LACROSSE UNIVERSITY

Cryptography

A PAPER SUBMITTED TO
THE FACULTY OF THE DIVISON OF SCIENCES
IN CANDIDACY FOR THE DEGREE OF
MASTER OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

By
Ayesha Asghar

August 30, 2006

To Khalida

Cryptography is concerned with the conceptualization, definition, and construction of computing systems that address security concerns.

Oded Goldreich

Contents

ILLUSTRATIONS

Figures Page

Tables

PREFACE

The aim of this Applied Knowledge Paper is to give a deep insight on cryptography. A detailed understanding has been given for cryptographic primitives and protocols.

The document starts by giving an introduction to cryptography, followed by its services and protocols.

Further the XML and JAVA cryptography. Followed by the projected problem and potential benefits. Next, cryptography has been reviewed in the real world.

Finally, the document wraps up the paper by evaluating the pros and cons and future of cryptography. The author of the paper hopes that this discussion will serve as a comprehensive guideline for the readers.

ACKNOWLEDGEMENTS

I would like to thank the auspicious instructors of Lacrosse University under whose guidance I’ve been able to come up with this report. I thank them for their cooperation through out and also for resolving all problems and every issue that I faced.

Next, I would like to thank my family who provided me with all the sources and material required for making this possible. Lastly; I would like to thank God Almighty who gave me the strength and will power to complete this piece of work.

ABBREVIATIONS

3DES	Triple Data Encryption Standard
ACL	Access Control Protocol
ADSL	Asymmetric Digital Subscriber Line
DES	Data Encryption Standard
IKE	Internet Key Exchange
IP	Internet Protocol
IPSec	Internet Protocol Security
IPX	Internet work Package Exchange
Kbps	Kilobits Per Second
L2TP	Layer Two Tunneling Protocol
PKEY	Public Key Encryption
PPTP	Point to Point Tunneling Protocol
SLA	Service Level Agreement
SSL	Secure Socket Layer
TCP/IP	Transmission Control Protocol/Internet Protocol
VPDN	Virtual Private Dialup Network
VPN	Virtual Private Network

1. Introduction

1.1. Purpose

The purpose of this paper is to describe a well-formed and a thorough understanding of the Cryptography. This paper also gives an insight into need and primitives of cryptography. Further it throws light on cryptographic services and protocols.

It focuses on the JAVA and XML cryptography. Further the applications of cryptography have been discussed in the real world.

This paper provides an understanding of the benefits and problems of Cryptography. And it concludes by critically evaluating the benefits of cryptography.

This paper will assist the novice programmers/non-programmers to have a better understanding of cryptography.

1.2. Scope

The paper provides answers to what, when, how and who questions related to cryptography; such as:

- What is Cryptography?
- What are cryptographic primitives?
- Cryptographic protocols.
- Projected benefits and potential problems.
- Business benefits of cryptography.
- Future of cryptography.

2. Cryptography

The origin of the word cryptology lies in ancient Greek. The word cryptology is made up of two components: "kryptos", which means hidden and "logos" which means word. Cryptology is as old as writing itself, and has been used for thousands of years to safeguard military and diplomatic communications. For example, the famous Roman emperor Julius Caesar used a cipher to protect the messages to his troops. Within the field of cryptology one can see two separate divisions: cryptography and cryptanalysis. The cryptographer seeks methods to ensure the safety and security of conversations while the cryptanalyst tries to undo the former’s work by breaking their systems.

The main goals of modern cryptography can be seen as: user authentication, data authentication (data integrity and data origin authentication), non-repudiation of origin, and data confidentiality. In the following section these services are elaborated more. Subsequently it’s explained that how these services can be realized using cryptographic primitives.

2.1. Cryptographic Services

2.1.1. User Authentication

If one logs to a computer system there must (or at least should) be some way that one can convince it of their identity. Once it knows one’s identity, it can verify whether one is entitled to enter the system. The same principal applies when one person tries to communicate with another: as a first step one wants to verify that one is communicating with the right person. Therefore there must be some way in which one can prove their identity. This process is called user authentication. There are several ways to obtain user authentication.

One can give him something only one knows: a password, a (pre-designed) user-id, a pin code, and so on. Or one could have some specific items with which one can identify oneself: a magnetic strip card, a smart card (a hand-held computer the size of a credit-card), a token. One might make use of biometric properties; it is a well-known fact that fingerprints, the shape of the hand and retinal pattern of a person are good decision criteria. These however require specialized equipment and thus a big investment. However, these biometric systems are not perfect: some legitimate users will inevitably fail the identification and some intruders will be accepted as genuine. Other techniques include measurements of how a person types their name or writes their signature, or can take into account the location of the user.

Fig. 1. User authentication

For the time being the first two methods are the ones generally applied, and many practical systems use a combination of both. Since the user's memory is limited, this information should not vary too much over time. Whether it is a password, a pin code or a user-id, all these items are being defined at a certain time and often don't change from there on. One might argue that one could change their password, but this is not done each time one accesses the computer. This indicates that someone who can eavesdrop with this information will later be able to impersonate the user. A similar observation holds true for a magnetic strip card or memory chip. All these systems provide static authentication only.

If the user possesses a device which can perform simple computations, the security can be increased significantly by introducing the well-known challenge-response idea. If a person tries to identify to the system, the system generates a random challenge and sends it to the person or to their device. In case of a token (a mini-calculator), the user will have to enter the challenge on the keyboard. The device will then compute the corresponding response, using secret information which has been assigned to one. This response is then sent back to the system, which verifies it (see figure 1). If more sophisticated protocols are used, the verifier does not need secret information (this requires public-key protocols), or will even not learn the secret of the users (this requires zero-knowledge protocols). Note that in this case the procedure does not authenticate the user but rather one's device. In order to increase the security, the user should authenticate with respect to the device, using something that one alone knows. This makes the device useless if it is stolen.

In general, one also requires that the computer authenticates itself to the person logging on. If both parties are authenticated to each other, the term *mutual authentication* is used.

2.1.2. Data Authentication

Data authentication consists of two components: the fact that data has not been modified (data integrity) and the fact that one knows who the sender is (data origin authentication).

2.1.3. Data Integrity

A data integrity service guarantees that the content of the message, that was sent, has not been tampered with. Data integrity by itself is not meaningful: it does not help one to know that the data one has received has not been modified, unless one knows it has been sent directly by the right person. Therefore it should always be combined with data origin authentication.

One should always be alert for possible intruders in their network or in their communication system. A well-known example is the Internet that connects universities and companies world-wide. Electronic mail over the Internet does not offer any security. As a consequence, an educated computer user can tap into the messages that are being transmitted over the line. It is very easy to read and modify someone’s electronic mail, which is commonly seen as being private.

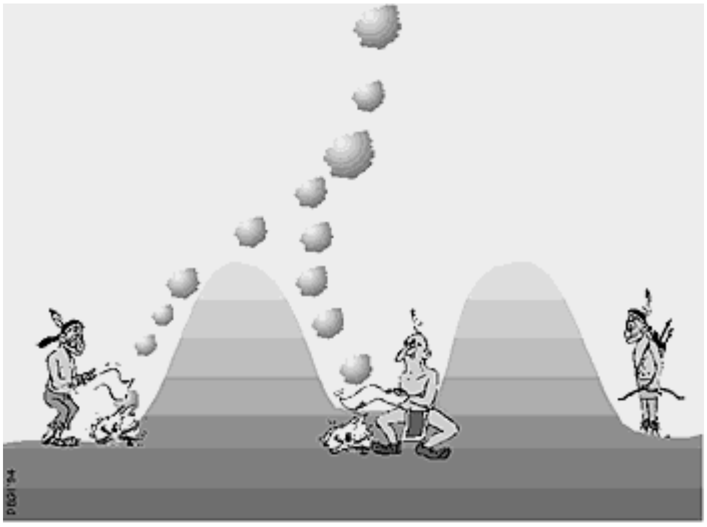

Fig. 2. Data Integrity

In general, take the point of view of figure 2. There is a person A, who sends a message to person B. There is also an enemy who taps the line between them. If one doesn't support data integrity, this enemy can just change the message and then relay it to B. B will not see that the message has been tampered with and will assume A really intended it the way he got it. One could argue that active wire-tapping is difficult. In general wire-tapping is only a matter of cost: tapping a telephone line is obviously easier than tapping a coaxial cable or a micro-wave. Active wire-taps (modifying and then relaying the messages) are also more difficult than passive wire-taps (listening in on the messages).

2.1.4. Data Origin Authentication

Here one wants to make sure that the person who is claiming to be the sender of the message really is the one from whom it originates. In figure 3, if a person A sends a message to B, but the enemy intercepts it and sends it to B, claiming A has sent it, how can B be sure of the real origin of this data? A variation on this theme is: the enemy could send a message to B claiming it A is the originator. Thanks to cryptography, there are techniques to ensure against this type of fraud.

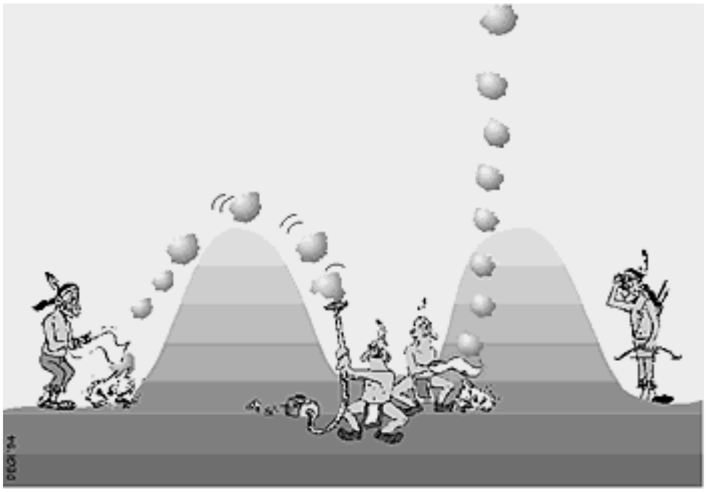

Fig. 3. Data origin authentication

2.1.5. Non-repudiation of Origin

Non-repudiation protects against denial by one of the entities involved in a communication of having participated in all or part of the communication. Non-repudiation with proof of origin protects against any attempts by the sender to repudiate having sent a message, while non-repudiation with proof of delivery protects against any attempt by the recipient to deny, falsely, having received a message.

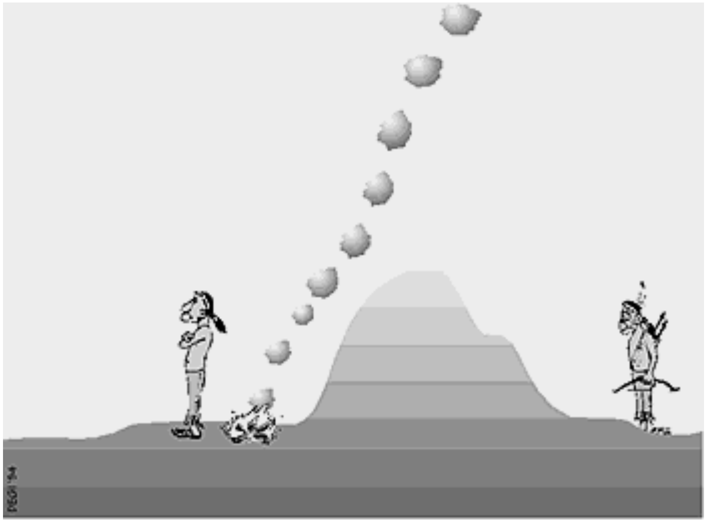

Fig. 4. Non-repudiation of origin

An example will illustrate the importance of non-repudiation of origin. Suppose B is the owner of a mail-order company and decides to let the customers order through electronic mail. It is really important that B can show to an arbitrary third party that A really ordered the things B is claiming otherwise it would be easy for a customer to deny the purchase of the goods (see figure 4). In a paper and pencil world, non-repudiation is provided by a manual signature.

2.1.6. Data confidentiality

This aspect of data security certainly is the oldest and best known. The example of Caesars cipher given in the introduction clearly demonstrates this. The fact that confidentiality was considered to be much more important than authentication of both sender and data, together with non-repudiation of origin can be explained as follows: the latter services have been provided implicitly by the physical properties of the channel: a letter was written in a recognizable handwriting, with a seal and a signature.

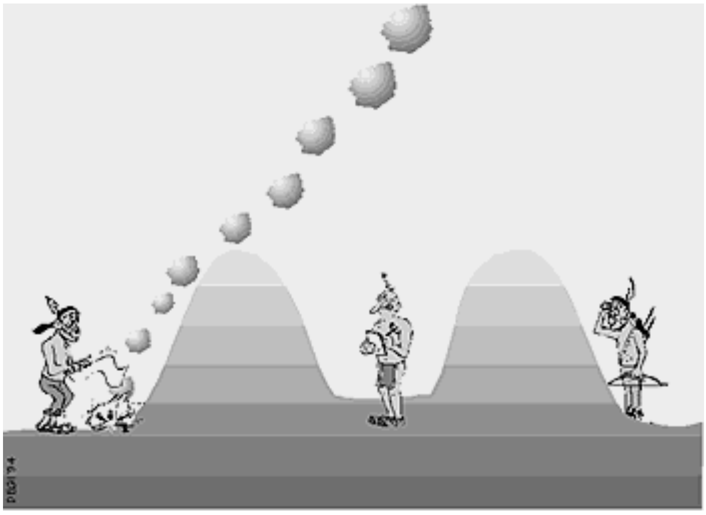

Fig. 5. Data confidentiality

With data confidentiality one tries to protect oneself against unauthorized disclosure of the message. Referring to figure 5, if A sends a message to B, but the enemy intercepts it, one wants to make sure that this enemy never understands his contents. Confidentiality protection is very important in the medical world and also in the banking sector. World-wide there are several million transactions each day and all of these have to be passed from one financial institution to another. If there were no way to protect confidentiality, everybody would be able to see who had purchased what, who has made what kind of withdrawal, and so on.

Clearly this would violate individuals and companies rights to privacy. In order to provide confidentiality, it is necessary to transform the message with a cipher.

2.2. Cryptographic Primitives

The above cryptographic services can be realized by several cryptographic primitives: one distinguishes between primitives for encryption, primitives for authentication, and cryptographic protocols. Encryption primitives can be used to provide confidentiality, authentication primitives can be used to provide data authentication. The protocols for user authentication and for key management will also be discussed.

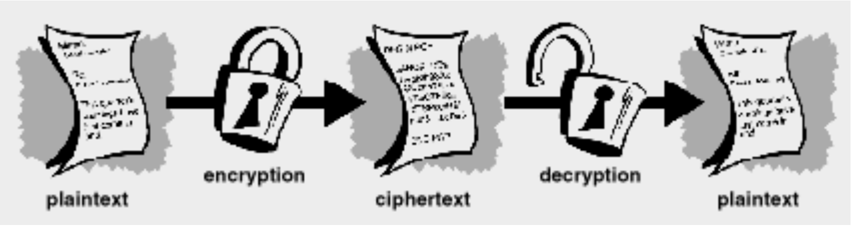

Fig. 6. Encryption and decryption

2.2.1. Encryption primitives

In cryptography one often makes use of encryption. With encryption one transforms the cleartext (or plaintext) into ciphertext. To get back to the original text, one applies the inverse transformation, called decryption. These transformations themselves are public: this makes it possible to analyze these algorithms and to develop efficient implementations. However they use a secret parameter: the keys which are known only by the sender and/or the receiver.

This key is the only thing one needs to know in order to encipher or decipher. Thus it is really important to manage one's keys and keep them secret where necessary.

In the following, two types of encryption primitives, symmetric or conventional ciphers and asymmetric or public-key ciphers are discussed.

2.2.2. Symmetric Ciphers

Basically there are two kinds of encryption-schemes. The oldest ones and most used until now are the symmetric ciphers. In these schemes, the key used to decipher the ciphertext is equal to the one used to encipher the plaintext.

The best known cipher in this category is the Data Encryption Standard (DES) that was adopted in 1977 by the American NBS (National Bureau of Standards) as FIPS 46. Since then it has been used all over the world and until now no major flaws have been discovered.

The DES makes use of a 56-bit key which is unfortunately short. Researchers have estimated that exhaustively searching for all possible values of this key in 1 day will currently require an investment of about US$ 200,000 (this requires only a few pairs of plaintext and corresponding ciphertext).

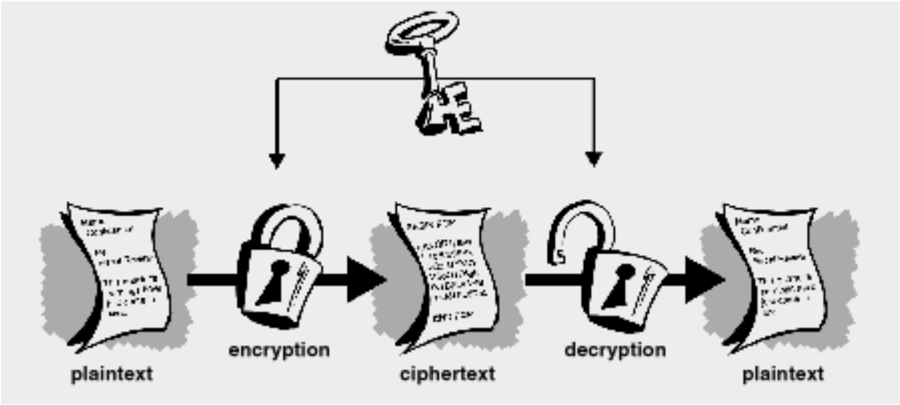

Fig. 7. Symmetric cipher

During the last years E. Biham and A. Shamir, and later M. Matsui have published attacks which break DES in the academic sense (i.e., they require significantly less operations), though this is no threat to the DES in practice, since they require huge amounts of known or chose plaintexts respectively.

Better security can be achieved using the triple-DES. In this way, one can effectively obtain a key of 112 bits and this is sufficiently large. At the same time, one is protected against further improvements in academic attacks on the DES.

It is not sufficient to choose a secure cipher; one also has to specify a secure mode of operation. Depending on the nature of the communication channel or storage space, one will choose between Cipher-Block-Chaining (CBC), Cipher-Feedback (CFB), and Output-Feedback (OFB). Encryption block by block (or Electronic Code Book (ECB) mode) will only be used for encryption of keys.

2.2.3. Asymmetric ciphers

The asymmetric or public-key ciphers are the most recent cryptographic tools. In contrary to the symmetric systems the key used to encipher and the one used to decipher are different. Each partner thus has two keys. One keeps one key secret and makes the other one public. If A wants to send a message to B, it's just enciphered with B's public key. Since B is the only one who has access to the secret key, B is the only one who can decipher the message and read the contents.

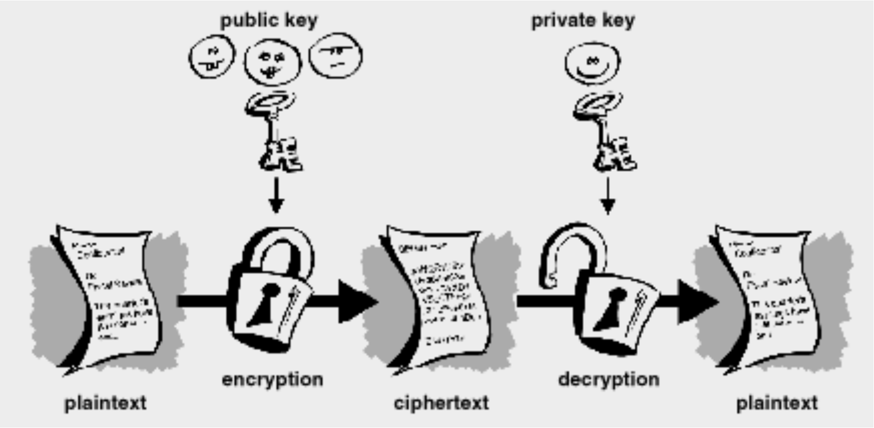

Fig. 8. Key management with asymmetric cipher

The most popular public-key cipher is the RSA system (RSA stands for Rivest, Shamir and Adleman, the names of the three inventors). The security of this scheme is related to the mathematical problem of factorization: it is easy to generate two large primes and to multiply them, but given a large number that is the product of two primes, it requires a huge amount of computation to find the two prime factors.

At the moment of writing of this text, the biggest number that has been factorized was about 430 bits long and attacks on numbers of 512 bits have been announced for 1997. Therefore the absolute minimum length of the key in the RSA system has to be set at 640 bits; 768 or 1024 bits are required for any system that requires security for more than a few months.

2.2.4. Symmetric versus asymmetric ciphers

The biggest drawback of the asymmetric systems up until now has been the relative low performance compared to the symmetric ones. For example, the implementation of DES on a 586/90 PC could achieve a 15 Mbit/s encryption rate while the RSA implementation on the same PC has only a 6 Kbits/s rate. Thus as one can see the DES is typically 1000 times faster than the RSA-scheme.

Public-key systems provide significant benefits in terms of key management: if every user generates their own key, only an authentic channel is required, eliminating (expensive) secret channels like couriers.

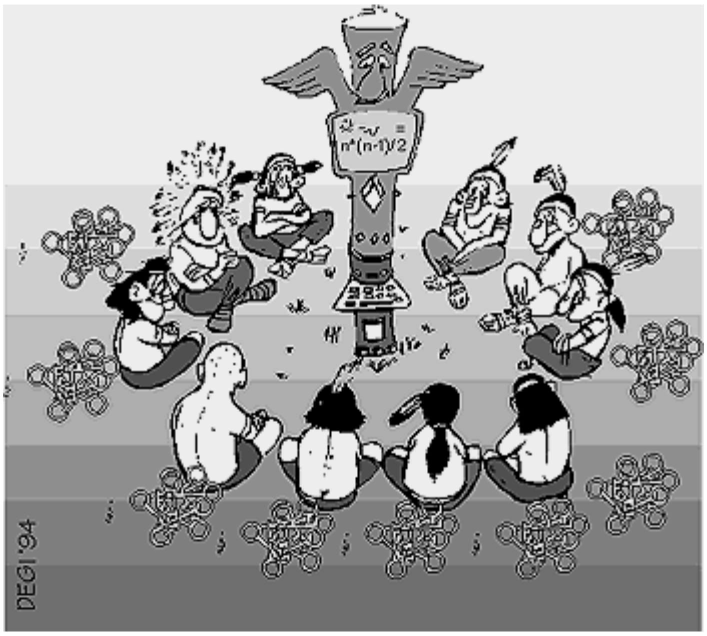

Fig. 9. Key management with asymmetric cipher

In systems without a central trusted server, the number of keys can be reduced. Indeed, suppose one has a network of n users each of whom wanting to communicate with the others. Since each communication requires a secret key, the total number of keys required equals n*(n-1)/2.

In the public-key system each user only needs a personal public/secret key pair, yielding a total of only 2n keys. If n equals 1000 this would mean 2000 versus 499500. In systems with a central management system, both approaches require the same number of keys. However, the central system can be off-line in the case of use of public key technology, which reduces the cost and minimizes the security risks.

In practice one thus often encounters hybrid systems in which one uses a public-key system for the distribution of the secret keys and a symmetric cipher for the bulk encryption of the data.

2.3. Authentication primitives

2.3.1. One-way functions and hash codes

A one-way function is defined as a function f such that for every x in the domain of f, f(x) is easy to compute; but for virtually all y in the range of f, it is computationally infeasible to find an x such that y=f(x). In addition one requires that it is hard to find a second pre-image: given an x and the corresponding value of f(x), it should be hard to find an x' different from x which has the same image under f.

One-way functions are used to protect passwords: one will store a one-way image of the password in the computer rather than the password itself. One applies then the one-way function to the input of the user and verifies whether the outcome agrees with the value stored in the table.

A hash function is a function which maps an input of arbitrary length into a fixed number of output bits. In order to be useful for cryptographic applications, a hash function has to satisfy some additional requirements. One can distinguish two types of hash functions. A MAC (Message Authentication Code) that uses a secret key, and an MDC (Manipulation Detection Code) that works without a key. For a MAC one requires that it should be impossible to compute the MAC without knowledge of the secret key. For an MDC one requires that it is a one-way function, and - in most cases - that it is collision resistant, which means that it should be hard to find two arguments hashing to the same result.

Hash functions can be used to protect the authenticity of large quantities of data with a short secret key (MAC), or to protect the authenticity of a short string (MDC). Sometimes an MDC is used in combination with encryption, which can yield protection of both confidentiality and authenticity.

There are several schemes which have been proposed for use as hash functions. The widely used construction for a MAC is the CBC mode of the DES (with an additional output transformation), as specified in ISO-9797. Several MDC's have been constructed based on the DES. Other dedicated designs are SHA (Secure Hash Algorithm or FIPS 180), and RIPE-MD 160. These hash functions achieve a very high throughput (Mbit/s), even in software implementations.

2.3.2. Digital signature

Public-key techniques can also be used for other purposes than for enciphering information. If Alice adds some redundancy to her message and transforms the result using her secret key, anyone who knows her public key can verify that by whom this message was sent(by verifying the redundancy). In this way one can create a digital signature, which is the equivalent of the hand-written signature on a document.

Since it is not physically connected to the signed data or the originator, it will depend on this data and on the secret key of the originator. Several signature schemes have been proposed. The RSA public-key cryptosystem is the only one which can be used for both enciphering and digital signatures. Schemes which can only be used for digital signature purposes are the DSA and the Fiat-Shamir scheme.

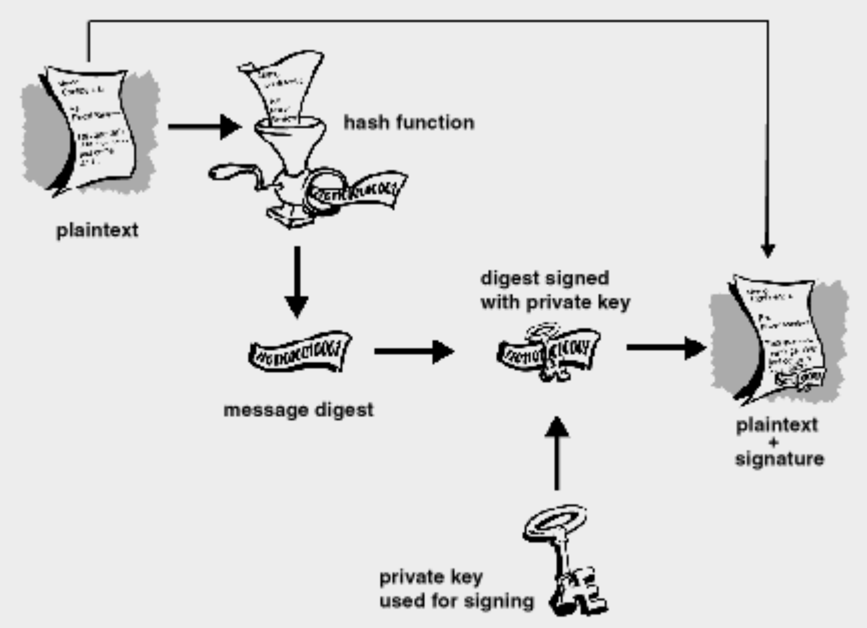

Fig. 10. Secure digital signatures

Note that it is possible to produce a digital signature based on conventional ciphers like the DES. However, these schemes are less efficient in terms of memory and computations. Other constructions use a conventional cipher in combination with tamper resistant hardware: this offers only a limited protection.

Assume Bob has received from Alice a digitally signed message. If Alice subsequently denies having sent the message, Bob can go to a third party (e.g., a judge), who will be able to obtain Alice's public key. Subsequently he can verify the validity of the signature. In this way a digital signature can provide non-repudiation of origin. It is easy to see that it provides in addition data authentication, i.e., data integrity and data origin authentication.

2.3.3. Hash functions versus digital signatures

Hash functions can only be used in a situation where the parties mutually trust each other: they cannot be used to resolve a dispute (unless one uses, in addition tamper resistant hardware).

As in the case of encryption, hash functions tend to be three orders of magnitude faster than digital signatures. This explains why in general one will first compute the hashcode of the message with a fast hash function and subsequently apply the digital signature to this short hashcode. This provides digital signatures which are not only faster and shorter, but also more secure.

2.4. Cryptographic Protocols

A cryptographic protocol is an interaction between one or more entities to achieve a certain goal. In fact, encryption and digital signatures can be seen as a special case of cryptographic protocols.

While a huge number of protocols have been developed, this section is restricted to two types of protocols: protocols for user authentication and protocols for key management.

2.4.1. User authentication protocols

The design of cryptographic protocols for user authentication is very complex. A large number of protocols have been presented in the available literature, many of which exhibit some weaknesses. The simplest protocol providing unilateral authentication consists of sending a password.

More complex challenge-response protocols can be designed in which the user does not transmit their secret information. They are based on an encryption algorithm, a MAC or a digital signature and the use, in addition, of so called nonces (never used more than once):

random numbers, sequence numbers or time stamps. More complex protocols are required to achieve mutual authentication.

2.4.2. Key Management Protocols

One of the main links in the cryptographic keychain is the key management protocol: every cryptographic service will make use of cryptographic keying material, whose confidentiality and/or integrity has to be protected. For the distribution of this keying material, one can use a new cryptographic primitive, and ultimately, a physical channel.

In this way one builds a key hierarchy: secret keys for bulk encryption with a symmetric cipher system will be encrypted using an asymmetric cipher system and signed with a digital signature scheme. The public keys of the asymmetric cipher can be distributed via an authentic channel which can be provided for example by combining conventional mail with voice authentication. An alternative is to sign these public keys with a single master key: now one only has to distribute a single master key via an authentic channel. These signed public keys are called certificates. The central authority certifies that a certain public key belongs to a particular user. The commonly used scheme nowadays in based on the ITU-T X.509 recommendation.

Note that there also exist public-key protocols which result in the agreement of a secret key between two parties, by exchanging public keys or parameters. A well known example in this class is the Diffie-Hellman key agreement scheme. This protocol is different from a key transport protocol, in which one party generates the secret key and enciphers it with the public key of the other party. The key agreement protocols have the advantage that they result in an increased security level.

In the context of public-key cryptography, revocation of public keys is very important: once the user's secret key is compromised, anybody can read their messages or forge their signatures. Although public-key systems require no on-line central management system, the system has to provide a means to protect the user in the case by warning the other users that their public key is no longer valid.

3. 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.

3.1. XML Cryptography

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.

3.2. 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 their 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.

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.

4. Projected benefits and potential problems

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.

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. 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.

5. Cryptography in the Real World

5.1. Applications of Cryptography

In the information dependent world in which one now lives, cryptography can be found all around, often in places where one would not expect it. When people think about encryption they tend to think about vast computer banks processing military and diplomatic communications, or a world war two rotor cipher machine slowly deciphering an order. In reality, cryptography - although obviously essential for the military and diplomatic services - has many commercial uses and applications. From protecting confidential company information, to protecting a telephone call, to allowing someone to order a product on the Internet without the fear of their credit card number being intercepted and used against them, cryptography is all about increasing the level of privacy of individuals and groups. For example, cryptography is often used to prevent forgers from counterfeiting winning lottery tickets. Each lottery ticket can have two numbers printed onto it, one plaintext and one the corresponding cipher. Unless the counterfeiter has crypt analysed the lottery's cryptosystem one will not be able to print an acceptable forgery.

In a world where virtually all data of any importance is held on a computer system the necessity of cryptography cannot be disputed.

5.2. Politics of Cryptography

Widespread use of cryptosystems is something most governments are not particularly happy about - precisely because it threatens to give more privacy to the individual, including criminals. For many years, police forces have been able to tap phone lines and intercept mail, however, in an encrypted future that may become impossible.

This has lead to some pretty strange decisions on the part of governments, particularly the United States government. In the United States, cryptography is classed as a munitions and the export of programs containing cryptosystems is tightly controlled. In 1992, the Software Publishers Association reached agreement with the State Department to allow the export of software that contained RSA's RC2 and RC4 encryption algorithms, but only if the key size was limited to 40 bits as opposed to the 128 bit keys available for use within the US. This significantly reduced the level of privacy produced. In 1997 this was increased to 56 bits. The US government has proposed several methods whereby it would allow the export of stronger encryption, all based on a system where the US government could gain access to the keys if necessary, for example the *clipper* chip.

The resolution of this issue is regarded to be one of the most important for the future of e-commerce.

6. Why is cryptography important for an 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 one's communications secret while penetrating those of the opponent. It may be less obvious why it is 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 one’s 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 one's system and steal the secrets and the secrets of one's customers and business partners. The cracker may use one's 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 one's customers and business partners, or even third parties, suffer because of insufficient computer security at one's organization, they may sue one 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.

7. Conclusion

From e-mail to cellular communications, from secure Web access to digital cash, cryptography is an essential part of today's information systems. Cryptography helps provide accountability, fairness, accuracy, and confidentiality. It can prevent fraud in electronic commerce and assure the validity of financial transactions. It can protect one’s anonymity or prove their identity. It can keep vandals from altering one’s Web page and prevent industrial competitors from reading one’s confidential documents. And in the future, as commerce and communications continue to move to computer networks, cryptography will become more and more vital.

But the cryptography now on the market doesn't provide the level of security it advertises. Most systems are not designed and implemented by cryptographers, but by engineers who think cryptography is like any other computer technology. It's not. One can't make systems secure by tacking on cryptography as an afterthought. One has to know what one is doing every step of the way, from conception through installation.

Billions of dollars are spent on computer security, and most of it is wasted on insecure products. After all, weak cryptography looks the same on the shelf as strong cryptography. Two e-mail encryption products may have almost the same user interface, yet one is secure while the other permits eavesdropping. A comparison chart may suggest that two programs have similar features, although one has gaping security holes that the other doesn't. An experienced cryptographer can tell the difference. So can a thief.

The people who break cryptographic systems don't follow rules; they cheat. They can attack a system using techniques the designers never thought of. Art thieves have burgled homes by cutting through the walls with a chain saw. Home security systems, no matter how expensive and sophisticated, won't stand a chance against this attack. Computer thieves come through the walls too. They steal technical data, bribe insiders, modify software, and collude. The odds favor the attacker: defenders have to protect against every possible vulnerability; but an attacker only has to find one security flaw to compromise the whole system.

Present-day computer security is a house of cards; it may stand for now, but it can't last. Many insecure products have not yet been broken because they are still in their infancy. But when these products are widely used, they will become tempting targets for criminals. The press will publicize the attacks, undermining public confidence in these systems. Ultimately, products will win or lose in the marketplace depending on the strength of their security.

No one can guarantee 100% security. But one can work toward 100% risk acceptance. Fraud exists in current commerce systems: cash can be counterfeited, checks altered, credit card numbers stolen. Yet these systems are still successful because the benefits and conveniences outweigh the losses. Privacy systems -- wall safes, door locks, curtains -- are not perfect, but they're often good enough. A good cryptographic system strikes a balance between what is possible and what is acceptable.

Strong cryptography can withstand targeted attacks up to a point -- the point at which it becomes easier to get the information some other way. A computer encryption program, no matter how good, will not prevent an attacker from going through someone's garbage. But it can prevent data-harvesting attacks absolutely; no attacker can go through enough trash to find every AZT user in the country.

The good news about cryptography is that one already has the algorithms and protocols needed to secure our systems. The bad news is that that was the easy part; implementing the protocols successfully requires considerable expertise. The areas of security that interact with people -- key management, human/computer interface security, access control -- often defy analysis. And the disciplines of public-key infrastructure, software security, computer security, network security, and tamper-resistant hardware design are very poorly understood.

Laws are no substitute for engineering. The U.S. cellular phone industry has lobbied for protective laws, instead of spending the money to fix what should have been designed correctly the first time. It's no longer good enough to install security patches in response to attacks. Computer systems move too quickly; a security flaw can be described on the Internet and exploited by thousands. Today's systems must anticipate future attacks. Any comprehensive system designed today is likely to remain in use for five years or more. It must be able to withstand the future: smarter attackers, more computational power, and greater incentives to subvert a widespread system. There won't be time to upgrade them in the field.

History has taught: never underestimate the amount of money, time, and effort someone will expend to thwart a security system. It's always better to assume the worst. Assume one’s adversaries are better than they are. Assume science and technology will soon be able to do things they cannot yet. Give oneself a margin for error. Give oneself more security than needed today. When the unexpected happens, one would be glad one did.

Bibliography

[Cryptography World, 2006]
Cryptography World. *The Cryptography Introduction and Guide.* Retrieved: July 18, 2006 from
http://www.cryptographyworld.com/

[Gollman, 2006]
Gollman, Dieter. *Computer Security, Second Edition,* John Willey & Sons Ltd, 2006.

[Kessler, 1998]
Kessler, Gary C. (May, 1998) *An Overview of Cryptography*
http://www.garykessler.net/library/crypto.html

[Microsoft, 2006]
Microsoft Research. *Cryptography.* Retrieved: July 21, 2006 from
http://research.microsoft.com/crypto/

[Terry, 2006]
Terry, Ritter. (2006, January 20) *A Basic Introduction to Crypto.*
http://www.ciphersbyritter.com/LEARNING.HTM

[SSH, 2006]
SSH Communications Security. *An Introduction to Cryptography.* Retrieved: July 20, 2006 from
http://www.ssh.com/support/cryptography/introduction/

[Webopedia, 2006]
Webopedia. *Cryptography.* Retrieved: July 10, 2006 from
http://en.wikipedia.org/wiki/Cryptography