



Cryptography: Introduction



Luca Veltri

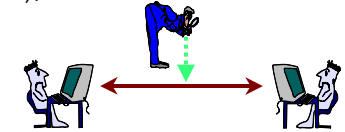
(mail.to: luca.veltri@unipr.it)

Course of Network Security, Spring 2013

<http://www.tlc.unipr.it/veltri>

Cryptography

- Cryptography (Greek: krupto+grafη: hidden/secret+writing) is the study of mathematical techniques related to information security in the presence of third parties (adversaries), with the following objectives:



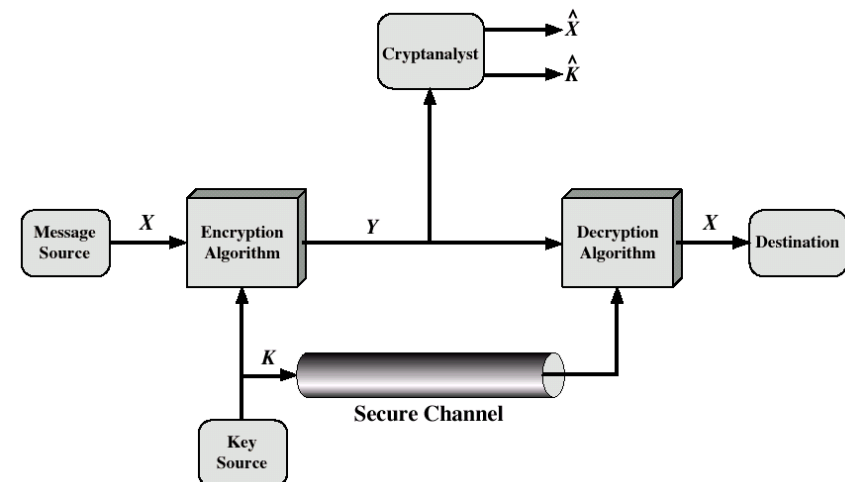
- **Confidentiality**
 - ensuring information is accessible only by authorized persons
- **Data integrity**
 - ensuring information is has not been altered by unauthorized or unknown means
- **Authentication**
 - verification of the identity of an entity
- **Non-repudiation**
 - preventing the denial of previous commitments or actions

- The most widely used tool for securing information and service
 - **it is one tool (not the only)**

Cryptography and Cryptanalysis

- Cryptography
 - **study of techniques for secure communication in the presence of third parties**
- Cryptanalysis
 - **the art and science of analyzing information systems in order to study the hidden aspects of the systems**
 - **cryptanalysis is used to breach cryptographic security systems and gain access to the contents of encrypted messages, even if the cryptographic key is unknown**
- Cryptology
 - **the scientific study of cryptography and cryptanalysis**

Model of classic cryptographic system



Cryptography: Basic Terminology



- **plaintext** - the original message
- **ciphertext** - the coded message
- **cipher** - algorithm for transforming plaintext to ciphertext
- **key** - info used in cipher known only to sender/receiver
- **encipher (encrypt)** - converting plaintext to ciphertext
- **decipher (decrypt)** - recovering ciphertext from plaintext

5

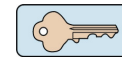
Very simple (and insecure) example of text-based cryptographic algorithm (shift cipher)

Algorithm: $X \leftarrow M+K \pmod{26}$ $K \in \{0,1,\dots,25\}$

Plaintext: C A S A

Ciphertext: R P H P

Key: $K = 15$



Caesar cipher:

key $K=3$

6

Cryptographic algorithm

- Mathematical functions used for encrypting (and decrypting) input data
- Can be characterized by:
 - **number of used keys**
 - single-key or secret / two-key or private-public
 - **mode for processing input data**
 - block / stream
 - **type of operation used for encryption/decryption**
 - substitution / transposition / product
 - **other**
 - speed, HW/SW implementation complexity, etc.

7

Security of a cryptographic algorithm

- The security of a cipher might rest in the secrecy of its restricted algorithm
 - $c = E(m)$
 - $m = D(c)$

however:

 - **whenever a users leaves a group, the algorithm must change**
 - **could be scrutinized by people smarter than you**
- Modern cryptography relies on keys, a selected value from a large set (a keyspace), e.g., a 1024-bit number $\rightarrow 2^{1024}$ values!
 - $c = E(k,m) = E_k(m)$
 - $m = D(k,c) = D_k(c)$
 - **change of authorized participants requires only a change in key**
 - **“Security should be based on secrecy of the key, not the details of the algorithm”**
 - Jean Guillaume Hubert Victor Francois Alexandre Auguste Kerckhoffs von Nieuwenhof, “La Criptographie Militaire”, 1883

8

Cryptography Attacks

- Cryptographic analysis (cryptoanalysis)
 - **Si basa sulla natura dell'algoritmo e sfrutta qualche conoscenza delle caratteristiche generali del testo in chiaro e/o qualche esempio di coppie testo in chiaro/testo cifrato**
 - non richiede necessariamente di scoprire la chiave
- Brute-force search
 - **si tenta ogni possibile decifrazione (ad esempio provando tutte le possibili chiavi) su un frammento di testo in chiaro**
 - **in media per avere successo occorre provare la metà delle chiavi possibili**
 - **deve essere possibile riconoscere quando si è trovato il testo in chiaro corretto**
- The loss of a key (also without cryptoanalysis or brute-force attack) is called a "compromise"

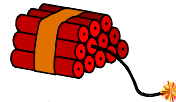
9

Brute force attack

- Method of defeating a cryptographic scheme by trying a large number of possibilities
 - **for example, for symmetric-key ciphers it typically means a brute-force search of the key space**
 - testing all possible keys in order to recover the plaintext used to produce a particular ciphertext
- In most schemes, the theoretical possibility of a brute force attack is recognized, but it is set up in such a way that it would be computationally infeasible to carry out
- The attacker has to determine when he succeeded
 - **By obfuscating the data to be encoded, brute force attacks are made less effective as it is more difficult to determine when one has succeeded in breaking the code**

10

Cryptographic break



- A cryptographic "break" is anything faster than an exhaustive search (brute force attack)
 - **example, an attack against a 128-bit-key cipher requiring 'only' 2^{120} operations (compared to 2^{128} possible keys) would be considered a break even though it would be, at present, quite infeasible**

11

Types of Cryptanalytic Attacks

- There are four basic cryptoanalytic classes of attacks:
 - **Ciphertext-only attack**
 - **Known-plaintext attack**
 - **Chosen-plaintext attack**
 - **Chosen-ciphertext attack**

12

Ciphertext only - Attack

- The bad guy has seen (and presumably stored) some ciphertext that can be analyzed
 - **One possible strategy to figure out the plaintext is to try all keys**
 - it is essential that he/she is able to recognize when he/she has succeeded (often called *recognizable plaintext* attack)
 - for example in case of normal text or known document formats (e.g. PostScript, etc.) with recognizable patterns
 - oppure considerando le parole più comuni (attacco a parole probabili)
 - **it is necessary to have enough ciphertext**
- E' l'attacco più difficile da realizzare

13

Known plaintext - Attack

- The bad guy knows a <plaintext, ciphertext> pair
- From that pairs, the attacker can try to figure out the mapping of some fraction of the text
- How it is possible to obtain the plaintext?
 - **the secret data does not remain secret forever (e.g. the name of an attacked city)**
- Some cryptographic schemes might be good enough to be secure against *ciphertext only* attacks but not against to *known plaintext* attacks
 - **in these cases, it is important to minimize the possibility for a bad guy to obtain <plaintext, ciphertext> pairs**

14

Chosen plaintext (or ciphertext) - Attack

- The bad guy can choose any plaintext and get the corresponding ciphertext from the system (or the contrary)
 - **e.g. there is a telegraph service that encrypt and transmit messages; the bad guy can ask the telegraph company to transmit any plaintext he/she wants**
- Some cryptographic schemes might be good enough to be secure against *ciphertext only* attacks and *known plaintext* attacks but not against to *chosen plaintext* attacks

15

Side channel attack

- Any attack based on information gained from the physical implementation of a cryptosystem, rather than theoretical weaknesses in the algorithms (compare cryptanalysis)
 - **For example, timing information, power consumption, electromagnetic leaks or even sound can provide an extra source of information which can be exploited to break the system**
- General classes of side channel attack include:
 - **Timing attack** — attacks based on measuring how much time various computations take to perform
 - **Power monitoring attack** — attacks which make use of varying power consumption by the hardware during computation
 - **TEMPEST (aka van Eck or radiation monitoring) attack** — attacks based on leaked electromagnetic radiation which can directly provide plaintexts and other information
 - **Acoustic cryptanalysis** — attacks which exploit sound produced during a computation (rather like power analysis)

16

Side channel attack (cont.)

- In all cases, that physical effects caused by the operation of a cryptosystem can provide useful extra information about secrets in the system, for example, the cryptographic key, partial state information, full or partial plaintexts and so forth
- Many side-channel attacks require considerable technical knowledge of the internal operation of the system on which the cryptography is implemented

17

Computational Difficulty

- Cryptographic algorithms should be reasonably efficient to compute for good guys (who know the keys)
- Cryptographic algorithms are often not impossible to attack without the key: a bad guy can simply try all possible keys until one works
- The security depends on how much work is necessary to succeed on attacking it
 - It is the **complexity** of launching the attack that secures us
- Attack complexities:
 - **data complexity**: a large number of expected inputs (e.g., ciphertext)
 - **processing complexity**: a large number of operations required
 - **storage complexity**: a large amount of storage units required
- Often a scheme can be made more secure by making the key longer

18

Computational and Unconditional Security

- **computational security**
 - given limited computing resources (e.g. time needed for calculations is greater than age of universe, or the cost required for the attack is not affordable), the cipher cannot be broken
- **unconditional security**
 - no matter how much computer power is available, the cipher cannot be broken since the ciphertext provides insufficient information to uniquely determine the corresponding plaintext
 - e.g. OTP (One Time Pad) cipher

19

Brute Force Search

- Remarks:
 - often possible, by simply trying every key
 - most basic attack, proportional to key size
 - assume either know / recognise plaintext

Key Size (bits)	Number of Alternative Keys	Time required at 1 encryption/ μ s	Time required at 10^6 encryptions/ μ s
32	$2^{32} = 4.3 \times 10^9$	$2^{31} \mu$ s = 35.8 minutes	2.15 milliseconds
56	$2^{56} = 7.2 \times 10^{16}$	$2^{55} \mu$ s = 1142 years	10.01 hours
128	$2^{128} = 3.4 \times 10^{38}$	$2^{127} \mu$ s = 5.4×10^{24} years	5.4×10^{18} years
168	$2^{168} = 3.7 \times 10^{50}$	$2^{167} \mu$ s = 5.9×10^{36} years	5.9×10^{30} years
26 characters (permutation)	$26! = 4 \times 10^{26}$	$2 \times 10^{26} \mu$ s = 6.4×10^{12} years	6.4×10^6 years

20

Cryptanalysis: Example (Language Redundancy)

- Example of cryptanalysis:
 - **cleartext: normal text**
 - **cryptography algorithm: monoalphabetic substitution**
- key concept:
 - **human languages are redundant: do not change relative letter frequencies**
 - **in English 'e' is by far the most common letter**
 - then T,R,N,I,O,A,S
 - other letters are fairly rare (e.g. Z,J,K,Q,X)
- cryptanalysis:
 - **based on language redundancy**
 - **discovered by Arabian scientists in 9th century**
 - **calculate letter frequencies for ciphertext**
 - **compare counts/plots against known values**
 - **have tables of single, double & triple letter frequencies**

21

Cryptanalysis: Example

- given ciphertext:


```
UZQSOVUOHXMKOPVGPQZPEVSGZWSZOPFPESXUDBMETSXAIZ
VUEP HZHMDZSHZOWSFPAPPDTSVPQUZWMXUZUHSX
EPYEPQDPDZSZUFPOMBZWPFPUPZHMDJUDTMOHMQ
```
- count relative letter frequencies
- guess P & Z are e and t
- guess ZW is th and hence ZWP is the
- proceeding with trial and error finally get:


```
it was disclosed yesterday that several informal but
direct contacts have been made with political
representatives of the viet cong in moscow
```

22

Fundamental principle of cryptography

- Fundamental principle of cryptography:

“If lots of people have failed to solve a problem, then it probably won’t be solved (soon)”

 - **cryptographers invent clever secret codes**
 - **cryptoanalysts attempt to break these codes**
 - **the two disciplines help each other!**
- Second fundamental principle of cryptography:

“Often breaking a cryptographic scheme is not the the only way of getting what you want”

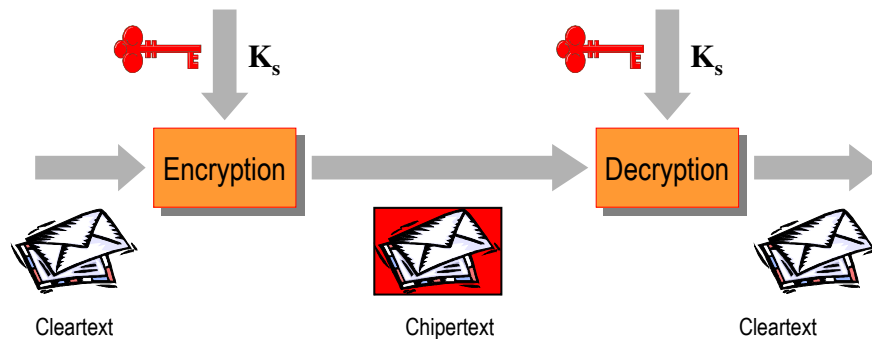
23

Different cryptographic algorithms

- Secret key cryptography (symmetric)
 - **the two communication parties share a common secret (key)**
- Public key cryptography (private/public key, or asymmetric)
 - **two different keys are used for two opposite functions (e.g. encryption and decryption)**
 - **one key can be publicly available (public key); the other is maintained secret for the owner (private key)**
- Hash algorithm (message digest/one way transformation)
 - **one-way transformation that maps a variable length message to a fixed length bit string**

24

Secret key cryptography (symmetric)



25

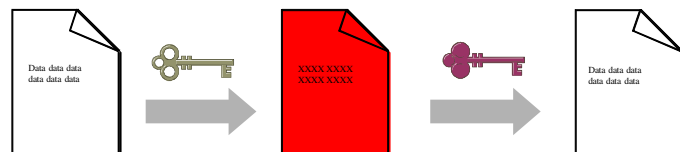
Characteristics

- Richiede una fase iniziale in cui ciascuna coppia di interlocutori si scambia la secret key in maniera sicura
- Il numero delle chiavi per realizzare una comunicazione reciproca tra N utenti (dispositivi) è pari a $N \times (N-1) / 2$ (se le chiavi rimangono sempre le stesse)
- Viene generalmente utilizzato per proteggere, mediante codifica, informazioni (file) in un repository locale o trasmesse
- La robustezza dell'algoritmo è normalmente misurata dalla lunghezza delle chiavi: 40 bit (debole), 128 bit (forte)
- Algoritmi più diffusi: DES, 3DES, RC2, RC4, IDEA, AES

26

Public key cryptography (asymmetric)

- Spesso detta crittografia asimmetrica
- La chiave utilizzata per cifrare (chiave di cifratura, K_e) è **diversa** dalla chiave utilizzata per decifrare (chiave di decifratura, K_d)
- Le chiavi si usano in coppie, di cui una detta privata (segreta) e l'altra detta pubblica (disponibile a tutti)
- Un messaggio, cifrato con la chiave privata, può essere decifrato solo con la chiave pubblica e viceversa
- Ciascuna coppia di chiavi è caratterizzata da proprietà peculiari



27

Characteristics

- Può richiedere una fase iniziale in cui gli interlocutori si scambiano le rispettive chiavi pubbliche
- Il numero delle chiavi è proporzionale a N per la comunicazione reciproca tra N utenti (dispositivi)
- Viene generalmente utilizzato per distribuire chiavi simmetriche in un ambiente distribuito
- Algoritmi più diffusi: RSA, Diffie-Hellman, DSA
- La robustezza del sistema dipende anche dal sistema di certificazione delle chiavi di cifratura (se esiste..)
 - L'impiego della crittografia a chiavi asimmetriche può comportare l'implementazione di una PKI (Public Key Infrastructure)

28

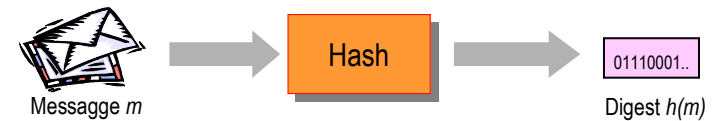
Symmetric vs. asymmetric encryption

- Entrambi gli algoritmi, simmetrici o asimmetrici, consentono di cifrare un messaggio; gli algoritmi simmetrici sono preferiti nei casi in cui sia necessaria una certa velocità di esecuzione della cifratura
- Un buon sistema crittografico usa al meglio i due tipi di algoritmo
 - **ad esempio:**
 - uso di algoritmo asimmetrico per lo scambio sicuro di chiavi e per l'autenticazione
 - uso di algoritmo simmetrico per lo scambio dei dati

29

Hash Algorithms

- Also known as message digests or one way transformations
- Hash function is a mathematical transformation m that takes a message of arbitrary length and computes a fixed-length (short) number/string $h(m)$
- Properties:
 - **for any message m , it relatively easy to compute $h(m)$**
 - **given $h(m)$ there is no way to find m in a way easier than computing all possibilities**
 - **it is computationally infeasible to find two values that hash to the same thing**



- Examples of hash algorithms: MD2, MD5, SHA

30

Cryptography is not useful for

- Protezione di sistema:
 - **Accesso non autorizzato alle risorse HW e SW**
 - **Esecuzione di programmi malevoli (virus, etc.)**
 - ...
- Protezione dei servizi:
 - **Sfruttamento di banchi nei programmi o protocolli**
 - **Attacchi mirati a rendere indisponibili dei servizi**
 - **Utilizzo illecito di servizi ed informazioni**
 - **Alterazione dei contenuti di DB o altri dispositivi di archiviazione**
 - ...

31