Consiglio Nazionale delle Ricerche

A brief overview of modern cryptographic tools with application to Internet

C. Anticoli

IIT B4-01/2002

Nota Interna

Maggio 2002

Istituto di Informatica e Telematica

A brief overview of modern cryptographic tools with application to Internet

Claud Anticoli

Abstract

Here a brief overview of some of the cryptographic tools used in Internet today is illustrated. This document briefly introduces cryptography starting from a short resume of historical events showing the evolution and role played by cryptography narrating some representative real life events in history. A more formal introduction mentioning the advantages obtainable with cryptography is then undertaken starting from symmetric and asymmetric cryptography. Other uses of cryptography are then explained to finally present two widely used communication protocols in Internet. Representative examples are provided reverting to a minimal mathematical approach.

Introduction

Information security is an issue that is receiving a growing attention in today's society and one of the main reasons for this is attributable to the Internet phenomena. During the design of Internet, great importance was given to its functionality rather than other aspects, such as information security, thus making it an inadequate means for certain applications. The unpredictable success of Internet, however, has not been influenced by it's own downfalls and it has by now proved to be a fundamental tool that cannot be discarded in almost all sectors of today's economy. For this reason great effort has been invested in the improvement of the services offered by Internet and one of these is obtained using cryptographic techniques that render Internet not only a useful means of doing business, but at times even indispensable. In fact, through cryptographic techniques, one can obtain: secure communication connections, identity proof, electronic signatures (arguably safer than hand signing a check or credit card receipt) among other benefits.

This work tries to give a general overview of the cryptographic scene and will describe some of the actual algorithms and protocols actually implemented by security related applications in Internet undertaking a minimal mathematical approach in order to give an idea of the typology of artifacts used. This work is intended to introduce the reader only to the cryptographic and algorithmic aspects that are involved in Internet omitting explanation of how exactly these concepts are practically implemented.

A short section will be dedicated to the history of cryptography and intends to mention only few of the many cryptographic related events that have conditioned today's standards. The main purpose of this section is to render the idea that cryptography is by no means a guaranty and history teaches us that often, whoever took this for granted paid serious consequences. A general description of present day ciphers will successively be undertaken including the introduction to one of the most sensational advents in cryptography during last century: the invention of asymmetric cryptography as a solution to the key distribution problem.

Other uses of cryptography will successively be presented to eventually conclude with two of the most famous key exchange communication protocols used in Internet: Kerberos and X.509.

A short history of cryptography

It is widely believed that cryptography dates back as far as 2000 B.C. and leads us to the Egyptian civilization where hieroglyphics were intentionally modified to make the understanding of the inscriptions more difficult. Hieroglyphics were engraved on tombstones narrating the story of important rulers and kings. The cryptic modifications were applied to make the inscriptions more intriguing to the audience but had no intention of hiding the text. It was believed that a story told this way, was more important and therefore worthy only of a 'king's story'.

The first recorded evidence of cryptography used to conceal correspondence shows that the Spartans, around 400 B.C., employed a cipher device called a 'scytale' to encipher and decipher their messages. It consisted of a parchment tape that was wrapped around a baton and a message was then written on the parchment parallel to the cylinder axis. The parchment was then unwrapped resulting in a transposition of the clear text making it unreadable and very difficult for those days to decipher unless in possession of a baton of equal radius.

Julius Caesar himself referred to encipherment techniques to communicate with his trusted collaborators of the senate and applied a mono-alphabetic substitution on the text by simply replacing each letter with the its preceding letter in the alphabet (e.g. 'f' becomes 'e', 'a' becomes 'z' and 'yes' becomes 'xdr').

The major downfall of these techniques is that once unmasked, they are completely of no use and are easily deciphered. The necessity for a different approach produced the first cryptographic schemes based on a 'shared finite key' used for the encipherment and decipherment of the message. This was a revolutionary discovery and indeed its use is still in vigor today. One of the first schemes of this type consists in a letter substitution according to a randomly generated key, which maps each letter to another in a meaningless way. This new approach implied that even if you were an expert cryptographer, without the key, the only apparent approach would have been to try out all 403 billion billion billion billion possible key combinations and verify which rendered a meaningful deciphered text (in reality thanks to the birthday problem this number is practically drastically reduced).

It is at this point in history that the endless battle between cryptographers and cryptanalysts commences. The first to make significant advances in the deciphering of the new 'unbreakable' code were the Arabs, namely Qalqashandi, who wrote a book on how to perform frequency analysis of ciphered text in order to discover the key and unveil the text. This technique relies on the fact that that certain letters of the alphabet statistically occur more often than others so for example, for an English text, the ciphered letter that occurs the most is probably an 'e' seeing that 'e' is the most recurring letter in the English language.

At the time this event astonished the cryptographic community because the mono alphabetic ciphering was thought to be 'unbreakable' and this typology of stupefaction was destined to repeat itself endlessly to the point that even today, cryptographers are starting to suspect that what seemed practically impossible to decipher is not only about to be deciphered but is already being deciphered by institutions with adequate resources (e.g. military entities or government institutions).

Queen Mary of Scots, found guilty of plotting for the English crown, was betrayed by her excessive trust in the fact that her ciphered correspondence was unreadable for the enemy. Instead the English Queen Elisabeth's investigators were decoding her messages and collected enough evidence to prove her guilt and unveil all her collaborators which were obviously sentenced too. This was achieved by applying a letter frequency analysis and this story is probably the most renowned episode of how cryptography and cryptanalysis influenced history and the moral behind the story shows us the two faces of cryptography (beneficial if successful but disastrous if broken)

After this breakthrough in cryptanalysis all efforts were aimed towards building a cryptographic technique, which at the time was predominantly vulnerable to the frequency analysis attack, immune to cryptanalysis. To this scope, a pioneer of western cryptography, the Italian Leon Battista Alberti, introduced the concept of a polyalphabetic substitution cipher where a different 'mapping key' is used for each letter and then repeated according to a certain mechanism. He also formalized the concept of symmetric cryptography and is considered as one the founding fathers of modern cryptography. Towards the 16th century the French cryptographer Blaise de Vigenere devised a final version of a ciphering technique that was again thought to be 'unbreakable' seeing that it resisted all frequency analysis attacks until Antoine Rossignol discovered a mathematical property that allowed to eventually conduct a frequency analysis in order to break the code. In this occasion Louis XIV benefited from Rossignol's discovery and obtained the Huguenots surrender by deciphering a message and 'reciphering' another message under the same key suggesting immediate surrender seeing that any hope of victory was futile. Louis XIV himself then later fell victim to the same feat when the English deciphered his messages revealing his plans for Poland.

Following these advents, various finite key cryptographic mechanisms were invented offering variants in order to resist the known cryptanalyst attack methods but were systematically deciphered. An interesting aspect of this matter is that whilst cryptanalysts were believed to require a linguistic back round in order to succeed in their quest, as the systems became more complex, mathematicians slowly proved to be more adapt to the task. This is explained by the growing complex nature of the cryptographic mechanisms and the Polish mathematician Marian Rejewski achieved one of the most clamorous successes in the history of cryptoanalysis when he 'broke' the German Enigma code used by the German U-Boot submarines in the Atlantic Ocean during the Second World War. The Enigma code was widely believed to be 'absolutely' secure unless the daily keys fell into enemy hands. The Germans firm belief in the security of the code revealed to be a fatal error leading to the defeat of their marine forces in the Atlantic Ocean, a key episode in the outcome of the Second World War. The Americans too, thanks to the deciphering of the Japanese ciphered codes, gained victory in the Pacific Ocean. The Enigma machine offered a mechanism that actuated a poly-alphabetic substitution but practically never reused the same 'key map'.

This meant that each letter was coded according to a different substitution key map and any mapping to the same letter was apparently nothing but a mere coincidence. The initial setup of the machine was of vital importance and was considered as the ‘key’ for the message. The machine offered 15 billion billion billion initial setup configurations (in one it’s initial versions). Rejewski applied a mathematical property of the mechanism to attack the problem and reduced the amount of initial possible setup conditions (i.e. the daily key) to roughly 150 000 by analyzing a sufficient amount of clear and corresponding ciphered text. The German's use of the machine was, from a security point of view, inappropriate because, for example, the daily transmission of weather bulletins (at exact times of day) provided the allies secret services with enough ciphered text and corresponding clear text in order to apply Rejewski’s method. At Bletchley Park (a branch site of the British Secret Services) auto-mechanisms had been devised to manually try all the possible initial setup conditions remaining. These machines, called ‘bombes’, were devised by Rejewski and later developed and extended by Alan Turing and represent one of the first prototypes of today’s electronic calculators (computers). Even though the German’s tried to counter act, by augmenting the machines complexity or by transmitting only ‘double’ enciphered text, the allies almost always managed to decipher the messages by simply constructing bigger and faster ‘bombes’.

There are many examples in history of cryptography used by the highest and most influential authorities through out the world and almost all of those based on a finite key mechanism have been deciphered. Cryptanalysis has played an important role in the outcome of many colossal events in history and perhaps the most illustrative example of this is the Zimmerman Telegraph episode. Arthur Zimmerman was the German Minister of Foreign Affairs in 1917 and in January, he sent a telegram to the German Ambassador in Mexico ordering him to invite Mexico to unite with Japan and to attack America promising the state of Texas and Arizona as a reward. This message was intercepted by the allies and sent immediately to America. It was this event that influenced America to finally abandon their neutrality and enter the the First World Was bringing victory to the Allies. One of the most difficult aspects of this episode was the handling of the information in such a way as to not raise suspicion that the Germans secret communications were being deciphered. Today, it is believed that governments purposely withheld information that would have saved lives in order to maintain secrecy in the deciphering abilities of the enemy code. These, however, are aspects of politics that go beyond the intent of this document. In fact, even though the historical events narrated may not appear to be pertaining to the scope of this document, they have been included to render an idea of the general importance and usage of cryptography, a lesson only 4000 years of history can teach.

Cryptography today

The ciphers mentioned above will now be more formally presented along with other types. Modern ciphers use a combination of different techniques, so it is important to define a basis of the cryptographic techniques available in order to obtain a synthetic yet precise description of those more complex.

Before proceeding it is important to have a clear understanding of the intent and goals of cryptography. By definition, cryptography is the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication and data origin authentication.

Cryptography can realize various objectives and a framework upon which all are derived is presented as five different goals, namely:

- confidentiality is the service used to keep the content of the information hidden from all except those authorized,
- data integrity is a service which addresses the unauthorized alteration of data,
- data authentication is a service related to verifying a claimed identity,
- electronic certification and digital signature provides guaranties against unauthorized modification and forgery of electronic documents
- non-repudiation is a service which prevents an entity denying previous commitments or actions.

By information we intend any data that is somewhat representable in a written form.

Ciphers belong to one of two main categories. These are ciphers based on symmetric techniques and ciphers based on asymmetric techniques. Asymmetric ciphers were formalized in the late 1970's and consist in two different methods and keys (but somewhat related) for ciphering and deciphering the text. Symmetrical ciphers on the other hand use the same key (or if not closely related) but not necessarily the same method to cipher and decipher messages and have existed since the invention of 'finite key' cryptographic schemes.

Apart from the apparent differences, another factor of great relevance distinguishes the two techniques (even though not necessarily true always). Symmetric techniques require a preventive secret sharing whilst asymmetric techniques do not and can privilege the users in obtaining a shared secret. This latter technique, in fact, was formalized initially to affront the key distribution problem resulting in a new branch of cryptography which is today used not only to resolve prior key sharing obstacles but has opened a whole new scenario of the uses of cryptography.

This work will present some renowned modern symmetric and asymmetric (public key schemes) techniques and will also give an overview of the protocols used to cover the key distribution problem. Other uses of cryptography will also be explained in a descriptive manner omitting 'real life' examples.

Symmetric Techniques

Of all the services of cryptography at disposition, symmetric encryption is first in the historical time-line of cryptography and therefore is treated first. A simple categorization of symmetrical cipher systems will be presented in order to then illustrate a few of the modern cipher algorithms used in Internet today.

A formal definition of a symmetric encryption scheme can be the following:

Definition: Consider an encryption scheme consisting of the sets of encryption and decryption transformation $\{E_e\text{: } e \varepsilon K\}$ and $\{D_d\text{: } d \varepsilon K\}$, respectively, where K is the

> key space. The encryption scheme is said to be symmetric-key if for each associated encryption/decryption key pair (*e*, *d*), it is computationally "easy" to determine *d* knowing only *e*, and to determine *e* from *d*. [Menezes, van Oorschot, Vanstone]

More attention should be paid to the term "easy" but for the purposes of this document will be omitted. The reader should, however, be aware of the necessity and existence of a rigorous approach which defines "computationally easy" in a theoretical and practical manner.

Block ciphers

A block cipher is an encryption scheme which breaks up the plaintext message to be transmitted into strings (called blocks) of a fixed length over an alphabet and encrypts one block at a time as apposed to a stream cipher that simply encrypts one letter at a time. Below is a list and brief explanation of different categories of block ciphers

- **Simple substitution ciphers**: Substitution ciphers are block ciphers which replace symbols (or group of symbols) by other symbols or groups of symbols. The number of distinct simple homophonic substitution ciphers is *q*! where *q* is the alphabet size (typically 26!).

- **Homophonic substitution cipher**: Here, different corresponding strings substitute a variety of block symbols in the plaintext message. This uniforms letter frequency at the expense of harder deciphering and expands ciphered text size.

- **Poly-alphabetic substitution ciphers**: A poly-alphabetic substitution cipher is a block cipher which uses a different key for the encoding of each correspondent letter in the block according to a mechanism which can depend on the symbol order in the block, on previous plaintext symbols of even cipher text symbols. Symbol frequency is not preserved which offers a great advantage in security making a frequency analysis more difficult. However, in many cases cryptoanalyzing a poly-alphabetic substitution cipher is not significantly harder than decoding a simple substitution cipher once the block length is determined. In fact, once the block length is determined, the cipher text can be divided into t groups (t being the block length) and a frequency analysis can be carried out on each group.

- **Transposition ciphers**: Another class of symmetric-key ciphers is the simple transposition cipher, which simply permutes the symbols in a block

- **Composition of ciphers (Product Ciphers)**: In order to describe product ciphers, the concept of a composition of functions is introduced. Compositions are a convenient way of constructing more complicated functions from simpler ones. Involutions are a simple class of functions which allow deciphering of a text by simply reapplying the cipher scheme used to cipher the plaintext. Note that a composition of involutions isn't necessarily an involution itself. Simple substitution ciphers and transposition ciphers do not individually provide very high level security but a combination of the two increases this level notably obtaining strong ciphers which some of the most practical and effective symmetric-key systems today make use of.

Confusion is an entity which describes the extent of making the relationship between key and cipher text as complex as possible (usually a substitution) where as diffusion refers to rearranging or spreading out the bits in the message so that redundancy in the plaintext is spread out over the ciphertext (usually via transposition). A round is then defined as a process which adds confusion or diffusion and each successive round is supposed to increase the respective confusion or diffusion. Modern block ciphers should obtain both in equal measure making the 'breaking' of the scheme very difficult implementing various combinations of the above ciphers.

Stream Ciphers

Stream ciphers are very simple block ciphers in the sense that they have block length equal to one symbol. The advantages of these type of ciphers consist in the fact that error propagations are not serious contrary to block ciphers where the transmission error of even a single bit will render at least one block undecipherable if not ruin the whole message. Another strength is that the encryption scheme can be changed for each letter. They are privileged especially when data must be processed one byte at a time due to memory restrictions.

A stream cipher applies simple encryption transformations according to the keystream being used. The keystream could be generated at random, or by an algorithm which generates the keystream from an initial small keystream (called a seed), or from the seed and previous ciphertext symbols. Such an algorithm is called a keystream generator.

A very simple cipher, but very important for historical reasons, is the Vernam cipher. It consists of an operation on the binary message $m_1m_2...m_t$ by a binary key stream $k_1k_2...k_t$ of the same length producing a ciphertext string $c_1c_2...c_t$ where:

$$. \qquad c_i = m_i \oplus k_i,\ 1 \leq i \leq t$$

If the key string is randomly chosen and never used again, the Vernam stream cipher is said to be one-time pad. A one-time pad stream cipher can theoretically be proven to be unbreakable, an attribute which very few (none of the most popular) modern commercial encryption schemes are privileged to have.

The direct communication line (the red telephone) between the White House's Oval Room and the Kremlin Presidential Room is rumored to be encrypted with a Vernam cipher (or at least was). Naturally the cost in generating and sharing keys long enough imply a quantity of resources which only government institutions and few others can afford.

Feistel cipher - A modern symmetric cipher

For synthetic reasons, only one representative generalized modern block cipher will be illustrated seeing that any attempt in choosing a representative quantity of real symmetrical ciphers to illlustrate would risk being incomplete given the enormous amount of symmetric ciphers available. The algorithms used are usually quite laborious and present no conceptual difficulties and, for real implementations, consultation of specialized material is advised.

The Feistel cipher was invented in the mid 1970’s an is defined in the following manner:

> Definition: a Fiestel cipher is an iterated cipher mapping a $2t$-bit plaintext (L_0, R_0), for t-bit blocks L_0 and R_0 , to a ciphertext (R_r, L_r), through an r-round process where $r \geq 1$. For $1 \leq i \leq r$, round i maps $(L_{i-1}, R_{i-1}) \rightarrow^{Ki} (L_i, R_i)$ as follows:
>
> $$L_i = R_i{-}1,\ R_i = L_i{-}1 \oplus f(R_i{-}1, K_i),$$
>
> where each subkey K_i is derived from the cipher key K.

Typically in a Feistel cipher, r ≥ 3 and often is even. The Feistel structure specifically orders the ciphertext output as (R_r, L_r) rather than (L_r, R_r), the blocks are exchanged from their usual order after the last round. Decryption is thereby achieved using the same r-round process but with sub-keys used in reverse order, K_r through K_1. The f function of the Feistel cipher may be a product cipher and f itself need not be invertible to allow inversion of the Feistel cipher.

The Feistel cipher lies at the base of one of the most famous symmetric ciphers recognized world-wide, the Data Encryption Standard (DES). The key length of DES is 56 bits long and rumor has it that this was intentionally limited to allow electronic surveillance by federal institutes in America. Today DES is no longer in use and has been replaced by an extended version with a 128 bit key and is know as Advanced Encryption Standard (AES).

Asymmetric techniques (Public-key cryptography)

One of the greatest downfalls of symmetric schemes is the need to share keys. This defect is a renowned challenge called 'the key distribution problem' and was publicly solved in the 1970’s. The necessity in sharing a key increases the difficulties involved in establishing cryptographic communications influencing mostly practical aspects as well as economic especially in a context as Internet where users might need to communicate with unfamiliar entities but nevertheless need certain assurances.

Asymmetric or public-key cryptography is a concept that was developed by Martin Hellman and Whitfield Diffie and was introduced in the mid 1970's even though the British secret services claim that the agent J .H. Ellis preceded them in the same discovery years before. Hellman and Diffie were particularly interested in resolving the key distribution problem and joined forces.

After various attempts, Hellman eventually came up with an idea that sealed the invention and is worth illustrating because summarizes the essential concept of public key cryptography. Suppose Anne wants to communicate with Bob in a confidential way so that Eve will not be able to intercept and read the message. Anne can write a ciphered message and send it via post. Even if Eve, who coincidentally works at the post-office, intercepts the message she will not be able to violate the code but neither will Bob be able to decipher the text unless Anne and Bob share a key. Now if obtaining a shared key is practically impossible (for whatever reason) this represents a seemingly insurmountable obstacle. They could phone each other, but this isn't totally secure seeing that Eve could intercept the phone call and gain knowledge of the secret key and use it to decipher the messages. Anne and Bob, after various experiments, come up with an idea which eventually solves their dilemma. Bob purchases a robust lock and sends it to Anne without locking it and with out the key. He keeps the key and all Anne has to do upon receiving the open lock is to use it to seal a metal box with the cleartext message inside and thereafter post it. If Eve gains possession of the box she will not be able to interfere seeing that even though she did witness the lock being sent from Bob to Anne, unfortunately the key was not included. When Bob eventually receives the box, he will be the only one in possession of the key and will be able to open the lock and recover the message. Note that, not even Anne, once she has locked the box is able to recuperate the message.

Public Key Infrastructures work this way: Certificate Authorities (Trusted Third Parties or TTL's) take the responsibility of keeping many of Bob's 'open locks' and deliver them to whoever requests one in order to send and encrypted message to Bob. The certificate authority supplies guaranties on the origin and authenticity of the lock but is by no means able to open closed locks. The locks are an analogy of public keys, used to cipher messages and only in possession of the private key can the message be deciphered. Anyone can gain possession of a public key and cipher a message destined to Bob, but if only Bob possesses the corresponding private key (note that he himself generated the private and corresponding public key) he will be the only entity able to recover the plain text message.

Hellman and Diffie suspected that any eventual practical implementation of their invention would revert to one-way functions. One-way functions can be defined in the following way:

> Definition: A function f from a set X to Y is called a one-way function is $f(x)$ is "easy" to compute for all x ε X but for "essentially all" elements y ε $Im(f)$ it is "computationally infeasible" to find any x ε X such that $f(x)=y$. [Menezes, van Oorschot, Vanstone]

Again, a rigorous definition of the terms "easy" and "computationally infeasible" is necessary but would detract from the simple idea being conveyed. For the purposes of this work, the intuitive meaning will suffice. A sub-set of mathematics which is rich of one-way functions is modular arithmetic's and indeed it is in this field that the first asymmetric algorithms invented originate from.

After their discovery, Hellman and Diffie then tried to find an algorithm (supposedly based on a one-way function) which practically achieved the advantages of their public-key scheme , however they were unable to resolve the new challenge and were destined to be preceded by three MIT researchers: Rivest, Shamir and Adleman . These latter invented and patented the first public scheme algorithm named RSA after the authors. Hellman and Diffie did, however, manage to invent previously a scheme which partially resolved the key distribution problem and is illustrated below even though its collocation would better suit the Key Establishment Protocol chapter. It is presented here because formally it was invented before RSA and probably helped reach the final version of the first public-key algorithm (i.e. RSA)[1].

The Diffie Hellman problem is simpler to understand than the RSA scheme and is usefulto understand the more complex RSA scheme and clearly illustrates how modular arithmetic's fit's into asymmetric cryptography. The notations are standard and common in literature. Brief explanations in the footnotes are given but for unfamiliar readers a consultation of adequate material is advised.

SUMMARY: A and B each send the other one message over an open channel.
RESULT: shared secret K known to both parties A and B.
One-time setup. An appropriate prime p and generator α of Z^*_p ($2 \leq \alpha \leq p$-2) are selected and published[2].
Protocol messages.

A → B: $\alpha^x \bmod p$ (1)
A → B: $\alpha^y \bmod p$ (2)

Protocol actions. Perform the following steps each time a shared key is required.

A chooses a random secret x, $1 \leq x \leq p$-2, and sends B message (1).
B chooses a random secret y, $1 \leq y \leq p$-2, and sends A message (2).
B receives α^x and computes the shared key as $K = (\alpha^x)^y \bmod p$.
A receives α^y and computes the shared key as $K = (\alpha^y)^x \bmod p$.

The key agreement does provide end users with a shared secret but it has various downfalls. The main downfall is the necessity of message exchange before the secret is shared and this is not suitable for applications such as e-mail where receivers are not necessarily on-line ready to accommodate the protocol therefore the key distribution problem was solved only partially. This downfall is not present in the RSA scheme below.

RSA Public-Key Scheme

Rivest, with the help of Shamir and Adleman invented an algorithm which relies on modular arithmetic's as foreseen. The RSA cryptosystem is the most widely used public-key cipher mechanism and can be used to provide both secrecy and digital signatures (see later). The RSA encryption scheme uses a public key to encrypt a message and the private key to decrypt the ciphertext. The keys are generated as following:

[1] Again the British Secret services claim that one of their mathematicians, Clifford Cocks, invented the scheme years before it's public release. Funnily enough, Clifford Cocks parted immediately from the idea of public-key encryption formalized by Ellis and discovered what in substance was to be patented as RSA whilst the Diffie-Hellman key exchange protocol was discovered successively by yet another of the secret services mathematicians, Malcolm Williamson before Diffie and Hellman

[2] Z^*_p is the multiplicative group of Z_p (i.e. all elements x of Z ∊ [0, p) such that gdc(x, p) = 1 (co primes) and α is a primitive element of Z^*_p such that $\alpha^t \equiv 1 \pmod{p}$ where t is the order of the multiplicative group (number of elements)

SUMMARY: each entity creates an RSA public key and a corresponding private key.
Each entity A should do the following:
Generate two different large random primes p and q, each roughly the same size.
Compute $n=pq$ and $\phi=(p-1)(q-1)$.
Select a random integer e, $1<e<\phi$, such that $\gcd(e, \phi)=1$.
Use the extended Euclidean algorithm[3] to compute the unique integer d, $1<d<\phi$, such that $ed\equiv 1 \pmod{\phi}$.
A's public key is (n, e); A's private key is d.

The algorithm for the message encryption and decryption is as follows:

SUMMARY: B encrypts a message for A, which A decrypts.
Encryption: B should do the following:
Obtain A's authentic public key (n, e).
Represent the message as an integer m in the interval $[0, n-1]$.
Compute $c = m^e \bmod n$.
Send the ciphertext c to A.
Decryption. To recover plaintext m from c, A should do the following:
Use the private key d to recover $m = c^d \bmod n$.

The proof that this works is omitted but can be easily found in literature.

The task faced by a passive adversary is that of recovering plaintext m from the corresponding ciphertext c, given the public information (n, e) of the intended receiver A. This is called the RSA problem (RSAP) and one possible approach which an adversary could employ to solving the RSA problem is to first factor n, and then compute ϕ and d just as A did. Once d is obtained, the adversary can decrypt any ciphertext intended for A. This can be proven to be as difficult as the factorization problem: given the public key, finding the private key is as "hard" as factorizing a very big number which formally takes $O((\log^3 n)\mathrm{lglglg}\; n)$ bit operations.

There are other possible typologies of attacks but none have yet sufficed to discourage it's current use and the RSA schemes remains today the most widely used public-key encryption and signature scheme. The RSA signature scheme is achieved in a very similar way and will be illustrated later.

ElGamal public-key encryption

The ElGamal public-key encryption scheme is similar to the Diffie-Hellman key agreement (see later) mechanism and its security is based on the intractability of also the discrete logarithm problem.

Definition: The discrete logarithm (LDP) problem is the following: given a prime p, a generator α of Z^*_p(multiplicative group of Z_p), and an element β ε Z^*_p, find the integer x, $0 \le x \le p-2$, such that:

$$\alpha^x \equiv \beta \qquad \pmod{p}.$$

Below is the description of the key generation:

[3] [Meenzes, van Oorshot and Vandstone] Algorithm 2.107

SUMMARY: each entity creates a public key and a corresponding private key.
Each entity A should do the following:

Generate a large random prime p and a generator α of Z^{*}_{p}.
Select a random integer a, $1 \le a \le p$-2, and compute α^{a} mod p.
A's public key is (p, α, α^{a}); A's private key is a.

Whereas the encryption scheme is the following:

SUMMARY: B encrypts a message m for A, which A decrypts.
Encryption: B should do the following:

Obtain A's authentic public key (p, α, α^{a}).
Represent the message as an integer m in the range $\{0, 1,\ldots,p\text{-}1\}$.
Select a random integer k, $1 \le k \le p$-2.
Compute $\gamma = \alpha^{k}$ mod p and $\delta = m\bullet (\alpha^{a})^{k}$ mod p.
Send the ciphertext c = (γ, δ) to A.

Decryption. To recover plaintext m from c, A should do the following:

Use the private key a to compute $\gamma^{p\text{-}1\text{-}a}$ mod p (note: $\gamma^{p\text{-}1\text{-}a} = \gamma^{-a} = \alpha^{-ak}$).
Recover m by computing $(\gamma^{-a}) \bullet \delta$ mod p.

The decryption of the above algorithm allows recovery of original plaintext because

$$\gamma^{-a}\bullet\delta = \alpha-^{ak}\bullet m\bullet\alpha^{ak} = m \qquad (\bmod\ p).$$

Symmetric-key vs. public-key cryptography

Symmetric-key and public-key encryption schemes have various advantages and disadvantages, some of which are common to both. This section highlights a number of these and summarizes features pointed out in previous sections.

Advantages of symmetric-key cryptography

1. Symmetric-key ciphers can be designed to have high rates of data throughput. Some hardware implementations achieve encrypt rates of hundreds of megabytes per second, while software implementations may attain throughput rates in the megabytes per second range.
2. Keys for symmetric-key ciphers are relatively short.
3. Symmetric-key ciphers can be employed as primitives to construct various cryptographic mechanisms including pseudorandom number generators, hash functions, and computationally efficient digital signature schemes, to name just a few.
4. Symmetric-key ciphers can be composed to produce stronger ciphers. Simple transformations which are easy to analyze, but on their own weak, can be used to construct strong product ciphers.
5. Symmetric-key encryption is perceived to have an extensive history, although it must be acknowledged that, notwithstanding the invention of rotor machines earlier, much of the knowledge in this area has been acquired subsequent to the

invention of the digital computer, and, in particular, the design of the Data Encryption Standard in the early 1970s.

Disadvantages of symmetric-key cryptography

1. In a two-party communication, the key must remain secret at both ends.
2. In a large network, there are many key pairs to be managed. Consequently, effective key management requires the use of an unconditionally trusted TTP.
3. In a two-party communication between entities A and B, sound cryptographic practice dictates that the key be changed frequently, and perhaps for each communication session.
4. Digital signature mechanisms arising from symmetric-key encryption typically re-quire either large keys for the public verification function or the use of a TTP.

Advantages of public-key cryptography

1. Only the private key must be kept secret (authenticity of public keys must, however, be guaranteed).
2. The administration of keys on a network requires the presence of only a functionally trusted TTP as opposed to an unconditionally trusted TTP. Depending on the mode of usage, the TTP might only be required in an "off-line" manner, as opposed to in real time.
3. Depending on the mode of usage, a private key/public key pair may remain unchanged for considerable periods of time, e.g., many sessions (even several years).
4. Many public-key schemes yield relatively efficient digital signature mechanisms. The key used to describe the public verification function is typically much smaller than for the symmetric-key counterpart.
5. In a large network, the number of keys necessary may be considerably smaller than in the symmetric-key scenario.

Disadvantages of public-key encryption

1. Throughput rates for the most popular public-key encryption methods are several orders of magnitude slower than the best known symmetric-key schemes.
2. Key sizes are typically much larger than those required for symmetric-key encryption, and the size of public-key signatures is larger than that of tags providing data origin authentication from symmetric-key techniques.
3. No public-key scheme has been proven to be secure (the same can be said for block ciphers). The most effective public-key encryption schemes found to date have their security based on the presumed difficulty of a small set of number-theoretic problems.

4. Public-key cryptography does not have as extensive a history as symmetric-key encryption, being discovered only in the mid 1970s.

Further uses of cryptography

Up to now symmetric and asymmetric techniques have been presented and their immediate and most intuitive use is the concealing of messages in order to obtain confidentiality. There are however other advantages which can be achieved through the use of cryptography and they have been mentioned above. This section will briefly present these other uses of cryptography excluding confidentiality. Some of these benefits were mostly invented after the invention of asymmetric cryptography which not only permitted to resolve the key distribution problem but enriched the utility of cryptography.

Digital Signatures

These techniques are designed to provide the digital counterpart to handwritten signatures and can be achieved using cryptography. In substance a digital signature of a message is a number dependant on some secret known only to the signer and on the content of the message being signed. Signatures must be verifiable without requiring access to the signers secret information. The idea is similar to asymmetric cryptography but it is complimentary. Whereas in public-key encryption schemes, the public key is used to encrypt the message and the private key to decrypt, digital signatures are obtained generating a number using the **private key** and verified with the **public key**.

Digital signatures have many applications in information security, including authentication, data integrity, and non-repudiation. One of the most significant applications of digital signatures is the certification of public keys in large networks. Certification is a means for a trusted third party (TTP) to bind the identity of a user to a public key, so that at some later time, other entities can authenticate a public key without assistance from a trusted third party. It now becomes clear how asymmetric cryptography surmounts the key distribution problem. A user, destined to receive an encrypted message, can send his certificate containing his public key issued by a TTL or Certificate Authority. The receiver, who desires to encrypt and send the message, can authenticate that the certificate was issued by the common Certificate Authority (CA) using the CA public key thus acquiring the guaranty that the public key received belongs to the intended recipient (aor at least the CA's guaranty).

The first method discovered was the RSA signature scheme (an RSA public key encryption compliment scheme) and is still widely used in Internet together with another digital scheme called *Digital Signature Algorithm* (DSA) proposed by the National Institute of Standards and Technology (NIST) and is the first digital signature scheme to be recognized by any government (1991).

Below is an illustration of the RSA digital scheme leaving DSA for further consultancy of dedicated material. The RSA signature scheme key generation is the same as for RSA public key encryption which represents an advantage allowing the use of the same private key to both decrypt and sign messages whilst the public key can be used to encrypt and verify messages. The public key consists of the pair (n, e) whilst the private key is d (see above RSA public key encryption key generation) and RSA signing scheme is as follows:

SUMMARY: entity A signs a message $m \in M$. Any entity B can verify A's signature and recover the message m from the signature.

Signature generation: Entity A should do the following:

Compute $m` = \mathrm{R}(m)$[4], an integer in the range $[0; n-1]$.

Compute $s = m^{`d} \bmod n$.

A's signature for m is s.

Verification: To verify A's signature s and recover the message m, B should:

Obtain A's authentic public key (n, e).

Compute $m` = s^e \bmod n$.

Verify that $m` \in M_R$; if not, reject the signature.

Recover $m = \mathrm{R}^{-1}(m`)$.

If s is a signature for a message m, then $s \equiv m^{`d} \bmod n$ where $m` = \mathrm{R}(m)$.Since $ed \equiv 1 \pmod{\phi}$, $s^e \equiv m^{`ed} \equiv m` \pmod{n}$. Finally, $\mathrm{R}^{-1}(m`) = \mathrm{R}^{-1}(\mathrm{R}(m)) = m$.

Data Integrity and message authentication

Assurances are typically required both that data actually came from its reputed source (data origin authentication), and that its state is unaltered (data integrity). These issues can-not be separated - data which has been altered effectively has a new source; and if a source cannot be determined, then the question of alteration cannot be settled (without reference to a source). Integrity mechanisms thus implicitly provide data origin authentication, and vice versa. Cryptography helps obtain these assurances and in the field a wide use of hash functions is implemented amongst other cryptographic functions

Definition: A hash function is a function h which has, as a minimum, the following two properties:

1. *compression* — h maps an input x of arbitrary finite bit length, to an output $h(x)$ of fixed bit length n.
2. *ease of computation* —given h and an input x, $h(x)$ is easy to compute.

Hash functions are used for data integrity in conjunction with digital signature schemes where for several reasons a message is typically hashed first and then the hash-value, as a representative of the message, is signed in place of the original message. A distinct class of hash functions, called message authentication codes (MACs), allows message authentication by symmetric techniques. MAC algorithms may be viewed as hash functions which take two functionally distinct inputs, a message and a secret key, and produce a fixed-size (say n-bit) output, with the design intent that it be infeasible in practice to produce the same output without knowledge of the key. MACs can be used to provide data integrity and symmetric data origin authentication, as well as identification in symmetric-key schemes.

[4] R(x) is a redundancy function chosen and publicly known
$R: M \rightarrow Z_n$ where M is the message space

A typical usage of (unkeyed) hash functions for data integrity is as follows. The hash-value corresponding to a particular message *x* is computed at time T_1 . The integrity of this hash-value (but not the message itself) is protected in some manner. At a subsequent time T_2 , the following test is carried out to determine whether the message has been altered, i.e., whether a message *x*` is the same as the original message. The hash-value of *x*' is computed and compared to the protected hash-value; if they are equal, one accepts that the inputs are also equal, and thus that the message has not been altered. The problem of preserving the integrity of a potentially large message is thus reduced to that of a small fixed-size hash-value. Since the existence of collisions is guaranteed in many-to-one mappings, the unique association between inputs and hash-values can, at best, be in the computational sense. A hash-value should be uniquely identifiable with a single input in practice, and collisions should be computationally difficult to find (essentially never occurring in practice).

Key establishment protocols

Here the act of sharing a key is presented along with related cryptographic techniques which provide shared secrets between two parties. There are, however, many protocols which allow for more than two parties, belonging to group, to share secrets in a mutually exclusive way. In this document only protocols which involve two parties and a Trusted Third Party used in Internet are presented. The two most used protocols for sharing key over Internet are Kerberos (based on symmetric techniques) and the X.509 (based on asymmetric techniques) will be considered in detail.

Kerberos authentication protocol

Kerberos is key transport protocol based on symmetric encryption. This technique requires a prior key sharing on behalf of two parties *A*, *B* with a Trusted Third Party *T*. Let's call the keys K_{AT} and K_{BT}. Here a session key *k* is chosen by *T* and sent to *A* upon request. *A* then sends the session key *k* to *B* and starts the communication.

SUMMARY: A interacts with trusted server T and party B.
RESULT: entity authentication of A to B with key establishment. The notations used are found in footnote
Notations:

E is a symmetric encryption algorithm.
NA is a nonce chosen by A; TA is a timestamp from A's local clock.
k is the session-key chosen by T ,to be shared by A and B.
L indicates a validity period (called the "lifetime").One-time setup. *A* and *T* share a key K_{AT}; similarly, *B* and *T* share K_{BT}.

Protocol messages.

$A \rightarrow T : A; B; N_A$ (1)
$A \leftarrow T : ticket_B$[5]; $E_{KAT}\,(k, N_A, L, B)$ (2)
$A \rightarrow B : ticket_B$; *authenticator*[6] (3)
$A \leftarrow B : E_k(T_A, B^*_{subkey})$ (4)

With this protocol a big issue is the fact that the Trusted Third Party generated the key and is in it's possession. In a way this avoids that only one of the entities absolves this task and thus obtaining a privileged position but this can easily be overcome. The next protocol allows for two parties to share a key without the need of the TTL to generate the secret key.

X.509 authentication protocol

This is a key transport mechanism based on asymmetric techniques. The main difference with the protocol above is that here the key is chosen by the end users i.e. parties A and B. The Trusted Third Party participates passively and will never be in possession of the key. The TTL, in this case acts as a Certificate Authority providing the means forverifying authenticity of A's and B's certificates.

SUMMARY: A sends B one message, and B responds with one message.
RESULT: mutual entity authentication and key transport with key authentication.
Notations.

$P_X(y)$ denotes the result of applying *X*'s encryption public key to data *y*.
$S_X(y)$ denotes the result of applying *X*'s signature private key to *y*.
r_A, r_B are never re-used numbers (to detect replay and impersonation).
$cert_X$ is a certificate binding party *X* to a public key suitable for both encryption and signature verification.

System setup.

Each party has its public key pair for signatures and encryption.
A must acquire (and authenticate) the encryption public key of *B* a priori.

Protocol messages. (An asterisk denotes items are optional.)

$A \rightarrow B : cert_A, D_A, S_A(D_A)$[7] (1)
$A \leftarrow B : cert_B, D_B, S_B(D_B)$[8] (2)

[5] $ticket_B = E_{KBT}\,(k; A; L)$ and

[6] $authenticator = E_k(A, T_A; A^*_{subkey})$

[7] $D_A = (t_A, r_A, B, data_1^*, P_B(k_1)^*)$

[8] $D_B = (t_B, r_B, A, r_A, data_2^*, P_A(k_2)^*)$

Conclusions

In this document a brief overview of cryptography was presented aiming to illustrate the goals one can achieve with modern cryptography. The context was applied to the Internet scenario and the algorithms briefly introduced were chosen to represent real cryptographic applications used in Internet. A short history of cryptography was included with the intent of rendering the importance of the role played by cryptography in today's society quoting real cryptographic related episodes in history. The future of cryptography depends on the technologies available and future inventions. Cryptography, as is today, is from a mathematical point of view not perfect. For example a lot of the asymmetric algorithms base their level of security on the intractability of well-known mathematical problems which are only "believed" to be unsolvable but have never been formally proven such as the factorization problem or discrete logarithm problem.

Great relevance was given to the key distribution problem and how asymmetric cryptography helped resolve this even though it is not the only solution. A detailed description of simple ciphers was presented to then use the concepts developed to illustrate the Feistel-cipher, bases of some of the most frequent ciphers used in Internet today. Other uses of cryptography such as electronic digital signatures and data integrity and authenticity checks were then explained.

Cryptography is a vast science and new algorithms are continuously being studied and developed. This science merits more attention during practical implementations because as history teaches us, cryptography presents two faces. It has happened that clamorous flaws have been discovered in the actual algorithms, already implemented in security applications, causing great nuisance and at times even damage. The use cryptography provides us with the tools needed to keep at pace with the ever changing world and slowly our daily lives will rely more and more on the benefits offered by a cryptography not to mention a 'secure' Internet.

Bibliography and References

[1] “Handbook of Applied Cryptography”, by A. Meenzes, P. van Oorschot and S. Vanstone.

[2] “The Code Book”, by Simon Singh.

[3] “Applied Cryptography”, by Bruce Schneier.

[4] “Secrets and Lies”, by Bruce Schneier.

[5] “Guide to Cryptography”, by Randall K.Nichols

[6] “Turings Treatise on Enigma”, retyped by Ralph Erskine, Philip Marks and Frode Weierud

[7] RSA Security Coorporation, http://www.rsasecurity.com/

[8] “Codes and Ciphers in the Second World War”, created by Tony Sale http://www.codesandciphers.org.uk/

[9] “A Short History of Cryptography”, by Fred Cohen.

[10] “The Code Breakers”, by David Kahn.