UNIVERSITÀ DEGLI STUDI
DI GENOVA

University of Studies of Genoa

Computer Security

Lorenzo Vaccarecci

2024-2025

Contents

1 Introduction **2**
- 1.1 Information Security . . . 2
  - 1.1.1 Key Concepts . . . 2
- 1.2 Protection countermeasures . . . 2
- 1.3 Security Properties . . . 3
  - 1.3.1 Confidentiality . . . 3
  - 1.3.2 Integrity . . . 3
  - 1.3.3 Authenticity . . . 3
  - 1.3.4 Availability . . . 3
  - 1.3.5 Accountability . . . 4
- 1.4 Implementing a security solution . . . 4

2 Introduction to Cryptography **5**
- 2.1 General schema . . . 5
- 2.2 Classification of security . . . 6
- 2.3 Cryptoanalysis . . . 6
  - 2.3.1 Brute-force attack . . . 6
  - 2.3.2 Cryptanalytic attack . . . 6
- 2.4 Mathematical formalization . . . 6
- 2.5 Block ciphers, stream ciphers and codes . . . 7
- 2.6 Simple Subsitution ciphers . . . 7
  - 2.6.1 Affine ciphers . . . 7
  - 2.6.2 Homophonic substitution ciphers . . . 7
  - 2.6.3 Polyalphabetic substitution ciphers . . . 7
  - 2.6.4 One-time pads (Vernam cipher) . . . 7
- 2.7 Transposition ciphers . . . 8
- 2.8 Composite ciphers . . . 8

Chapter 1

Introduction

Computer security deals with the preventon and detection of unauthorized actions by users of a computer system.

- **Authorization** is central to definition
- Sensible only relative to a security policy, stating who (or what) may perform which actions

1.1 Information Security

Is even more general. It deals with information independent of computer security.

1.1.1 Key Concepts

- Security concerns the protection of assets from threats
- Owners value their assets and want to protect them
- Threat agents also value assets, and seek to abuse them
- Owners analyse threats to determine which ones apply; these are the risks that can be costed. This helps the selection of countermeasures, which reduce the vulnerabilities (may remain leaving some residual risk, owners seek to minimise that risk)

$$\text{Risk}_E = \text{Pr}_E \cdot \text{I}_E$$

where Pr_E is the probability of an event E and I_E is the impact of the event E.

1.2 Protection countermeasures

- **Prevention**: try to prevent security breaches by system design and employing appropriate security technologies as defences
- **Detection**: in the event of a security breach, we try to ensure that it will be detected. Logging and MACs (file hashes to detect alteration) are primary methods of detection, although intrusion detection systems which activelyy watch for intruders are more common
- **Response**: in the event of a security breach, we must respond or recover the assets. Responses range from restoring backups through to informing appopriate concerned parties or law-enforcement agencies

1.3 Security Properties

1.3.1 Confidentiality

Information is not learned by unauthorized principals

Confidentiality is sometimes characterised as the unauthorized reading of data, when considering **access control** measures. But in general we are concerned with unauthorized learning of information, which is more subtle to contend with. Confidentiality presumes a security policy saying who or what can access our data. The security policy is used for access control.

- **Privacy**: confidentiality for individuals
- **Secrecy**: confidentiality for organizations
- **Anonymity**: keeping one's identity private

1.3.2 Integrity

Data has not been maliciously altered

Intergrity has more general meanings elsewhere, but in computer security we are concerned with preventing the possibly malicious alteration of data, by someone who is not authorized to do so. Integrity in this sense can be characterised as the unauthorized writing of data. Again, this presumes a security policy saying who or what is allowed to alter the data.

1.3.3 Authenticity

Data or services available only to authorized identities

Authentication is th processo of verification of identity of a person or system. Some form of authentication is a pre-requisite if we wish to allow access to services or data to some people but deny access to others, using an access control system. Methods for authentication are often characterised as:

- something you have
- something you know
- something you are

Also, where you are may be implicity or explicity checked. Several methods can be combined for extra security.

1.3.4 Availability

Data or services can be accessed in a reliable and timely way

Threats to availability cover many kinds of external environmental events as well as accidental or malicious attackks in software. In computer security we're concerned with protecting against the second kind of threat, rather than providing more general forms of fault-tolerance or dependability assurance. Ensuring availability means preventing **denial od service** (DoS) attacks, insofar as this is possible. It's possible to fix attacks on faulty protocols, but attacks exhausting available resources are harder, since it can be tricky to distinguish between an attack and a legitimate use of service.

1.3.5 Accountability

Actions are recorded and can be traced to the party responsible

If prevention methods and access controls fail, we may fall back to detection: keeping a secure audit trail is important so that actions affecting security can be traced back to the responsible party. A stronger form of accountability is non-repudation, when a party cannot later deny some action. Creating an audit trail with machine logs is a tricky problem: if a system is compromised, the logs may also be tampered with. Ways around that problem are to send log messages to an append-only file, a separate server, or even a physically isolated printer.

1.4 Implementing a security solution

- A **security analysis** surveys the threats which pose risks to assets, and then proposes policy and solutions at an appropriate cost.
- A **threat model** documents the possible threats to a system, imagining all the vulnerabilities which might be exploited
- A **riks assessment** studies the likelihood of each threat in the system environment and assigns a cost value, to find the risks
- A **security policy** addresses the threats, and describes a coherent set of countermeasures

The costs of countermeasures is compared against the risks, and juggles to make a sensible trade-off. This allows a security solution to be designed, deploying appropriate technologies at an appropriate cost. Partly this is a budgeting exercise; but it's also important to spend effort in the right place.

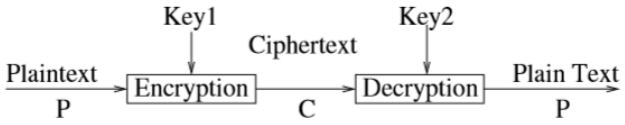
Chapter 2

Introduction to Cryptography

How do we turn untrustworthy channels into trustworthy ones?

- **Confidentiality**
- **Integrity**
- **Authenticity**

2.1 General schema



Where $E_{K_1}(P) = C, D_{K_2}(C) = P$

- **Security depends on secrecy of the key, not of the algorithm**
- **Symmetric** algorithms: $K_1 = K_2$ or are easily derived from each other
- **Asymmetric** or **public key** algorithms: different keys which cannot be derived from each other and a **public key** can be published without compromising the **private key**

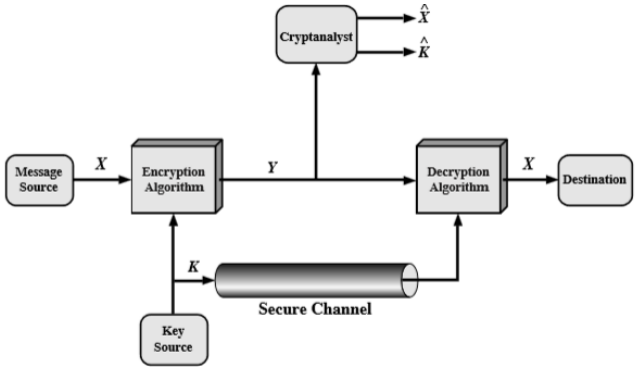


Figure 2.1: Model of Symmetric Cryptosystem

2.2 Classification of security

- **Unconditional Security**: system is secure even if adversary has unbounded computing power since the ciphertext provides insufficient information to uniquely determine the corresponding plaintext. Security measured using information theory
- **Conditional Security**: system can be broken in principle, but this requires more computing power than a realistic adversary would have. Security measured using complexity theory

2.3 Cryptoanalysis

Science of recovering the plaintext from ciphertext without the key

But typical objective is to recover key not just message.

2.3.1 Brute-force attack

- Always possible: try all possible keys (2^{bit} number of keys)
- Its cost (eavily) depends on the key size
- It assumes that plaintext is known or recognisable

2.3.2 Cryptanalytic attack

- **Ciphertext only**: only the ciphertext is known and the attacker tries to deduce the key
- **Known plaintext**: ciphertext only + plaintext
- **Chosen plaintext**: same as above but attacker can choose plaintext
- **Adaptive chosen plaintext**: cryptanalyst can not only choose plaintext, but he can modify the plaintext based on encryption results
- **Chosen ciphertext**: cryptanalyst can choose different ciphertexts to be decrypted and gets access to the decrypted plaintext

2.4 Mathematical formalization

- $\mathcal{A}$: the alphabet
- $\mathcal{M} \subseteq \mathcal{A}$: the message space, $M \in \mathcal{M}$ is a plaintext
- $\mathcal{C}$: the ciphertext space, whose alphabet may differ from $\mathcal{M}$
- $\mathcal{K}$: the key space
- $\forall e \in \mathcal{K}$ determines a bijective function $E_e : \mathcal{M} \to \mathcal{C}$. E_e is the encryption function
- $\forall d \in \mathcal{K}$ determines a bijective function $D_d : \mathcal{C} \to \mathcal{M}$. D_d is the decryption function

Applying E_e or D_d is called **encryption** or **decryption** respectively.
An **encryption scheme** (or **cipher**) consists of a set $\{E_e : e \in \mathcal{K}\}$ and a corresponding set $\{D_d : d \in \mathcal{K}\}$ with the property that for each $e \in \mathcal{K}$ there is a unique $d \in \mathcal{K}$ such that $D_d = E_e^{-1}$.

$$D_d(E_e(m)) = m \quad \forall m \in \mathcal{M}$$

2.5 Block ciphers, stream ciphers and codes

- **Block cipher**: breaks up the plaintext message into strings (blocks) of a fixed length t and encrypts one block at a time
- **Stream cipher**: $t = 1$
- **Code**: work on words of varying length

2.6 Simple Subsitution ciphers

- **Caesar cipher**: each plaintext character is replaced by the character $k \mod 26$ positions down the alphabet
- **ROT13**: shift each letter by 13 positions
- **Alphanumeric**: substitute numbers for letters

2.6.1 Affine ciphers

An affine cipher is a monoalphabetic substitution cipher such that

$$e(m) = (a \cdot m + b) \mod |\mathcal{A}|$$

for the cipher to be invertible, a must be coprime with $|\mathcal{A}|$. The decryption function is

$$d(c) = a^{-1} \cdot (c - b) \mod |\mathcal{A}|$$

where a^{-1} satisfies $1 = a \cdot a^{-1} \mod |\mathcal{A}|$
Substitution ciphers are easily broken by frequency analysis and because of the "small" key space (26! possible keys).

2.6.2 Homophonic substitution ciphers

A homophonic substitution cipher replaces each a with a randomly chosen string from $H(a)$. To decrypt a string c of t symbols, one must determine an $a \in A$ such that $c \in H(a)$. The key for the cipher is the sets $H(a)$.
Example

$$A = \{1, 2\} \quad B = \{3, 4\} \quad C = \{5, 6\}$$
$$ABC = 135, 246, 145, \ldots$$

2.6.3 Polyalphabetic substitution ciphers

A polyalphabetic substitution cipher is a block cipher with block length t over alphabet $\mathcal{A}$ where:

- $\mathcal{A}$ consists of sequences of permutations of $\mathcal{A}$ of the form $(e_1, \ldots, e_t)$
- $E_e(m_1, \ldots, m_t) = (e_1(m_1), \ldots, e_t(m_t))$
- decryption key for e is $d = (e_1^{-1}, \ldots, e_t^{-1})$

2.6.4 One-time pads (Vernam cipher)

A one-time pad is a stream cipher defined on $\mathcal{A} = \{0, 1\}$. Message $m_1, \ldots, m_n$ is encrypted by a randomly chosen binary key string $k_1, \ldots, k_n$.

Limitation of stream ciphers

- Key must not be reused

Malleability

$$F(E(K, M)) = E(K, G(M))$$

- $E(K, M) = K \oplus M$
- $F(X) = G(X) = N \oplus X$

$$F(E(K, M)) = N \oplus (K \oplus M) = K \oplus (N \oplus M) = E(K, N \oplus M) = E(K, G(M))$$

2.7 Transposition ciphers

Permute the order of the symbols in the plaintext without changing the symbols themselves

For block lenght t, let $\mathcal{K}$ be the set of permutations on $\{1, \ldots, t\}$. For each $e \in \mathcal{K}$ and $m \in \mathcal{M}$

$$E_e(m) = m_{e(1)} \ldots m_{e(t)}$$

The set of all such transformations is called a transposition cipher. To decrypt $c = c_1 c_2 \ldots c_t$ compute $D_d(c) = c_{d(1)} \ldots c_{d(t)}$

2.8 Composite ciphers

Ciphers based on just substitutions or transpositions are not secure, we can combined them but it's difficult to do by hand so cipher machines were developed (Enigma machine).