certain property: they cannot be observed without being changed. This represents the basis of quantum technology. Optical quantum cryptography seeks to use the photon's attribute of polarization to generate a random encryption key. Varying polarization states can create a stream of data that can be digitized. An attribute such as this is of great value in encryption because any eavesdropper will never see the data as it originally existed. Furthermore, intrusion detection is intrinsic to the system [8].

Let us consider a simple example of a generic photon-based scheme, as illustrated in Figure 6. A sender, utilizing a highly-precise laser device, emits photons of varying and random polarizations to a receiving device. The photon stream sent represents the key. Once the key is agreed upon by the receiver and sender, there is system-assured secrecy of the key. Assume that the polarization attribute of each photon of light is known in the sender-receiver domain. The polarization of a photon, once "viewed" or measured, is irreversibly changed according to quantum behavior. So, since the receiver can compare the states of the photons on transmission and reception, those that have been altered can be rejected for inclusion in the key on a photon-by-photon basis.

The encryption key, once established as secret, can be used for encrypting data for secure communications with certainty that the messages are indeed secret and impenetrable [8].

A quantum cryptography system based on photons would require full optical lines for connection between the sending and receiving laser device nodes. For longer-haul distances, optical repeaters would need to be introduced into the system, and this presents a hurdle.

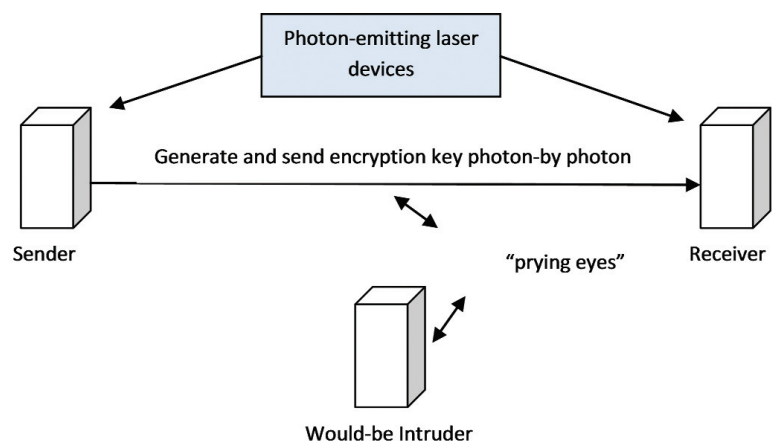


Figure 6: A quantum cryptography key system.

Any intermediary device would have to “read” the photon, and therefore alter it. Recent research and development is currently underway to solve this issue [5]. Possible solutions include passive designs that operate parallel to and outside of the communications system.

Although quantum cryptography is still in the research stage, the future appears promising for the technology. It is anticipated that quantum cryptography will improve in the coming years with advances bringing its use into the mainstream of general communication systems, maybe replacing current technologies in incremental steps. No doubt, quantum-based encryption could offer a super-encryption technique that would inspire trust in business as well as consumer communication applications.

Encryption has come a long way: from hand-written secret codebooks on parchment paper, to handheld cipher disks, Morse code blocking, and now the emergence of a technology based on some of the smallest particles known in the form of quantum cryptography. As secure and secret communications are necessary, the pursuit of better and more robust methods will continue, as well as attempts to break any new technologies. The quest goes on, and it is expected that in the coming decades, many new and exciting encryption schemes and models will be born.

References

1. Casad, J. 2004. *Sams Teach Yourself TCP/IP* 3rd Ed. Sams Publishing.
2. Chapple, M. 2003. Diffie-Hellman key exchange. http://searchnetworking.techtarget.com/tip/1,289483,sid_gci879100,00.html.
3. Davis, T. 2003. RSA encryption. <http://www.geometer.org/mathcircles/RSA.pdf>.
4. Forouzan, B. 2006. *Data Communication and Networking* 4th Ed. McGraw-Hill, New York, NY.
5. Hesseldahl, A. 2006. A quantum leap in data encryption. *Business Week*. http://www.businessweek.com/technology/content/nov2006/tc20061106_302053.htm?campaign_id=bier_tcv.g3a.rssd1109z.
6. Kerner, S. M. 2005. SSL: Your key to e-commerce security. http://www.ecommerce-guide.com/solutions/secure_pay/article.php/3510761.
7. Tropical Software. DES encryption. <http://www.tropsoft.com/strongenc/des.htm>.
8. Vittorio, S. Privacy through uncertainty. <http://www.csa.com/discoveryguides/crypt/overview.php>.
9. Weisstein, E. W. Fermat's little theorem. <http://mathworld.wolfram.com/FermatsLittleTheorem.html>.
10. Wrixon, F. B. 2005. *Codes, Ciphers, Secrets and Cryptic Communication*. Black Dog & Leventhal Publishers, Inc.

Biography

Ed DeHart (edd0617@ecu.edu) has a Bachelor of Science degree in mathematics from Barton College, and is a last-year graduate student at East Carolina University pursuing a Master of Science in technology systems with a concentration in digital communications. His interests include music, digital media, and foreign languages.

Make a difference...

CHANGING THE WORLD

At Sandia National Laboratories, our primary mission is to secure a peaceful and free world through technology. You'll never be bored working at Sandia. There are so many exciting challenges and stimulating directions your career can take. Be a Sandian. Join the team that is changing the world.


We have opportunities for college graduates at the Bachelor's, Master's, and Ph.D. levels in:

<ul style="list-style-type: none"> ■ Electrical Engineering ■ Mechanical Engineering ■ Nuclear Engineering ■ Computing Engineering ■ Material Sciences 	<ul style="list-style-type: none"> ■ Computer Science ■ Chemistry ■ Information Technologies/ Information Systems ■ Biological Sciences ■ Business Administration ... and more
---	---

We also offer internship, co-op, and post-doctoral programs.

www.sandia.gov

Sandia is an equal opportunity employer.
We maintain a drug-free workplace.


Sandia National Laboratories
 operated by
LOCKHEED MARTIN

