Technology Briefing
Cryptography: A Security Tool of the Information Age
Sergai Boukhonine

Introduction

This paper gives a broad overview of cryptography: its definition, its historical development, the current state of the art, its significance for Information Systems Managers, and future tendencies and development. The paper lists local researchers and practitioners of cryptography and provides useful references to a variety of sources for cryptographic information.

What is cryptography?

We first define the essential terms: cryptography, cryptanalysis, cryptology, and encryption.

The Merriam-Webster Collegiate Dictionary provides the following definitions:

Table 1. Definitions of a few essential cryptography terms

Term	Definition
Cryptography	1. secret writing 2. the enciphering and deciphering of messages in secret code or cipher 3. cryptanalysis
Cryptanalysis	1. the solving of cryptograms or cryptographic systems 2. the theory of solving cryptograms or cryptographic systems : the art of devising methods for this
Cryptology	the scientific study of cryptography and cryptanalysis
Encryption	1. encipherment 2. encoding

In essence, cryptography is the art of both secret writing (encryption). reading secret writing (decryption) and penetrating the secret writing of others (cryptanalysis). Cryptographers devise new methods and algorithms of secret writing; cryptanalysts attempt to break those algorithms. Encryption is done through use of either encipherment or encoding (the difference between them will be explained later on).

Why is cryptography important for my organization?

It is easy to see why secret writing is important to governments or the military; outcomes of battles and wars often depend upon keeping your communications secret while penetrating those of your opponent. It may be less obvious why is it important to business organizations. Here are a few reasons:

- Importance of intellectual property versus “brick and mortar” assets
- Threat of industrial espionage by competitors and even foreign governments
- Need for secure access to bank accounts and electronic transfers of funds
- Requirement for secure E-commerce
- Desire to avoid legal liability.

In the past, a firm could keep its information relatively secure without turning to cryptography. The situation changed with the arrival of computers and the Internet. Listed below are differences between past and present business practices that have elevated the importance of cryptography in the commercial sector.

Past

- Business communications were typically done using regular mail, landline telephones, telegraph, etc. -- all relatively secure media
- Strong laws protected privacy of telephone conversations and regular mail.
- Company secrets were usually kept on paper and could be locked away in a secure safe.
- Ubiquitous paper trails existed that could be reconstructed and used to detect fraud.

Present

- Email is pervasive. It is cheap and fast. It is also extremely accessible and insecure. Many experts liken email messages to postal cards readable by anybody rather than sealed envelopes. Email is archived and kept by intermediaries. Email use will continue to grow in no small part because “snail mail” is slow and getting more expensive. The anthrax scare also accelerated movement away from the U.S. postal system.
- Company secrets are kept in computer memory and can be potentially accessed by either outside hackers or, more commonly, by malicious employees.
- E-commerce is becoming increasingly important for many organizations and it is heavily dependent on security.
- Companies increasingly adopt wireless technologies that are inherently insecure (one can protect a telephone cable from wiretapping but wireless communications can be intercepted by anybody).
- There is a gradual movement to paperless society (e.g. some airlines charge extra for paper tickets when electronic tickets are available) that renders paper trails obsolete but also places additional data security requirements.
- Most legal experts agree that there are weak legal protections of email and computer file privacy, yet:
- There is a growing threat of legal liability for lax computer security (it will not help improve your reputation either).

An interesting example of how technology can jeopardize security as well as potentially lead to legal problems is *warchalking*. Warchalkers walk or drive about, looking for wireless computer networks and make chalk marks on sidewalks or building walls to indicate accessible locations. People who have computers with wireless capabilities can use these networks to check email, surf the Web, etc. Warchalking, which originated in London, is gaining devout followers in the U.S. It may sound innocent enough, but a malicious cracker can use a chalk mark to break into your system and steal your secrets and the secrets of your customers and business partners. He may use your network and broadband connection to launch a DNS (denial of service) attack, a virus or a worm, etc. Similar risks occur when using public networks such as those in hotels, airports, or Internet Cafes.

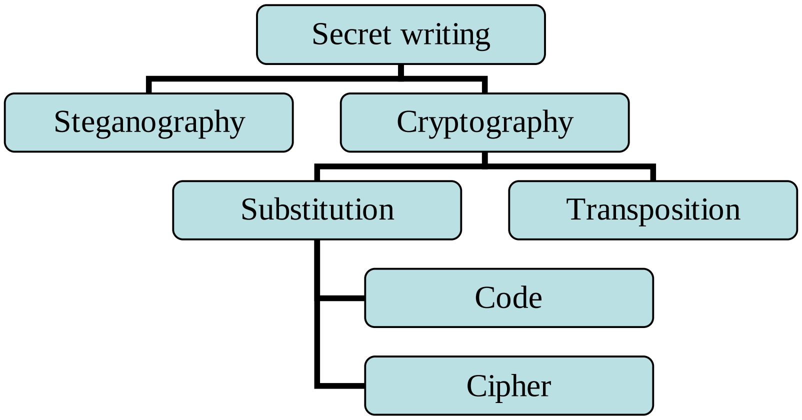
There are legal precedents for holding organizations liable when malicious crackers use their computers to launch a DNS attack. The bottom line is: if your customers and business partners, or even third parties, suffer because of insufficient computer security at your organization, they may sue you with potential financial or reputation loss.

The inevitable conclusion is that an organization must assume its email messages can, and probably will, be intercepted, its corporate networks hacked into, its secret computer files accessed by malicious insiders and /or outsiders. Thus, sensitive email and files must be encrypted. Encryption is also a central part of other security technologies such as authentication, digital signatures, etc.

What is the history of secret writing?

There are different ways to protect messages.

Figure 1. Main branches of secret writing (adopted from Singh (1999)).



Steganography

The easiest way to keep a message secret is to hide the very fact of its existence. This technique is called steganography, which in Greek literally means covered writing. Singh (1999) describes how ancient Chinese wrote messages on fine silk, then rolled the silk into a small ball and covered the messages with wax. The wax balls would then be swallowed. According to an excellent book on steganography and watermarks by Katzenbeisser and Petitcolas (2000),

> *…the most famous examples of steganography go back to antiquity. In his Histories…Herodotus (c. 486-425 B.C.) tells how around 440 B.C. Histiæus shaved the head of his most trusted slave and tattooed it with a message which disappeared after the hair had re-grown. The purpose was to instigate a revolt against the Persians. Astonishingly, the method was still used by some German spies at the beginning of the 20th century... Herodotus also tells how Demeratus, a Greek at the Persian court, warned Sparta of an imminent invasion by Xerxes, King of Persia: he removed the wax from a writing tablet, wrote his message on the wood underneath and then covered the message with wax. The tablet looked exactly like a blank one (it almost fooled the recipient as well as the customs men).*

A very popular form of steganography is writing letters in invisible ink. As children, many people experimented by writing letters with such “secret inks” as lemon juice, milk, or even urine. When paper is heated, the hidden writing is revealed.

Katzenbeisser and Petitcolas (2000) also tell about

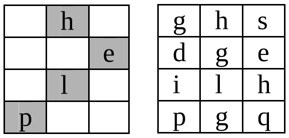
> *…hiding secret messages in spaces no larger than a full stop or small dot of ink…by 1860 the basic problems of making tiny images had been solved by Dragon, a French photographer: during*

> *the Franco-Prussian War of 1870–1871, while Paris was besieged, messages on microfilm were sent out by pigeon post... During the Russo-Japanese war of 1905, microscopic images were hidden in ears, nostrils, and under fingernails... by World War messages to and from spies were reduced to microdots by several stages of photographic reduction and then stuck on top of printed periods or commas in innocuous cover material such as magazines.*

An acrostic is another popular form of steganography. The idea is to hide words in a text. "How easily life passed" is an acrostic for "help." Words can also be hidden in music by substituting letters by notes. Katzenbeisser and Petitcolas (2000) note that this technique can be further improved:

> *… when the message is hidden at random locations in the cover-text. This idea is the core of many current steganographic systems. In a security protocol developed in ancient China, the sender and the receiver had copies of a paper mask with a number of holes cut at random locations. The sender would place his mask over a sheet of paper, write the secret message into the holes, remove the mask, and then compose a cover message incorporating the code ideograms. The receiver could read the secret message at once by placing his mask over the resulting letter. In the early 16th century Cardan (1501–1576), an Italian mathematician, reinvented this method which is now known as the Cardan grille.*

Figure 2. A simple Cardan grille.



	h	
		e
	l	
p		

g	h	s
d	g	e
i	l	h
p	g	q

After the invention of the phonograph and then the telephone, messages would be concealed by shifting voice recordings or telephone conversations to a frequency band inaudible to a human ear. Contemporary examples of traditional steganography include hiding information in the least significant bits of image or audio files. However, the "security-by-obscurity" (Katzenbeisser and Petitcolas) principle of traditional steganography proved to be weak and unable to provide proper information protection. Little by little, steganography fell into oblivion. However, it attracted renewed interest in the 1990s principally because of the proliferation of digital image, audio, and video files; increased capacity and bandwidth of the Internet; the advent of digital watermarking – a technique used to prevent unauthorized copying and distribution of digital content.

Increased capacity and bandwidth of the Internet created conditions for massive exchange of digital image, audio, and video files, and the posting of such files on Internet sites. Many Internet sites, such as auction sites (Ebay, etc.) allow anonymous postings. Digital content files as a rule are large and hence provide considerable hiding space. Together these conditions created fertile ground for explosive growth of steganography on the Internet. This growth was largely fueled by the Internet underground. The "serious" cryptographic community, whether in government, academia, or industry, was essentially caught unawares. There are literally hundreds of steganography (or "stego") programs available for download on the Internet; most of them are free (a site listed at the end of this paper boasts of listing 80+ programs for platforms ranging from Windows to Mac to Linux). Unfortunately, not all steganography users use it for benign or innocuous purposes. According to the *New York Times* (see reference at the end), terrorist associated with Osama bin Laden and Al Qaeda were trained to exchange messages by posting pictures on the Internet. Many images on the Internet appear to have steganography in them. These hidden messages can be hard to detect. There are computer programs that can detect steganography by looking for statistical deviations by looking for statistical deviations from the expected pattern of data in image, audio, or video files. However, such programs often have difficulty detecting hidden messages in JPEG files as well as detecting very short messages. To make it even harder, hidden messages are often encrypted for extra protection. Use of steganography by criminals and terrorists worried law enforcement, academic, and commercial cryptography communities so much that many researchers stopped publishing any further work on steganography detection methods for fear of tipping off the evildoers.

An interesting application of steganography is anonymity protection. Katzenbeisser and Petitcolas (2000) define it as attempting to "hide the meta-content of messages, that is, the sender and the recipients of a message." One way to do it is by using an anonymous remailer. It quite obvious that email anonymity can and is probably used by criminals to evade law enforcement. On the other hand, one should not forget that steganography and anonymous remailers are used by citizens of nations with dictatorial and oppressive political regimes to promote ideals of freedom, democracy, and human dignity.

Another purpose of using steganography is to protect digital copyright by using the so-called digital watermarks. According to Katzenbeisser and Petitcolas (2000):

> *…watermarks do not always need to be hidden, as some systems use visible digital watermarks…, but most of the literature has focused on imperceptible (invisible, transparent, or inaudible, depending on the context) digital watermarks which have wider applications. Visible digital watermarks are strongly linked to the original paper watermarks which appeared at the end of the 13th century... Modern visible watermarks may be visual patterns (e.g., a company logo or copyright sign) overlaid on digital images and are widely used by many photographers who do not trust invisible watermarking technique…*

Digital watermarks must be robust to prevent tampering or removal. Once implemented, search robots can use digital watermarks to monitor copyrighted material on the Internet. Collected information can be used for preventing unauthorized use or collecting royalties. There are numerous providers of digital watermarking technologies such as Digimarc (Web site listed at the end). Digital watermarking technology will no doubt gain more importance as the entertainment industry intensifies attempts to protect digital copyrights and lawmakers pass bills regulating them.

Cryptography[1]

While steganography tries to hide the very existence of a message, cryptography attempts to hide its meaning. Cryptography itself can be divided into transposition and substitution. *Transposition* is a rearrangement of letters of a message according to a certain algorithm. For example, HELP is encrypted as PLEH by writing it backwards or EHPL by swapping each pair of letters. Transposition is simple but provides little security.

Substitution is a more robust and versatile form of cryptography. As the name suggests, characters in the original text (known as plaintext in crypto speak), are substituted by other characters or symbols using certain algorithm. Substitution of letters is known as cipher whereas substitution of words and phrases is known as code (although people sometimes use these terms interchangeably). Substitution techniques have ancient roots. Singh (1999) writes that cryptography by substitution was described in Kama-sutra, a famous Indian book written in the fourth century AD, as a useful skill for women who wanted to keep their liaisons secret. A recommended technique was to pair alphabet letters at random (A and K, L and X, etc.).

Julius Caesar wrote about using ciphers for military purposes in *Gallic Wars*. Caesar used a shift cipher of 3 places.

Table 2. An example of Caesar cipher.

Plaintext	information technology changes everything
Ciphertext	lqirupdwlrq whfkqrorjb fkdqjhv hyhubwklqj

To decipher the ciphertext above, a cryptanalyst needs to know its *key*. In this case, the key is 3. Hence this type of key is very easy to implement. It is easy to see how weak this type of cipher is, since there are only

[1] Most of the information on the history of cryptography was derived from Singh (1999), whose book provides a fascinating account of the development of cryptography throughout the ages. Singh, Simon "The code book: the evolution of secrecy from Mary Queen of Scots to quantum cryptography," Doubleday, New York, 1999

25 possible keys (shifts) for the English alphabet. It is possible to increase security of a shift cipher by using two or more shift numbers, e.g. 3 and 7. The character would be shifted by 3, second by 7, third by 3, etc.

If any random rearrangement of letters is possible, it is possible to generate a huge number of distinct ciphers – over 400,000,000,000,000,000,000,000,000 (Singh (1999)). It is nearly impossible to break a cipher like that by *brute force*. However, both sender and receiver must have a copy of the key and keep it from falling into enemy's hands. To keep a key simple to remember, a keyword may be used. The first letters of the plain alphabet are substituted for the keyword. The remainder is just the remaining letters of the alphabet. A longer keyword or phrase enhances security.

Table 3. A substitution cipher with an ISRC keyword.

Plain alphabet	abcdefghijklmnopqrstuvwxyz
Cipher alphabet	isrcdefghjklmnopqtuvwxyzab
Plaintext	information
Ciphertext	hneotmivhon

Simple monoalphabetic substitution ciphers were unbroken until Arab scholars invented *frequency analysis* around the 9th century AD. The central idea is that some letters are used more often than others (e.g. e in English). Frequency analysis is also applicable to two letter groups, three letter groups, etc. In addition to simple frequency analysis, there is a multitude of other cryptanalysis techniques. E.g. certain letters may usually precede or follow some letters (e.g. q is usually followed by u) but seldom or never other letters. Frequency analysis also applies to words (e.g. *THE* is the most common three letter word in English). Generally speaking, language has patterns and cryptanalysts learned to exploit them.

In Europe, methods to break monoalphabetic cipher were not discovered until 16th century. Beside ciphers, European cryptographers widely used codes, which seemed more secure since words are more difficult to analyze by frequency analysis methods than letters. However, codes have an inherent weakness: it is difficult to come up with codes for many words. Codebooks are bulky and may easily fall into enemy's hands. A famous example of a code phrase was radioed by the Commander of the Imperial Japanese Fleet Admiral Yamamoto. It said "*Climb Mount Niitaka*"; it was the signal to attack Pearl Harbor.

In 1580, French diplomat Blaise de Vigenère published his *Traictè de Chiffres* which described a polyalphabet cipher know as the indecipherable cipher. His method uses 26 Caesar shift alphabets and a keyword, which determines from which shift alphabet a substitute letter, is taken. This method is invulnerable to frequency analysis and has a great number of keys.

The "indecipherable cipher" remained impervious to attacks until the XIX century. In 1854 it was cracked by the famous inventor Chares Babbage, who noticed its periodic character. However, Babbage did not publish his discovery (possibly due to its use by the British during the Crimean War). In 1863 it was independently discovered by Prussian Friedrich Wilhelm Kasiski, who got the credit for breaking the "indecipherable cipher."

In 1883, a Flemish man living in France Auguste Kerckhoff enunciated the first principles of cryptography, in which he asserts that cryptographers must assume that the method used to encipher data is known to the enemy, so security must lie only in the choice of key. Before the beginning of WWI, the French military created a formidable cryptographic establishment. During the WWI, the French routinely intercepted and deciphered German military traffic. The British also had major successes. The biggest success was the British interception and decipherment in 1917 of a secret German telegram to their embassy in Mexico, in which the German Foreign Ministry instructed the ambassador to ask Mexico to attack the U.S. together with Germany. Mexico was promised return of Texas and California. The decrypted telegram was given to the Americans who promptly overcame their previous reluctance and entered the war against Germany.

In 1918, the German inventor Arthur Scherbius founded a company dedicated to creating a mechanical encryption machine that came to be known as Enigma. It was the first truly successful mechanical implementation of the polyalphabet cipher. The German military were keen to avoid previous mistakes and gave Enigma the green light. All in all, the German military bought more than 30,000 Enigma machines. Fortunately for the Allies, the French obtained Enigma blueprints in 1931. They shared the information with their Polish allies. A young mathematician Marian Rejewski at the Polish Biuro Szyfrow broke the Enigma machine (although subsequent German improvement restored its secrecy). After the Polish defeat in 1939, all results of Polish research were given to the British. The British cryptanalysts at Bletchley Park headed by Alan Turing finally broke the German ciphers; their breakthrough turning the tide of the war for the Atlantic.

American code breakers were equally successful at breaking Japanese ciphers. The battle of Midway in 1942 was won in no small part due to the American knowledge of Japanese plans. Later, American fighter pilots intercepted and shot down a plane with Admiral Yamamoto on board due to intercepted and decrypted radio communications. Although the Japanese had their own decryption success stories, there was one American cryptographic system that they could never penetrate – the famous Navajo code talkers. Their distinctive language defied all attempts by the best Japanese cryptanalysts.

Cryptography after the WWII

The most important developments in cryptography after WWII are arguably the following: computer and public key cryptography. Both profoundly changed the field. Computer cryptanalysis made hand as well as mechanical encryption obsolete. Encryption became immeasurably more complex. The computer has brought cryptography, which previously was a realm of diplomats, spies and generals, into the mainstream. But even the mighty computer could not have done it without the invention of *public key cryptography*.

What is public key cryptography?

All cryptographic systems discussed so far are known as symmetric or private key. This simply means that the same key is used for both encryption and decryption of a plaintext. To use symmetric key cryptography, both sender and receiver must possess the same key.

While this requirement may not seem that problematic at first sight, in practice it presents a formidable problem of key distribution. Keys must be changed often because use of the same key over and over again creates patterns. Cryptanalysts thrive on patterns and repetitions. While keys are being distributed, they may fall into the wrong hands. While the U.S. government can afford to employ diplomatic couriers, who deliver keys and codebooks to embassies abroad in briefcases chained to their wrists, key distribution becomes a nearly insurmountable barrier to individuals and business organizations.

Problems with key distribution led to a breakthrough discovery of public or asymmetric key cryptography. Public key cryptosystems employ pairs of mathematically related keys: a public key and a private key. It is practically impossible to derive a private key from the public key. Here is how it works: if Alice wants to send Bob an encrypted message, she encrypts it using Bob’s public key, which is available to anybody. The encrypted message, however, can now only be decrypted using Bob’s private key, which is presumably known to Bob alone.

Public key cryptography is possible because of the so called one-way functions. They are easy to compute, but hard to invert. An everyday example of a one-way function is breaking an egg or a vase. A mathematical example is modular arithmetic. The two keys, public and private, are two large integer numbers. One can easily derive a public key from a private key but not vice versa, primarily because of difficulty of factoring large integer numbers (a public key N is a product of two primary numbers p and q which together constitute a private key).

Foundations of the asymmetric key cryptography were described by Diffie, Hellman and Merkle of Stanford University in 1975. In 1978, Rivest, Shamir and Adleman at MIT invented the most widely adopted public key cryptosystem RSA. However, the first discovery of the asymmetric key cryptography was made in the secret laboratories of the British government. In 1969, James Ellis formulated the asymmetric key theory and in 1973 Clifford Cocks wrote a practical one-way function.

What are the currently available algorithms and technologies?

Private (symmetric) key cryptography algorithms[2]

DES (Data Encryption Standard) and its derivatives: *double DES* and *triple DES*. Originally developed by Horst Feistel at IBM's Thomas J. Watson Laboratory and known as Lucifer. In1976, Lucifer was adopted as a standard called DES. At the insistence of the National Security Agency, the key length was limited to 56 bits. DES proved to be a very robust standard and it was quite secure until the end of the 90s when it was it was demonstrated that it was no longer unbreakable. In 1997, the National Institute of Standards and Technology (NIST) announced that DES will be eventually replaced by the Advanced Encryption Standard (AES).

AES (Advanced Encryption Standard) is based on the Rijndael (pronounced "Rhine-doll") algorithm. Its minimum key size is 128 bit and maximum one is 256 bit. According to the NIST website,

> *When considered together, Rijndael's combination of security, performance, efficiency, ease of implementation and flexibility make it an appropriate selection for the AES.*
>
> *Specifically, Rijndael appears to be consistently a very good performer in both hardware and software across a wide range of computing environments regardless of its use in feedback or non-feedback modes. Its key setup time is excellent, and its key agility is good. Rijndael's very low memory requirements make it very well suited for restricted-space environments, in which it also demonstrates excellent performance. Rijndael's operations are among the easiest to defend against power and timing attacks.*
>
> *Additionally, it appears that some defense can be provided against such attacks without significantly impacting Rijndael's performance. Rijndael is designed with some flexibility in terms of block and key sizes, and the algorithm can accommodate alterations in the number of rounds, although these features would require further study and are not being considered at this time. Finally, Rijndael's internal round structure appears to have good potential to benefit from instruction-level parallelism.*

Despite being so new, AES has already been implemented in a software package named Steganos Security Suite 4, which is available for 39.95 Euros (see the Web site listed at the end). Without a doubt, AES will enjoy widespread popularity and will be implemented in numerous software packages and ASIC chips.

DEA (International Data Encryption Standard) was authored by Xuejia Lai and James Massey in 1990. It was strengthened in 1991 following the discovery of differential cryptanalysis. IDEA operates on 64 bit plaintext blocks and has a 128 bit key. It uses both substitution and transposition methods. IDEA is patented by Swiss Ascom Tech AG. IDEA has been sold seen 1993 both as a VLSI chip and software. IDEA is also used by the popular PGP software.

Blowfish was created by Bruce Schneier. Its key-length is variable. In its full form, it has 16 rounds of encryption.

[2] Information of crypto algorithms is derived from Nichols (Nichols, Randall K. "ICSA Guide to Cryptography," McGraw-Hill, New York, 1999) as well as numerous Internet sources

RC5 (Rivest Cipher Number 5) was invented by Dr. Ronald Rivest. It is considered to be a very strong cipher. RSA Laboratories has tested the algorithm and is satisfied with the results. Its implementations are available from RSA Security and other vendors.

Public (asymmetric) key cryptography algorithms

RSA (Rivest, Shamir, and Adleman) was developed in 1977-78. RSA is a premier public key cryptosystem.

DH (Diffie-Hellman Key Agreement Algorithm) was initially developed in 1976. This protocol gives two users, working over an insecure communication channel, an ability to independently calculate the same secret key from publicly communicated numbers.

ECDH (Elliptic Curve Diffie-Hellman Key Agreement Algorithm) is a permutation of DH which uses so called elliptic curve discrete logarithm problem and is considered more efficient.

RPK (Raike Public Key) system was developed by William Raike of New Zealand University. It is based on the discrete logarithm problem.

Hybrid cryptography

PGP (Pretty Good Privacy) was released in 1991 by Phil Zimmermann. PGP is not really an encryption algorithm in itself; rather it is a computer program that uses both public key encryption (RSA) and private key encryption (IDEA) combining them in a way that is both secure and efficient. The reason why Zimmermann chose to combine the two algorithms is the huge difference in speed between them. Private key encryption is much faster than public key cryptosystems. In the 1980s, when Zimmermann started his work, only government and large companies had computers powerful enough to make RSA implementation practical. PGP works by using the RSA algorithm to encrypt a private key that is used by the IDEA algorithm to encrypt the message itself. PGP can also be used for file encryption; in this case only the IDEA algorithm is used. Currently, PGP is sold as a commercial product by Network Associates, Inc. (formerly ViaCrypt and PGP Inc.). It is, however, available as a freeware for individual non-commercial use and can be downloaded from their Web site (listed at the end).

Authentication systems, Public Key Infrastructure (PKI), and Certification Authorities (CA)

Authentication goals include the prevention of:

- masquerade attacks,
- content tampering and modification,
- timing modification (delay or replay) and denial of service attacks,
- message repudiation.

There are three basic techniques for achieving these objectives:

- message encryption (cipher text acts as its own authenticator),
- cryptographic checksum also known as message digest (a value that is computed over a sequence of bytes; it changes for every change in the original message),
- hash function (a known function that maps a message into a fixed-length hash value, which serves as the authenticator),
- digital signature (prevention of message repudiation). When Alice sends a message to Bob, first she creates a message digest. Then she encrypts the message digest with her private key creating the digital signature. If Bob knows Alice's public key, he can go ahead and decrypt Alice's digital signature thus confirming her identity (notice how a public and private key roles are switched here).

Here are examples of authentication/digital signature systems.

SHA-1 (Secure Hash Algorithm version 1) was developed by NIST in 1993. It is based on algorithms developed by Ronald Rivest (MD4 (message digest) and MD5).

DSA (The Digital Signature Algorithm) was proposed by the NIST in 1991 and became a standard in 1993. It is the first digital signature to be accepted by the U.S. government.

ECDSA (Elliptic Curve DSA) is an elliptic curve analogy of DSA.

Kerberos was developed at MIT in the 1980s as a user identification, authentication and authorization system.

OPS (Open Profiling Standard) is backed by a few prominent Internet companies such as VeriSign and Netscape as well as privacy activists such as the Electronic Frontier Foundation.

Unfortunately, digital signature systems do not completely solve the problem of impersonation. Digital signatures work fine when you deal with someone you know and can confirm her public key. However, authentication becomes more difficult when you deal with strangers. Suppose you have received an email from an individual who claims to be the Emperor of Vulgaria. His message is digitally signed and his public key is attached. However, this information does not really help you confirm his identity. One theoretical possibility would be to look up the Emperor of Vulgaria in hypothetical Yellow Pages listing his public key. However, what if the sender happens to be named Jack Smith? There might be many individual that bear this name. At any rate, it is impossible to confirm the identity of a stranger without resorting to a trustworthy third party.

Vendors of Public Key Infrastructure (PKI) and Certification Authorities (CA) in essence play the role of trustworthy third parties. When Alice emails Bob for whom she is a complete stranger, Alice may want to get an identifying certificate from Carol. A certificate is a digitally signed statement by a CA that independently confirms Alice's identity. An example of a certificate is *VeriSign Digital IDs*. *VeriSign Digital IDs* use RSA cryptography with 1024 key length and are used by thousands of Web servers and hundreds of thousands of individuals. The idea of certificates sounds appealing, but closer examination raises many questions (Ellison and Schneier, 2000):

- Who gave the CA the authority to issue certificates, i.e. who made it trusted?
- How secure are CA's computers? Can a malicious hacker add his own public key to CA's directories?
- CA may be an authority on making certificates, but not on their content (an example is Secure Socket Layer (SSL) server certificate which gives name of keyholder (a corporation) and Domain Name System (DNS) name for the server. No CA is an authority on DNS).
- How secure are CA's practices?
- Single Sign-On (SSO) practice when an employee signs in once by plugging in her smart-card is vulnerable to an unauthorized individual gaining physical access to a signed-on computer.

What is JAVA and XML cryptography?

XML and JAVA are very popular technologies for Web enabled and network applications. Both technologies have powerful security and cryptography features.

XML Cryptography

Mactaggart (see XML primer link listed at the end) indicates that

> *…traditional methods of establishing trust between parties aren't appropriate on the public Internet or, indeed, on large LANs or WANs. Trust mechanisms based on asymmetric cryptography can be very useful in such situations, but the ease of deployment and key management, the extent of interoperability, and the security offered are, in reality, far less than*

> *the enthusiastic vendors of different Public Key Infrastructures (PKI) would have us believe. There are particular difficulties in dealing with hierarchical data structures and with subsets of data with varying requirements as to confidentiality, access authority, or integrity.*

It is quite easy to encrypt or digitally sign any document in its entirety. What XML makes possible is selective encryption of parts of a document, different encryption levels for different recipients, signing of only parts of a document possibly by different people, etc. The core element in the XML encryption syntax is the *EncryptedData* element which, with the *EncryptedKey* element, is used to transport encryption keys from the originator to a known recipient, and derives from the *EncryptedType* abstract type. Data to be encrypted can be arbitrary data, an XML document, an XML element, or XML element content; the result of encrypting data is an XML encryption element that contains or references the cipher data. When an element or element content is encrypted, the *EncryptedData* element replaces the element or content in the encrypted version of the XML document. There are still a number of unresolved problems, but researchers around the world are working on making XML cryptography more convenient and robust.

JAVA Cryptography

Despite numerous security holes discovered over the years, JAVA is generally recognized as providing the best development tools from the standpoint of security. The *java.security* package includes classes used for authentication, e.g. message digest and digital signature. A message digest (cryptographic checksum) is a value that is computed over a sequence of bytes. A message digest changes for every change in the original message. A recipient of the message can re-compute its digest and confirm if the message is authentic. If a message digest is encrypted with the sender's private key, it is considered digitally signed. It may be decrypted with the sender's public key thus confirming her identity. While the *java.security* package includes cryptography-based classes, it does not include classes for actual encryption and decryption of data. The latter are included in the *javax.crypto* package which is known as Java Cryptography Extension. Both *java.security* and *javax.crypto* packages are provider-based, so developers can choose among a number of implementations. For actual encryption and decryption, a developer may use a number of different algorithms, including DES and its derivatives, IDEA, etc. An example of a practical implementation of Java cryptography is the Phaos Crypto toolkit which provides a set of core cryptography algorithms in an easy to use Java API, including AES, DES and derivatives, RC2, RC4, Blowfish, RSA, DSA, MD5, and many others (more information available of the company Web site listed at the end).

While JAVA/XML combination is a popular choice for platform-neutral development, other popular development tools such as C/C++ and C# offer their own implementations of cryptographic systems.

What are the projected benefits of and potential problems with cryptographic technologies?

According to Nichols (1999), good encryption provides the following benefits:

- Data confidentiality (secrecy)
- Data integrity (protection against forgery or tampering)
- Authentication of message originator
- Electronic certification and digital signature
- Non-repudiation (neither sender nor receiver can deny a document – essential for contractual arrangements)

A major problem with cryptographic systems is that, if not implemented correctly, they may be vulnerable to attacks while providing a false sense of security. In a sense, bad security is worse than no security at all.

It is important to remember that cryptography is only one element of computer security. There are numerous ways to defeat a cryptosystem without even resorting to cutting edge cryptanalysis (these will be explained later).

Since cryptography is a subject of paramount importance to national security, intelligence gathering and law enforcement agencies around the world often attempt to regulate use of encryption. The U.S. government regulates exports of cryptographic systems by classifying them as munitions and requiring licenses for their export (the author of PGP Phil Zimmermann was prosecuted by the FBI for allegedly exporting his encryption software without a license). This requirement should be borne in mind when installing and using encryption systems in foreign branches of American organizations. The latest changes to export regulations for the first time allow exports of commercial encryption systems with keys exceeding 64 bits (see Web site listed at the end of this paper). Besides U.S. regulations, international and multinational organizations must pay attention to government regulations in all countries where they have a presence.

Generally speaking, government security agencies would like to retain the ability to decipher encrypted messages when necessary. With this in mind, the American Escrowed Encryption Standard was adopted in 1994. It contained two encryption chips known as Clipper and Capstone that would be used for voice and computer communications respectively. These systems failed to be universally adopted because of privacy concerns. However, the tug of war between law enforcement needs on the one hand, and privacy concerns on the other, is bound to continue.

How secure are contemporary encryption technologies?

When considering security of contemporary encryption technologies, one should distinguish between cryptanalysis attacks and attacks NOT using cryptanalysis methods as well as combinations of cryptanalysis/other types of attacks.

Security against Cryptanalytic Attacks

An IT manager may not readily recognize the importance of understanding relative security of different encryption algorithms. However, most contemporary software and hardware systems allow for use of many alternative encryption algorithms. E.g. Windows 2000 uses a Crypto API which is certified for use with DES, triple DES, Secure Hash Algorithm and Digital Signature Algorithm. An IT manager may want to consider the fact that DES is being phased out by the U.S. government and is not considered to be highly secure anymore. There are cryptographic products out there that simply do not offer enough security and IT managers should be aware of this.

With the exception of one-time pad cipher, all today's crypto algorithms are theoretically vulnerable to attacks. A crypto algorithm is considered strong if an attack against it is practically infeasible. That said, the fact that a protocol can withstand both brute force attacks and other attacks using known algorithms does not guarantee that this protocol will remain impervious as new attack algorithms are developed.

According to Dr. Denning of Georgetown University, there are four basic methods of attacking a cryptosystem (Nichols, 1999):

- cipher-text only (only cipher text is available),
- known-plain text (some plain text-cipher text pairs are available due to an easily guessable nature of a cipher text),
- chosen-plain text (both plain and cipher texts are known; e.g. a known record is planted into a database and changes are noted)
- chosen-cipher text (used for attacks against public-key systems; both plain text, cipher text and public key are known – the challenge is to guess the private key)

All of the above methods and their permutations have been used by cryptanalysts to attack contemporary cryptosystems. Let us take a look at how secure some of them are:

DES was consider to be very secure when it was first introduced. It is believed that the NSA built a massively parallel DES breaking computer in the 1980s (one can argue that the NSA's cryptanalysis capabilities are primarily a threat of foreign governments and various evildoers rather than to business organizations). By the 1990s, DES no longer was considered secure for applications requiring very high security.

Table 4. DES strength (adopted from Nichols (1999) based on data of Dan Ryan at SAIG).

Threat	*Budget*	*Time to Break (40 bit key)*	*Time to Break (56 bit key)*
Hacker	Tiny	1 week	Infeasible
Small Business	$10K	12 minutes	556 days
Corporation	$300K	24 seconds	19 days
Big Corporation	$10M	0.7 seconds	13 hours
Government	$300M	0.0002 seconds	12 seconds

In 1998, the Electronic Frontier Foundation announced a DES-cracker project to be built for under $250,000 (Nichols). The resulting machine successfully found a 56-bit key in 56 hours. One can reasonably conclude that DES is even more vulnerable today, although it may still be acceptable for applications with moderate security requirements.

Double DES may appear more secure than DES because the encryption process is repeated twice. Nevertheless, it is vulnerable to a common plain text attack known as the man in the middle (MIM). Thus double DES is seldom used. However, triple DES is considered to be reasonably secure and is widely used.

AES is considered to be secure (thus its selection as the new standard).
The AES specifies three key sizes: 128, 192 and 256 bits. In decimal terms, this means that there are approximately:
3.4 x 10^{38} possible 128-bit keys;
6.2 x 10^{57} possible 192-bit keys; and
1.1 x 10^{77} possible 256-bit keys.
In comparison, DES keys are 56 bits long, which means there are approximately 7.2 x 10^{16} possible DES keys. Thus, there are on the order of 10^{21} times more AES 128-bit keys than DES 56-bit keys. A computer that takes just a second to break DES would require 149 trillion years to break AES.

IDEA is more secure than DES largely because of its 128 bit key. According to experts, it does well when confronted with brute force attacks but may be vulnerable to other types of attacks.

Blowfish can withstand both brute force and differential cryptanalysis attacks. If all 16 rounds of Blowfish encryption are used, it is considered a strong algorithm, although there are some reduced implementations of Blowfish that are considered insecure.

RC5 is considered secure against both brute force and differential cryptanalysis attacks if all 16 rounds of encryption are used. If the number of encryption rounds is reduced, attacks against RC5 may be effective. On July 14, 2002 an international project *distributed.net* cracked a 64 bit RC5 key. It took 1757 days and 331,252 people were involved. Thus the days of 64 bit keys may be numbered; they should not be used for encryption of documents that may remain sensitive for years to come (see their Web site for more information).

RCA cipher is considered secure. The main attack against it involves factoring large integer numbers. So far mathematicians failed to discover a way to accelerate factoring significantly and many believe that it may be impossible in principle. Then again, this problem may just be waiting for the right person to tackle it. One problem with RCA is that in order to achieve high security level, its key must be much larger in length than those of elliptic curve cryptography algorithms such as *ECDH (Elliptic Curve Diffie-Hellman Key Agreement Algorithm).* The ECC (Elliptic Curve Cryptography) is currently considered to be optimal; it is highly secure as well as efficient.

Security against Attacks Not Dependent on Cryptanalysis

As mentioned earlier, it is not necessary to use cryptanalysis to defeat a cryptosystem. In fact, there are many possibilities including:

- Viruses, worms, and other malicious code that would secretly record and transmit a secret key residing in a computer memory;
- TEMPEST attacks that read electromagnetic waves recording everything that happens in a computer system;
- Differential Power Analysis attacks that attack smart cards and cryptographic tokens (require physical proximity);
- Unauthorized physical access to a computer containing a crypto key;
- Unauthorized physical access to a computer which has been issued a certificate by a Certification Authority (allows impersonation).

Attacks listed above and other threats underscore a fact that cryptography cannot provide computer security on its own. It is only a part of comprehensive computer security, albeit an important one.

What is the future of secret writing?

At the moment, cryptographers seem to have an upper hand over cryptanalysts. Current encryption standards appear nearly unbreakable (unless scientists in secret government laboratories know something that the rest of us don't). However, if and when a new technology called *quantum computing* is implemented, all currently used encryption technologies may become instantly obsolete.

While detailed description of quantum computing is beyond our scope (there was an ISRC technology briefing on quantum computing during the 2000-2001 season and there are resources listed at the end of this paper), the central idea is that a particle, e.g. a photon, can simultaneously be in a *superposition of states* i.e. be in more than one state at once. A quantum computer would deal with quantum bits (qubits) that can simultaneously represent both 0 and 1 by simultaneously spinning in different directions. As Singh (1999) indicates, with 250 spinning particles, each representing both 0 and 1, a quantum computer could perform 10^{75} simultaneous computations, completing them all practically instantaneously. Incidentally, 10^{75} is greater than the number of atoms in the universe. In essence, quantum computers can compute faster because they can accept as inputs not one number but many different numbers and subsequently perform a computation on all of these numbers simultaneously. This can be viewed as a massive parallel computation, but instead of having many processors working in parallel we have only one quantum processor performing all computations at once. Some natural applications for a quantum computer would include factoring very big integers (necessary to crack the RCA algorithm), as well as searching lists at an incredibly high speed (necessary to crack the DES algorithm).

Currently, nobody knows exactly when quantum computing will become a reality, but when and if it does, it will signal the end of traditional cryptography. After all, the biggest problem with current cryptosystems is that they are theoretically breakable – it just takes too long. The Gartner Group provides the following forecasts of when quantum computers will be available:

> *By 2006, asymmetric cryptography will be vulnerable to factoring attacks that use quantum computers (0.6 probability).*
> *By 2010 through 2012, intelligence agencies will have affordable quantum computers; by 2015, quantum computers will be affordable to large enterprises (0.7 probability).*
> *Enterprises that use cryptography to protect against well-funded threats will need to develop migration plans to stronger techniques by 2008 (0.6 probability) for implementations beginning in 2010 (0.7 probability).*
> *A 10-qubit, special-purpose quantum computer will be available by 2004 (0.6 probability), and a 10-to-100-qubit, general-purpose computer will be available within the next 10 years (0.6 probability).*

However, cryptanalysts need not rejoice prematurely. There is a potentially unbreakable crypto known as *quantum cryptography*.

Quantum cryptography is based on photon physics. A photon vibrates as it travels through space. The angle of vibration is known as polarization of the photon. Using different polarizations, Alice can transmit a unique symmetric key to Bob each time she needs to send a message. An eavesdropper has no chance to intercept this key because of unique properties of quantum physics (it is impossible to read a photon message without altering its contents). Given the current state of human knowledge, quantum cryptography is *theoretically unbreakable*. It would signal the end in the battle between cryptographers and cryptanalysts, the end of cryptological history. Cryptographers would be absolute victors. A stunning thing about quantum cryptography is that it does not require quantum computing and may become available within a decade or two, long before quantum computing! As Singh notes, in 1995 researchers at the University of Geneva succeeded in implementing quantum cryptography in an optic cable stretching for 23 kilometers. Scientists at the Los Alamos National Laboratory conduct experiments with quantum cryptography in air, with an eye on creating a crypto system that would operate via satellites.

Who (locally) is involved with cryptographic technologies?

Houston has a variety of resources for those interested in cryptography.

Academic resources

Much computer security and cryptography research is done at the Department of Computer Science at Rice University. Assistant professor Dan Wallach is involved in research and teaches a computer systems security course that deals with cryptographic issues (his personal page is listed at the end). The other professor at Rice involved with IT security research is Peter Druschel. There are also a number of students involved in security research such as Adam Stubblefield and Christian Coarfa. Rice researchers offer corporate consulting and lectures in computer security and cryptography.

Cryptography and computer security research is also conducted by scientists at the Department of Computer Science at the University of Houston. Professor Ernst Leiss offers security consulting as well. One of his corporate talks "… gives an overview of three aspects of data security, namely statistical databases, authorization systems and crypto systems. In particular, it reviews cryptography, from classical or symmetric encryption to public-key or asymmetric encryption and presents (fairly informally) the underlying principles of the RSA scheme for public-key cryptography. Then it discusses the problem of authentication, including password approaches, digital signatures, and interrogative protocols." Dr. Leiss's personal page is listed at the end. Associate professor Albert Cheng does research on computer and Internet security (his page is listed at the end). The Department of Computer Science at UH offers courses dealing with computer security and cryptography, such as COSC 7371: Data Security.

Other resources

There are quite a few companies offering training, consulting and services in computer security and cryptography in Houston. Here are a few examples:

- Schlumberger provides solutions that resolve network security issues. These solutions are offered as part of the DeXa Suite of Services. Schlumberger also offers Network Solutions Security Support Kit, SSK 3.0, a unique combination of Schlumberger smart card-based solutions and 3-G International (3GI) authentication technology to help manage enterprise-wide security (see the listed Web page).
- SANS institute offer practical training courses covering topics such as need for cryptography, types of encryption, real-world cryptosystems, VPNs, steganography, PGP, signing data, key management, and key servers (see their Web page).
- RSA Security, one of the premier cryptography and computer security companies, offers training seminars in Houston (see the Web site for details).

- EDS offers computer security training through the Cyber Security Institute as well as a range of computer and data security products (see their Web site).
- The Gartner Group offers a wealth of information on security and cryptography as well as consulting services (see their Web site).

How can my organization become involved with cryptographic technologies?

Chances are your organization already uses some form of cryptography. To learn more, IT employees may undergo cryptography training offered by a number of consulting firms locally and nationally. For smaller organizations, PGP offers an inexpensive yet reliable crypto solution. A larger organization may want to get in touch with a company offering comprehensive scalable solutions such as RSA Security.

For More Information

Online resources

http://members.tripod.com/steganography/stego/software.html	List of 80+ steganography programs
http://www.digimarc.com/	Site of Digimarc, a provider of digital watermarking solutions
http://theory.lcs.mit.edu/~rivest/crypto-security.html	Personal page of Ron Rivest, one of the inventors of RSA and a leading authority on all things cryptographic. Provides a variety of links to cryptographic resources
http://www.youdzone.com/cryptobooks.html	Rating of many books on cryptography
http://www.bxa.doc.gov/Encryption/Default.htm	Commercial encryption export controls
http://www.pgpi.org	The international PGP home page
http://csrc.nist.gov/encryption/aes/aesfact.html	Description of AES on NIST Web site
http://www-106.ibm.com/developerworks/xml/library/s-xmlsec.html/index.html	XML security primer
http://www.nue.et-inf.uni-siegen.de/~geuer-pollmann/xml_security.html	XML security links
http://www.securingjava.com/toc.html	Java security
http://www.qubit.org/	Center for Quantum Computing
http://www.slb.com/Hub/Docs/tt/nws/security/	Schlumberger's security solutions
http://www.rsasecurity.com/training/courses_schedules/locations/misc.html	RSA Security training sites
http://www.eds.com/services_offerings/csi/so_cybersecurity_overview.shtml	EDS' Cyber Security Institute
http://www3.gartner.com/Init	The Gartner group Web site
http://www.warchalking.org	The warchalking Web site
http://rechten.kub.nl/simone/ds-lawsu.htm	Overview of legal issues surrounding use of digital signature
http://www.distributed.net/pressroom/news-20020926.html	RC5 64 bit key cracking information
http://www.steganos.com/en/sss/index.htm	Information on Steganos Security Suite 4
http://www.phaos.com/products/crypto/crypto_datasheet.pdf	Information on Phaos Crypto toolkit

Articles

Kolata, Gina "Veiled Messages of Terror May Lurk in Cyberspace," New York Times, October 30, 2001

Ellison, Carl and Schneier, Bruce "Ten Risks of PKI: What You're not Being Told about Public Key Infrastructure," *Computer Security Journal*, Volume XVI:1, 2000

Froomkin, Michael "The Essential Role of Trusted Third Parties in Electronic Commerce," Oregon Law Review 49, 1996. It is available on the Web at www.law.miami.edu/~froomkin/articles/trusted1.htm

Wheatman, Vic and McRory, Lindsay "Quantum Computers: The End of Public-Key Cryptography?" Gartner Group Research Note, January 4, 2002

Books

Singh, Simon "The code book: the evolution of secrecy from Mary Queen of Scots to quantum cryptography," Doubleday, New York, 1999

Katzenbeisser, Stefan and Petitcolas, Fabien A. P. "Information Hiding Techniques for Steganography and Digital Watermarking," Artech House, Inc., Boston, 2000

Nichols, Randall K. "ICSA Guide to Cryptography," McGraw-Hill, New York, 1999

Schneier, Bruce "Applied Cryptography," John Wiley & Sons, 1996

Local contacts

http://www.cs.rice.edu/~dwallach/	Personal page of Dan Wallach at Rice
http://www.cs.rice.edu/~astubble/wep/wep_attack.pdf	Example of Rice crypto research
http://www.cs.rice.edu/~astubble/	Personal page of Adam Stubblefield at Rice
http://www.acm.org/top/people/leiss.html	Personal page of Ernst Leiss
http://www.cs.uh.edu/Faculty/leiss.html	UH page of Ernst Leiss
http://www.cs.uh.edu/~acheng/acheng.html	UH page of Albert Cheng
http://www.sans.org/Houston/	SANS Institute's offerings in Houston