

CS 3002 Information Security

Fall 2023

1. Explain key concepts of information security such as design principles, cryptography, risk management,(1)
2. Discuss legal, ethical, and professional issues in information security (6)
3. Analyze real world scenarios, model them using security measures, and apply various security and risk management tools for achieving information security and privacy (2)
4. Identify appropriate techniques to tackle and solve problems of real life in the discipline of information security (3)
5. Understand issues related to ethics in the field of information security(8)



ISO/IEC 27001: 2013

Week # 1 & 2

Dr. Nadeem Kafi Khan

Assessment Instruments with Weights

- Semester Assessments – 20% (2 Assignments 5% + 4 Quizzes 10% + Project 5%)
 - Mid-Term 1 Exam – 15%
 - Mid-Term 2 Exam – 15%
 - End-Term Exam – 50%
- Late submissions will only awarded a maximum of 20% weightage of the respective assessment.
 - Plagiarism punishment up to 20 weightage.
- ❖ Project topics are limited to **IT Security Management and Risk Assessment Problems and their solutions including use of any tools covered in this course.**
 - ❖ **Proposals based on programming are not entertained.**
 - ❖ Marks distribution: 20% proposal , 60 Documentation and Execution, 20% Presentation

ASSIGNMENT # 1 (18th Sep ← → 28th Sep)

ASSIGNMENT # 2 (16th Oct ← → 26th Oct)

PROJECT PROPOSAL SUBMISSION → Week # 7

PROJECT SUBMISSION → Week # 12

OVERVIEW

1.1 Computer Security Concepts

- A Definition of Computer Security
- Examples
- The Challenges of Computer Security
- A Model for Computer Security

1.2 Threats, Attacks, and Assets

- Threats and Attacks
- Threats and Assets

1.3 Security Functional Requirements

1.4 Fundamental Security Design Principles

1.5 Attack Surfaces and Attack Trees

- Attack Surfaces
- Attack Trees

1.6 Computer Security Strategy

- Security Policy
- Security Implementation
- Assurance and Evaluation

1.7 Standards

1.8 Key Terms, Review Questions, and Problems

LEARNING OBJECTIVES

After studying this chapter, you should be able to:

- ◆ Describe the key security requirements of confidentiality, integrity, and availability.
- ◆ Discuss the types of security threats and attacks that must be dealt with and give examples of the types of threats and attacks that apply to different categories of computer and network assets.
- ◆ Summarize the functional requirements for computer security.
- ◆ Explain the fundamental security design principles.
- ◆ Discuss the use of attack surfaces and attack trees.
- ◆ Understand the principle aspects of a comprehensive security strategy.

A Definition of Computer Security

The NIST Internal/Interagency Report **NISTIR 7298 (*Glossary of Key Information Security Terms*, May 2013)** defines the term *computer security* as follows:

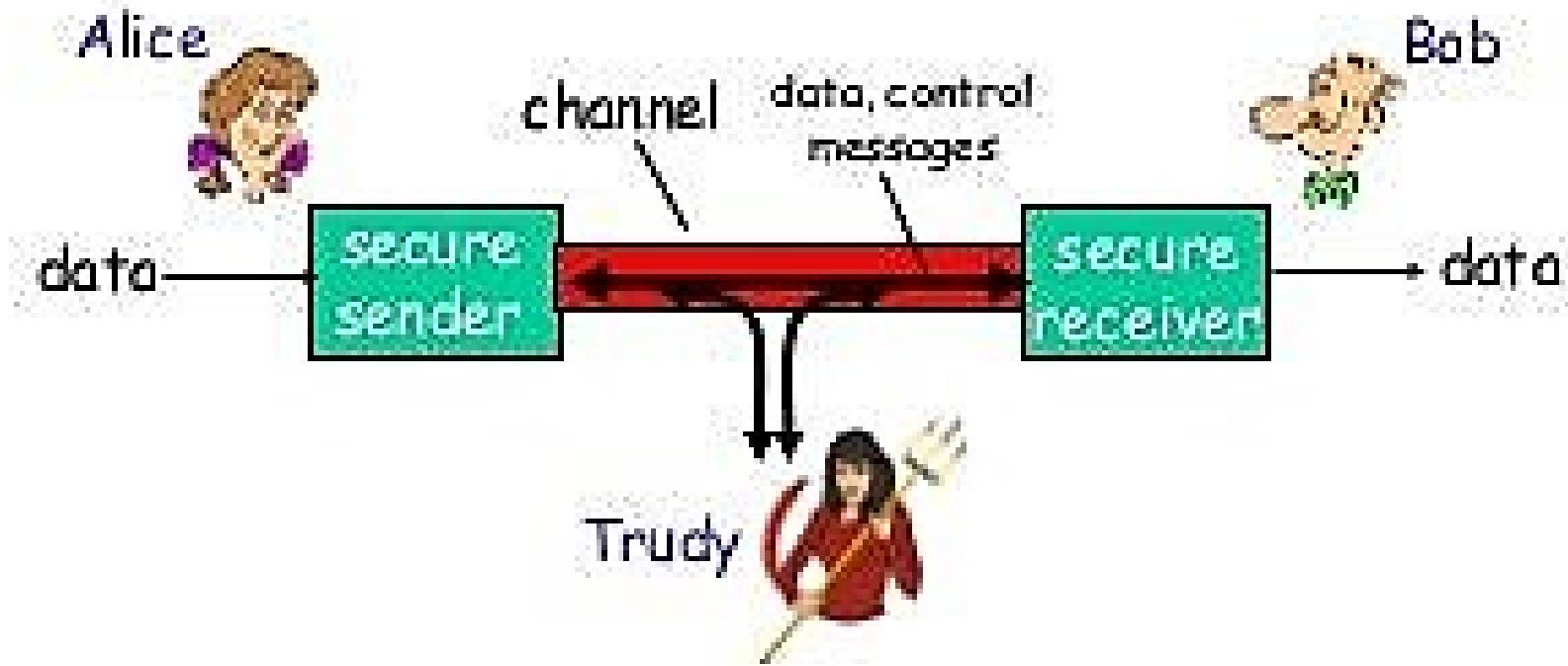
Computer Security: Measures and controls that ensure confidentiality, integrity, and availability of information system assets including hardware, software, firmware, and information being processed, stored, and communicated.

This definition introduces three key objectives that are at the heart of computer security:

- **Confidentiality:** This term covers two related concepts:
 - **Data confidentiality:**¹ Assures that private or confidential information is not made available or disclosed to unauthorized individuals.
 - **Privacy:** Assures that individuals control or influence what information related to them may be collected and stored and by whom and to whom that information may be disclosed.
- **Integrity:** This term covers two related concepts:
 - **Data integrity:** Assures that information and programs are changed only in a specified and authorized manner.
 - **System integrity:** Assures that a system performs its intended function in an unimpaired manner, free from deliberate or inadvertent unauthorized manipulation of the system.
- **Availability:** Assures that systems work promptly and service is not denied to authorized users.

Friends and enemies: Alice, Bob, Trudy

- well-known in network security world
- Bob, Alice want to communicate "securely"
- Trudy (intruder) may intercept, delete, add messages



CIA triad

- **Confidentiality:** Preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information. A loss of confidentiality is the unauthorized disclosure of information.
- **Integrity:** Guarding against improper information modification or destruction, including ensuring information nonrepudiation and authenticity. A loss of integrity is the unauthorized modification or destruction of information.
- **Availability:** Ensuring timely and reliable access to and use of information. A loss of availability is the disruption of access to or use of information or an information system.

Two additional concepts are needed to present a complete picture

- **Authenticity:** The property of being genuine and being able to be verified and trusted; confidence in the validity of a transmission, a message, or message originator. This means verifying that users are who they say they are and that each input arriving at the system came from a trusted source.
- **Accountability:** The security goal that generates the requirement for actions of an entity to be traced uniquely to that entity. This supports nonrepudiation, deterrence, fault isolation, intrusion detection and prevention, and after-action recovery and legal action. Because truly secure systems are not yet an achievable goal, we must be able to trace a security breach to a responsible party. Systems must keep records of their activities to permit later forensic analysis to trace security breaches or to aid in transaction disputes.

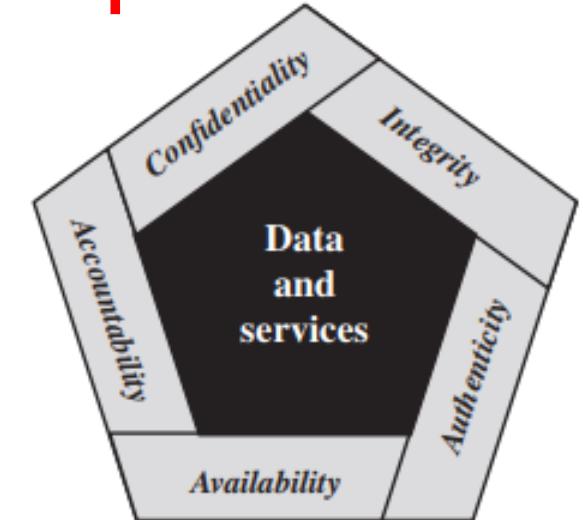


Figure 1.1 Essential Network and Computer Security Requirements

three levels of impact on organizations



Low



Moderate



High

FIPS (Federal Information Processing Standards)

Real World Examples: Confidentiality, Integrity and Availability

Impact of Risk:



Low



Moderate



High

CONFIDENTIALITY Student grade information is an asset whose confidentiality is considered to be highly important by students. In the United States, the release of such information is regulated by the Family Educational Rights and Privacy Act (FERPA). Grade information should only be available to students, their parents, and employees that require the information to do their job. Student enrollment information may have a moderate confidentiality rating. While still covered by FERPA, this information is seen by more people on a daily basis, is less likely to be targeted than grade information, and results in less damage if disclosed. Directory information, such as lists of students or faculty or departmental lists, may be assigned a low confidentiality rating or indeed no rating. This information is typically freely available to the public and published on a school's website.

Real World Examples: Confidentiality, Integrity and Availability

INTEGRITY Several aspects of integrity are illustrated by the example of a hospital patient's allergy information stored in a database. The doctor should be able to trust that the information is correct and current. Now, suppose an employee (e.g., a nurse) who is authorized to view and update this information deliberately falsifies the data to cause harm to the hospital. The database needs to be restored to a trusted basis quickly, and it should be possible to trace the error back to the person responsible. Patient allergy information is an example of an asset with a high requirement for integrity. Inaccurate information could result in serious harm or death to a patient, and expose the hospital to massive liability.

An example of an asset that may be assigned a moderate level of integrity requirement is a website that offers a forum to registered users to discuss some specific topic. Either a registered user or a hacker could falsify some entries or deface the website. If the forum exists only for the enjoyment of the users, brings in little or no advertising revenue, and is not used for something important such as research, then potential damage is not severe. The Webmaster may experience some data, financial, and time loss.

An example of a low integrity requirement is an anonymous online poll. Many websites, such as news organizations, offer these polls to their users with very few safeguards. However, the inaccuracy and unscientific nature of such polls is well understood.

Real World Examples: Confidentiality, Integrity and Availability

AVAILABILITY The more critical a component or service is, the higher will be the level of availability required. Consider a system that provides authentication services for critical systems, applications, and devices. An interruption of service results in the inability for customers to access computing resources and staff to access the resources they need to perform critical tasks. The loss of the service translates into a large financial loss in lost employee productivity and potential customer loss.

An example of an asset that would typically be rated as having a moderate availability requirement is a public website for a university; the website provides information for current and prospective students and donors. Such a site is not a critical component of the university's information system, but its unavailability will cause some embarrassment.

An online telephone directory lookup application would be classified as a low availability requirement. Although the temporary loss of the application may be an annoyance, there are other ways to access the information, such as a hardcopy directory or the operator.

The Challenges of Computer Security

Computer security is both fascinating and complex. Some of the reasons are as follows:

1. Computer security is not as simple as it might first appear to the novice. The requirements seem to be straightforward; indeed, most of the major requirements for security services can be given self-explanatory one-word labels: confidentiality, authentication, nonrepudiation, and integrity. But the mechanisms used to meet those requirements can be quite complex, and understanding them may involve rather subtle reasoning.
2. In developing a particular security mechanism or algorithm, one must always consider potential attacks on those security features. In many cases, successful attacks are designed by looking at the problem in a completely different way, therefore exploiting an unexpected weakness in the mechanism.
3. Because of Point 2, the procedures used to provide particular services are often counterintuitive. Typically, a security mechanism is complex, and it is not obvious from the statement of a particular requirement that such elaborate measures are needed. Only when the various aspects of the threat are considered do elaborate security mechanisms make sense.
4. Having designed various security mechanisms, it is necessary to decide where to use them. This is true both in terms of physical placement (e.g., at what points in a network are certain security mechanisms needed) and in a logical sense [e.g., at what layer or layers of an architecture such as TCP/IP (Transmission Control Protocol/Internet Protocol) should mechanisms be placed].
5. Security mechanisms typically involve more than a particular algorithm or protocol. They also require that participants be in possession of some secret information (e.g., an encryption key), which raises questions about the creation, distribution, and protection of that secret information. There may also be a reliance on communications protocols whose behavior may complicate the task of developing the security mechanism. For example, if the proper functioning of the security mechanism requires setting time limits on the transit time of a message from sender to receiver, then any protocol or network that introduces variable, unpredictable delays may render such time limits meaningless.
6. Computer security is essentially a battle of wits between a perpetrator who tries to find holes, and the designer or administrator who tries to close them. The great advantage that the attacker has is that he or she need only find a single weakness, while the designer must find and eliminate all weaknesses to achieve perfect security.
7. There is a natural tendency on the part of users and system managers to perceive little benefit from security investment until a security failure occurs.
8. Security requires regular, even constant monitoring, and this is difficult in today's short-term, overloaded environment.
9. Security is still too often an afterthought to be incorporated into a system after the design is complete, rather than being an integral part of the design process.
10. Many users and even security administrators view strong security as an impediment to efficient and user-friendly operation of an information system or use of information.

A Model for Computer Security

We start with the concept of a **system resource** or **asset**, that users and owners wish to protect. The assets of a computer system can be categorized as follows:

- **Hardware:** Including computer systems and other data processing, data storage, and data communications devices.
- **Software:** Including the operating system, system utilities, and applications.
- **Data:** Including files and databases, as well as security-related data, such as password files.
- **Communication facilities and networks:** Local and wide area network communication links, bridges, routers, and so on.

A Model for Computer Security

In the context of security, our concern is with the **vulnerabilities** of system resources. [NRC02] lists the following general categories of vulnerabilities of a computer system or network asset:

- The system can be **corrupted**, so it does the wrong thing or gives wrong answers. For example, stored data values may differ from what they should be because they have been improperly modified.
- The system can become **leaky**. For example, someone who should not have access to some or all of the information available through the network obtains such access.
- The system can become **unavailable** or very slow. That is, using the system or network becomes impossible or impractical.

These three types of vulnerability correspond to the concepts of integrity, confidentiality, and availability

A Model for Computer Security

Corresponding to the various types of vulnerabilities to a system resource are **threats** that are capable of exploiting those vulnerabilities. A threat represents a potential security harm to an asset. An **attack** is a threat that is carried out (threat action) and, if successful, leads to an undesirable violation of security, or threat consequence. The agent carrying out the attack is referred to as an attacker or **threat agent**. We can distinguish two types of attacks:

- **Active attack:** An attempt to alter system resources or affect their operation.
- **Passive attack:** An attempt to learn or make use of information from the system that does not affect system resources.

We can also classify attacks based on the origin of the attack:

- **Inside attack:** Initiated by an entity inside the security perimeter (an “insider”). The insider is authorized to access system resources but uses them in a way not approved by those who granted the authorization.
- **Outside attack:** Initiated from outside the perimeter, by an unauthorized or illegitimate user of the system (an “outsider”). On the Internet, potential outside attackers range from amateur pranksters to organized criminals, international terrorists, and hostile governments.

A Model for Computer Security

Finally, a **countermeasure** is any means taken to deal with a security attack. Ideally, a countermeasure can be devised to **prevent** a particular type of attack from succeeding. When prevention is not possible, or fails in some instance, the goal is to **detect** the attack then **recover** from the effects of the attack. A countermeasure may itself introduce new vulnerabilities. In any case, residual vulnerabilities may remain after the imposition of countermeasures. Such vulnerabilities may be exploited by threat agents representing a residual level of **risk** to the assets. Owners will seek to minimize that risk given other constraints.

Table 1.1 Computer Security Terminology

Adversary (threat agent)

Individual, group, organization, or government that conducts or has the intent to conduct detrimental activities.

Attack

Any kind of malicious activity that attempts to collect, disrupt, deny, degrade, or destroy information system resources or the information itself.

Countermeasure

A device or techniques that has as its objective the impairment of the operational effectiveness of undesirable or adversarial activity, or the prevention of espionage, sabotage, theft, or unauthorized access to or use of sensitive information or information systems.

Risk

A measure of the extent to which an entity is threatened by a potential circumstance or event, and typically a function of 1) the adverse impacts that would arise if the circumstance or event occurs; and 2) the likelihood of occurrence.

Table 1.1 Computer Security Terminology

Security Policy

A set of criteria for the provision of security services. It defines and constrains the activities of a data processing facility in order to maintain a condition of security for systems and data.

System Resource (Asset)

A major application, general support system, high impact program, physical plant, mission critical system, personnel, equipment, or a logically related group of systems.

Threat

Any circumstance or event with the potential to adversely impact organizational operations (including mission, functions, image, or reputation), organizational assets, individuals, other organizations, or the Nation through an information system via unauthorized access, destruction, disclosure, modification of information, and/or denial of service.

Vulnerability

Weakness in an information system, system security procedures, internal controls, or implementation that could be exploited or triggered by a threat source.

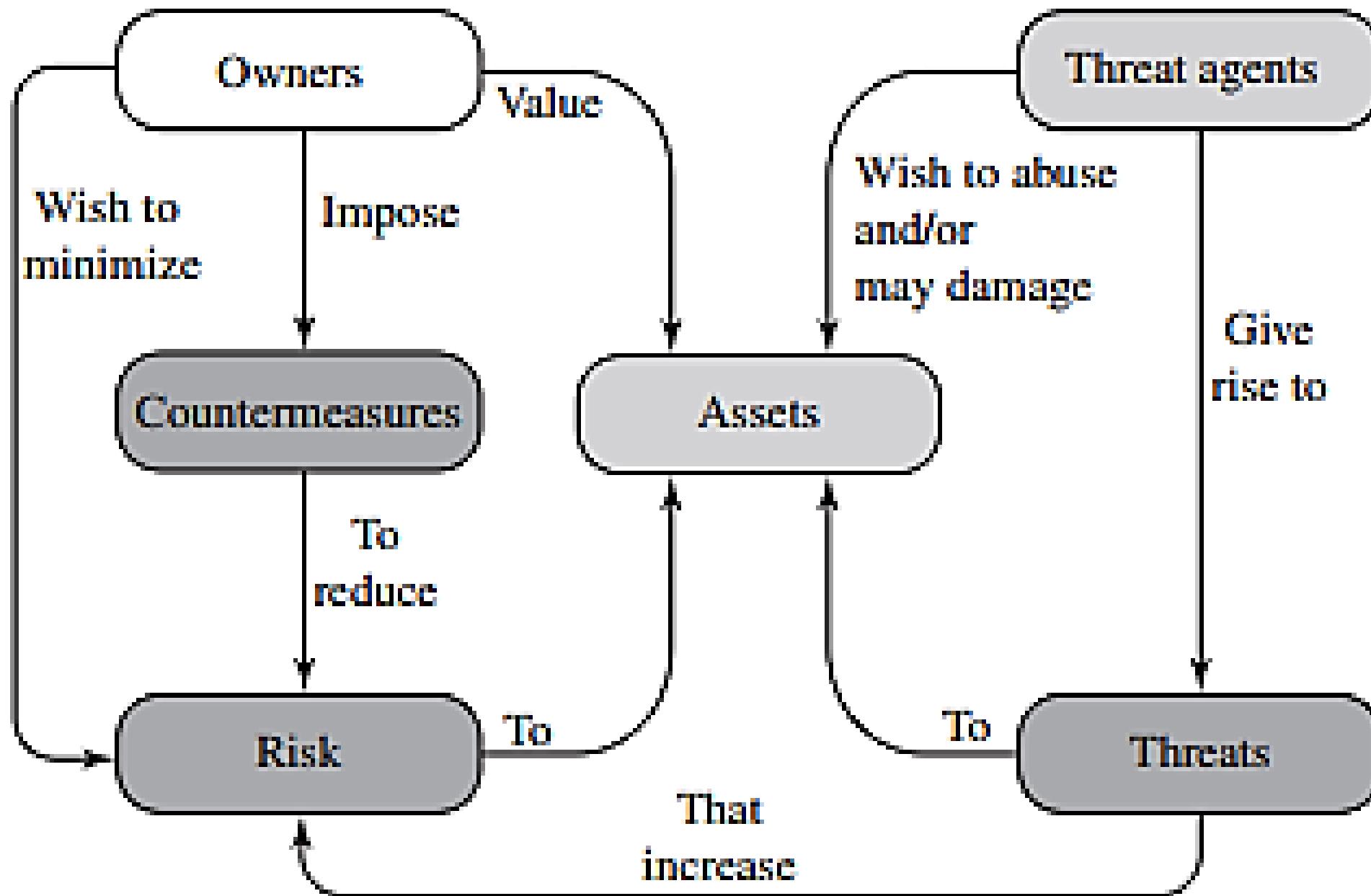


Figure 1.2 Security Concepts and Relationships

Table 1.2 Threat Consequences, and the Types of Threat Actions that Cause Each Consequence

Threat Consequence	Threat Action (Attack)
Unauthorized Disclosure A circumstance or event whereby an entity gains access to data for which the entity is not authorized.	Exposure: Sensitive data are directly released to an unauthorized entity. Interception: An unauthorized entity directly accesses sensitive data traveling between authorized sources and destinations. Inference: A threat action whereby an unauthorized entity indirectly accesses sensitive data (but not necessarily the data contained in the communication) by reasoning from characteristics or by-products of communications. Intrusion: An unauthorized entity gains access to sensitive data by circumventing a system's security protections.
Deception A circumstance or event that may result in an authorized entity receiving false data and believing it to be true.	Masquerade: An unauthorized entity gains access to a system or performs a malicious act by posing as an authorized entity. Falsification: False data deceive an authorized entity. Repudiation: An entity deceives another by falsely denying responsibility for an act.

Table 1.2 Threat Consequences, and the Types of Threat Actions that Cause Each Consequence

Threat Consequence	Threat Action (Attack)
Disruption A circumstance or event that interrupts or prevents the correct operation of system services and functions.	Incapacitation: Prevents or interrupts system operation by disabling a system component. Corruption: Undesirably alters system operation by adversely modifying system functions or data. Obstruction: A threat action that interrupts delivery of system services by hindering system operation.
Usurpation A circumstance or event that results in control of system services or functions by an unauthorized entity.	Misappropriation: An entity assumes unauthorized logical or physical control of a system resource. Misuse: Causes a system component to perform a function or service that is detrimental to system security.

Computer system

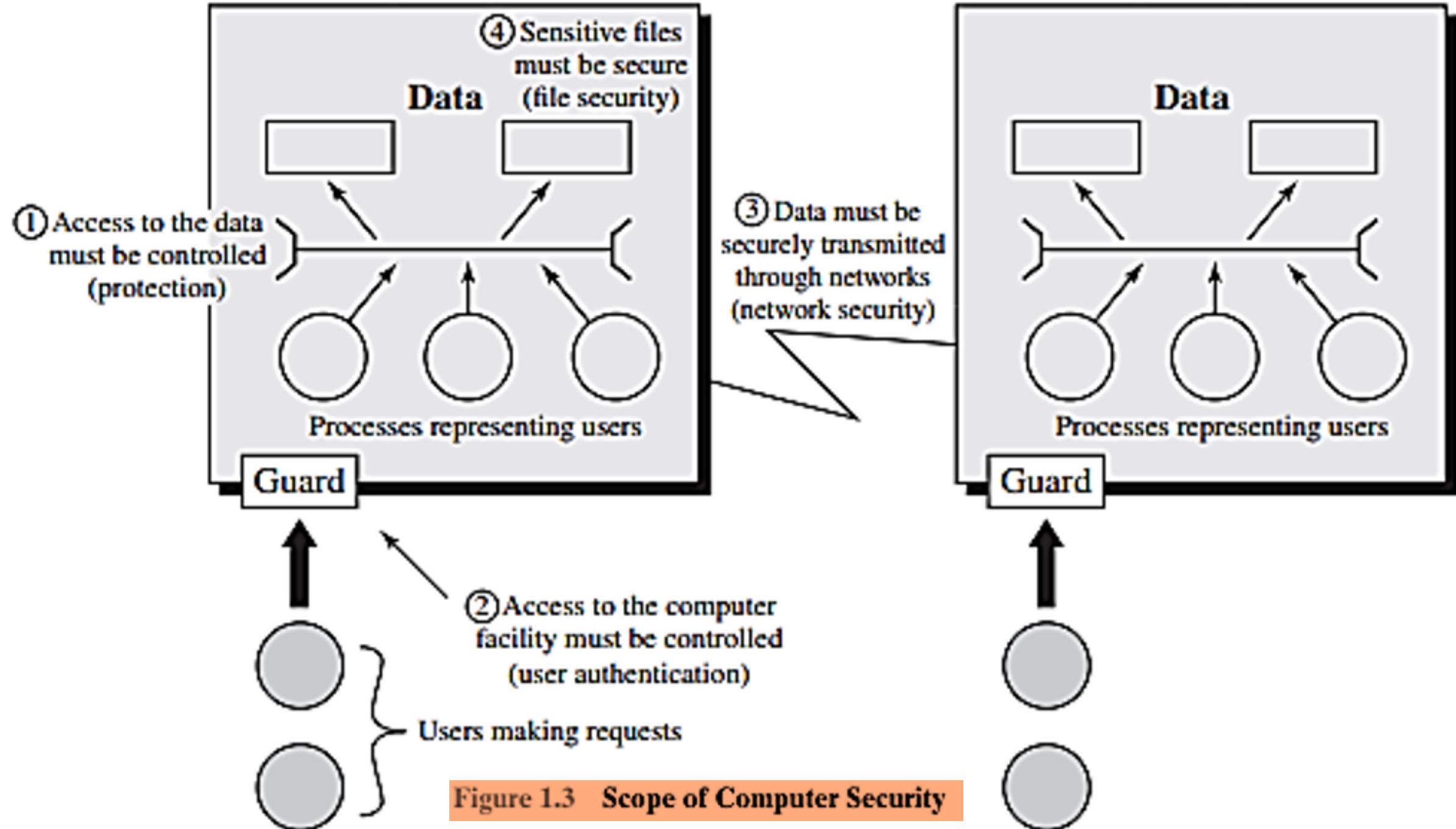


Figure 1.3 Scope of Computer Security

Table 1.3 Computer and Network Assets, with Examples of Threats

	Availability	Confidentiality	Integrity
Hardware	Equipment is stolen or disabled, thus denying service.	An unencrypted USB drive is stolen.	
Software	Programs are deleted, denying access to users.	An unauthorized copy of software is made.	A working program is modified, either to cause it to fail during execution or to cause it to do some unintended task.
Data	Files are deleted, denying access to users.	An unauthorized read of data is performed. An analysis of statistical data reveals underlying data.	Existing files are modified or new files are fabricated.
Communication Lines and Networks	Messages are destroyed or deleted. Communication lines or networks are rendered unavailable.	Messages are read. The traffic pattern of messages is observed.	Messages are modified, delayed, reordered, or duplicated. False messages are fabricated.

1.4 FUNDAMENTAL SECURITY DESIGN PRINCIPLES

Despite years of research and development, it has not been possible to develop security design and implementation techniques that systematically exclude security flaws and prevent all unauthorized actions. In the absence of such foolproof techniques, it is useful to have a set of widely agreed design principles that can guide the development of protection mechanisms. The National Centers of Academic Excellence in Information Assurance/Cyber Defense, which is jointly sponsored by the U.S. National Security Agency and the U. S. Department of Homeland Security, list the following as fundamental security design principles [NCAE13]:

- Economy of mechanism
- Separation of privilege
- Isolation
- Fail-safe defaults
- Least privilege
- Encapsulation
- Complete mediation
- Least common mechanism
- Modularity
- Open design
- Psychological acceptability
- Layering
- Least astonishment

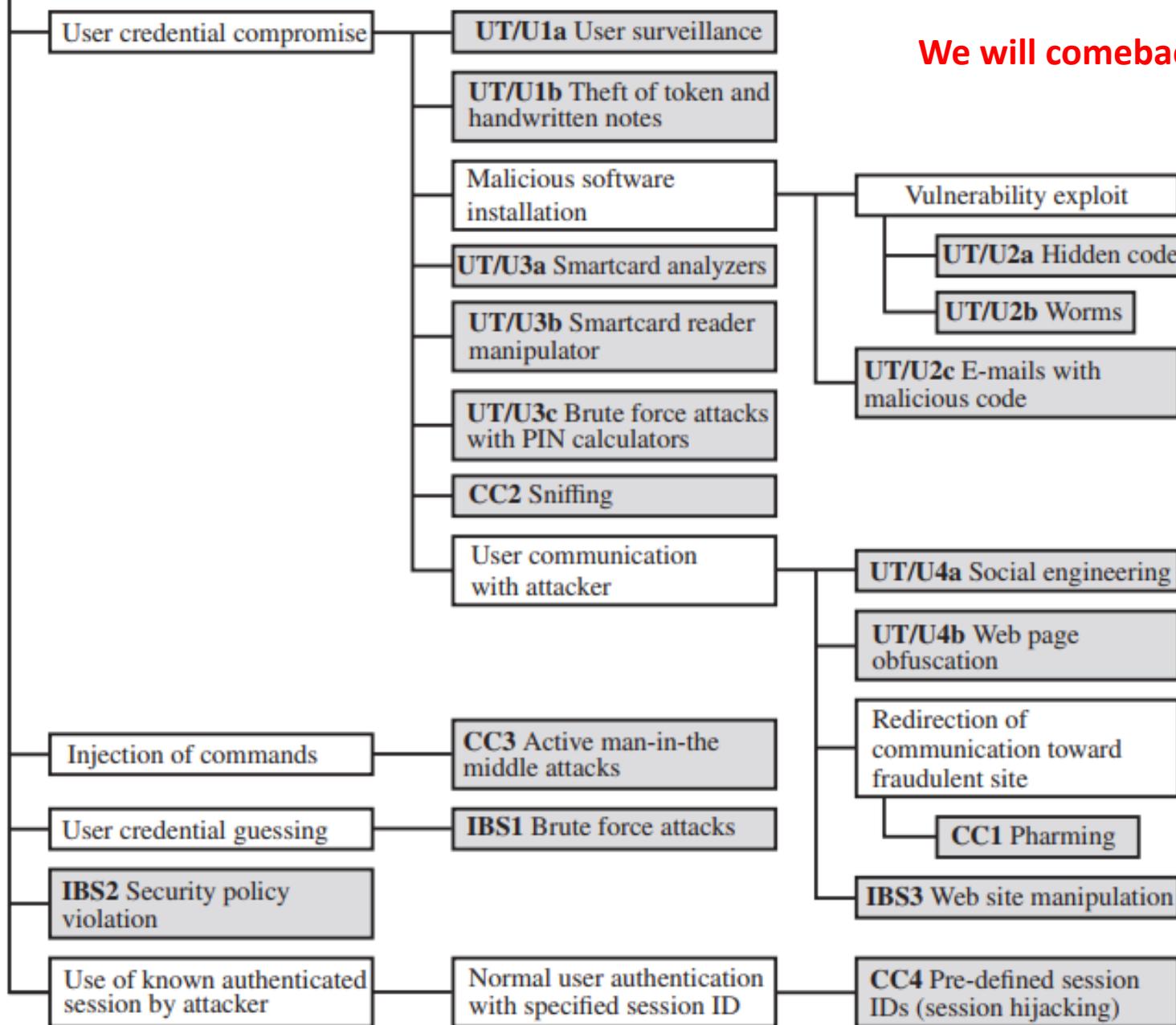
Attack surfaces can be categorized in the following way:

- **Network attack surface:** This category refers to vulnerabilities over an enterprise network, wide-area network, or the Internet. Included in this category are network protocol vulnerabilities, such as those used for a denial-of-service attack, disruption of communications links, and various forms of intruder attacks.
- **Software attack surface:** This refers to vulnerabilities in application, utility, or operating system code. A particular focus in this category is Web server software.
- **Human attack surface:** This category refers to vulnerabilities created by personnel or outsiders, such as social engineering, human error, and trusted insiders.

Attack Trees

An attack tree is a branching, hierarchical data structure that represents a set of potential techniques for exploiting security vulnerabilities [MAUW05, MOOR01, SCHN99]. The security incident that is the goal of the attack is represented as the root node of the tree, and the ways by which an attacker could reach that goal are iteratively and incrementally represented as branches and subnodes of the tree. Each subnode defines a subgoal, and each subgoal may have its own set of further subgoals, and so on. The final nodes on the paths outward from the root, that is, the leaf nodes, represent different ways to initiate an attack.

Bank Account Compromise



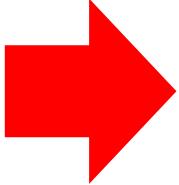
We will comeback to this slides later.

Figure 1.5 An Attack Tree for Internet Banking Authentication

1.6 COMPUTER SECURITY STRATEGY

We conclude this chapter with a brief look at the overall strategy for providing computer security. [LAMP04] suggests that a comprehensive security strategy involves three aspects:

- **Specification/policy:** What is the security scheme supposed to do?
- **Implementation/mechanisms:** How does it do it?
- **Correctness/assurance:** Does it really work?



ISO 9001:2015 employs the process approach, which incorporates the Plan-Do-Check-Act (PDCA) cycle and risk-based thinking. This means the organisation needs to:

1. determine required process inputs and expected outputs
2. assign responsibilities and authorities for processes
3. identify risks and opportunities for processes, and plan to address these

InfoSec Practice

- Identification of Assets
- Identify Vulnerabilities
 - (CIA + Authentication + Auditable)
- Risk Assessment → Possible Threats
- Known and possible Attacks
- Test and Implement known Countermeasures
 - Based on community Best Practices
- Governmental and Community based efforts
 - Groups (Meetups) + CERTs + Conferences + Certifications

1.7 STANDARDS

- **National Institute of Standards and Technology:** NIST is a U.S. federal agency that deals with measurement science, standards, and technology related to U.S. government use and to the promotion of U.S. private sector innovation. Despite its national scope, NIST Federal Information Processing Standards (FIPS) and Special Publications (SP) have a worldwide impact.
- **ITU-T:** The International Telecommunication Union (ITU) is a United Nations agency in which governments and the private sector coordinate global telecom networks and services. The ITU Telecommunication Standardization Sector (ITU-T) is one of the three sectors of the ITU. ITU-T's mission is the production of standards covering all fields of telecommunications. ITU-T standards are referred to as Recommendations.
- **Internet Society:** ISOC is a professional membership society with worldwide organizational and individual membership. It provides leadership in addressing issues that confront the future of the Internet, and is the organization home for the groups responsible for Internet infrastructure standards, including the Internet Engineering Task Force (IETF) and the Internet Architecture Board (IAB). These organizations develop Internet standards and related specifications, all of which are published as Requests for Comments (RFCs).
- **ISO:** The International Organization for Standardization (ISO) is a worldwide federation of national standards bodies from more than 140 countries. ISO is a nongovernmental organization that promotes the development of standardization and related activities with a view to facilitating the international exchange of goods and services, and to developing cooperation in the spheres of intellectual, scientific, technological, and economic activity. ISO's work results in international agreements that are published as International Standards.

CHAPTER 2

CRYPTOGRAPHIC TOOLS

2.1 Confidentiality with Symmetric Encryption

Symmetric Encryption

Symmetric Block Encryption Algorithms

Stream Ciphers

2.1 CONFIDENTIALITY WITH SYMMETRIC ENCRYPTION

Encryption is the process of converting information or data into a code, especially to prevent unauthorized access.

Symmetric Encryption

Symmetric encryption, also referred to as conventional encryption or single-key encryption, was the only type of encryption in use prior to the introduction of public-key encryption in the late 1970s. Countless individuals and groups, from Julius Caesar to the German U-boat force to present-day diplomatic, military, and commercial users, have used symmetric encryption for secret communication. It remains the more widely used of the two types of encryption.

2.1 CONFIDENTIALITY WITH SYMMETRIC ENCRYPTION

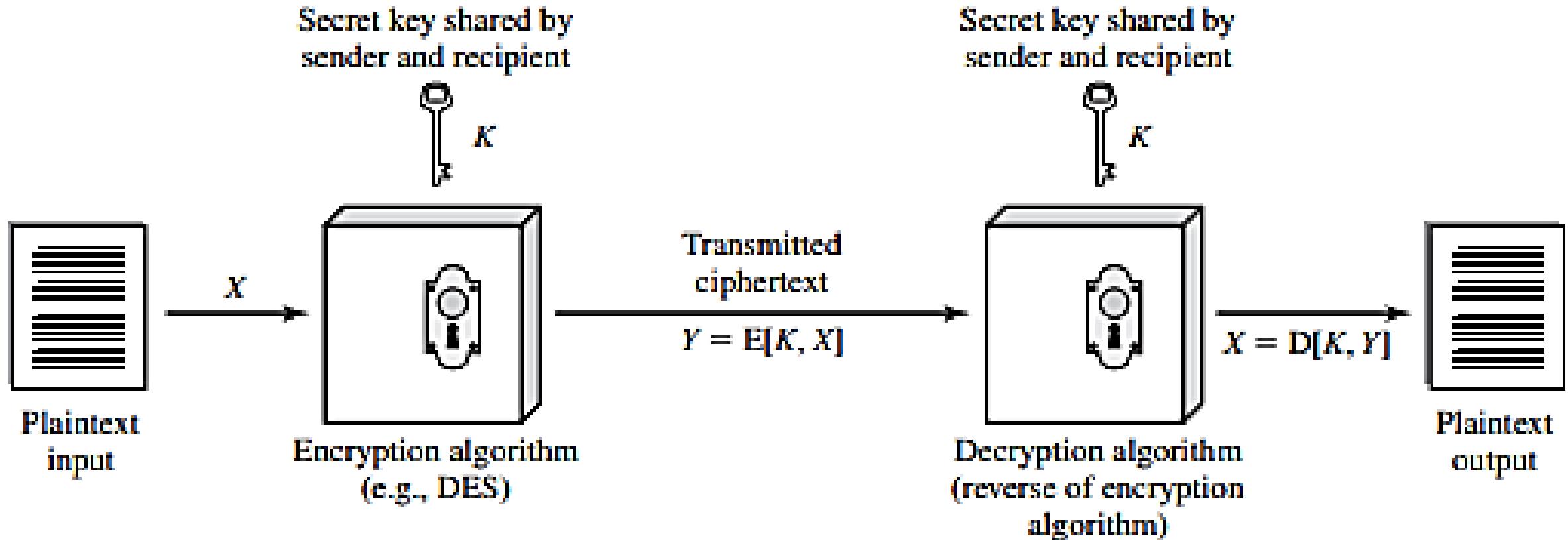


Figure 2.1 Simplified Model of Symmetric Encryption

2.1 CONFIDENTIALITY WITH SYMMETRIC ENCRYPTION

A symmetric encryption scheme has five ingredients (see Figure 2.1):

- **Plaintext:** This is the original message or data that is fed into the algorithm as input.
- **Encryption algorithm:** The encryption algorithm performs various substitutions and transformations on the plaintext.
- **Secret key:** The secret key is also input to the encryption algorithm. The exact substitutions and transformations performed by the algorithm depend on the key.
- **Ciphertext:** This is the scrambled message produced as output. It depends on the plaintext and the secret key. For a given message, two different keys will produce two different ciphertexts.
- **Decryption algorithm:** This is essentially the encryption algorithm run in reverse. It takes the ciphertext and the secret key and produces the original plaintext.

2.1 CONFIDENTIALITY WITH SYMMETRIC ENCRYPTION

There are two requirements for secure use of symmetric encryption:

1. We need a strong encryption algorithm. At a minimum, we would like the algorithm to be such that an opponent who knows the algorithm and has access to one or more ciphertexts would be unable to decipher the ciphertext or figure out the key. This requirement is usually stated in a stronger form: The opponent should be unable to decrypt ciphertext or discover the key even if he or she is in possession of a number of ciphertexts together with the plaintext that produced each ciphertext.
2. The sender and receiver must have obtained copies of the secret key in a secure fashion and must keep the key secure. If someone can discover the key and knows the algorithm, all communication using this key is readable.

2.1 CONFIDENTIALITY WITH SYMMETRIC ENCRYPTION

Attacking a Symmetric Encryption Scheme (1)

- **Brute-force attack** is to try every possible key on a piece of cipher text until an intelligible translation into plaintext is obtained.
 - On average, half of all possible keys must be tried to achieve success. That is, if there are x different keys, on average an attacker would discover the actual key after $x/2$ tries.
 - This means size of key has a great impact of breaking the symmetric Encryption schedule. Larger the key the more time it will takes to try all possible key combinations.

Table 2.2 Average Time Required for Exhaustive Key Search

Key Size (bits)	Cipher	Number of Alternative Keys	Time Required at 10^9 decryptions/ μs	Time Required at 10^{13} decryptions/ μs
56	DES	$2^{56} = 7.2 \times 10^{16}$	$2^{55} \mu\text{s} = 1.125 \text{ years}$	1 hour
128	AES	$2^{128} = 3.4 \times 10^{38}$	$2^{127} \mu\text{s} = 5.3 \times 10^{21} \text{ years}$	$5.3 \times 10^{17} \text{ years}$
168	Triple DES	$2^{168} = 3.7 \times 10^{50}$	$2^{167} \mu\text{s} = 5.8 \times 10^{33} \text{ years}$	$5.8 \times 10^{29} \text{ years}$
192	AES	$2^{192} = 6.3 \times 10^{57}$	$2^{191} \mu\text{s} = 9.8 \times 10^{40} \text{ years}$	$9.8 \times 10^{36} \text{ years}$
256	AES	$2^{256} = 1.2 \times 10^{77}$	$2^{255} \mu\text{s} = 1.8 \times 10^{60} \text{ years}$	$1.8 \times 10^{56} \text{ years}$

2.1 CONFIDENTIALITY WITH SYMMETRIC ENCRYPTION

Attacking a Symmetric Encryption Scheme (2)

- How to break a cipher in less time than the time taken to perform a brute-force attack with half key space?
- More information about plain text is needed
 - Language of the plain text. This make it easy to recognize the result.
 - If the text message has been compressed before encryption, then recognition is more difficult.
 - A numerical file that has been compressed, the problem becomes even more difficult.

2.1 CONFIDENTIALITY WITH SYMMETRIC ENCRYPTION

Attacking a Symmetric Encryption Scheme (2)

- **Cryptanalysis.** Cryptanalysis refers to the process of analyzing information systems in order to understand 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.**
- These attacks rely on the **nature of the algorithm** plus perhaps some knowledge of the general characteristics of
 - the plaintext, or
 - some knowledge of some plaintext-ciphertext pairs.
- This type of attack exploits the characteristics of the algorithm to attempt to deduce a specific plaintext or to deduce the key being used. If the attack succeeds in deducing the key, the effect is catastrophic: All future and past messages encrypted with that key are compromised.

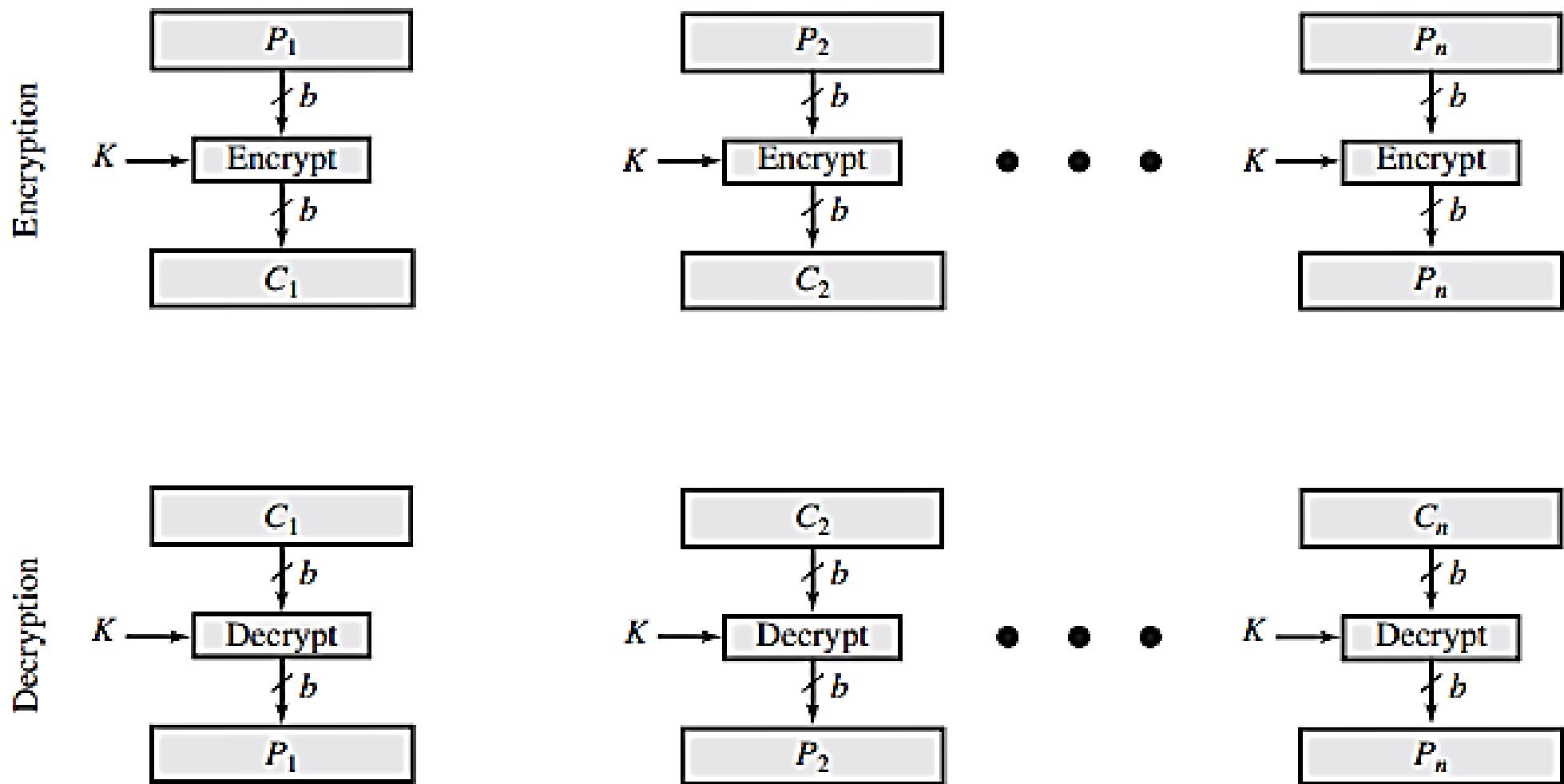
Types of Symmetric Encryption

Symmetric Block Encryption Algorithms

The most commonly used symmetric encryption algorithms are block ciphers. A block cipher processes the plaintext input in fixed-size blocks and produces a block of ciphertext of equal size for each plaintext block. The algorithm processes longer plaintext amounts as a series of fixed-size blocks. The most important symmetric algo-

Stream Ciphers

A *block cipher* processes the input one block of elements at a time, producing an output block for each input block. A *stream cipher* processes the input elements continuously, producing output one element at a time, as it goes along. Although block ciphers are far more common, there are certain applications in which a stream cipher is more appropriate.



(a) Block cipher encryption (electronic codebook mode)

Figure 2.2 Types of Symmetric Encryption

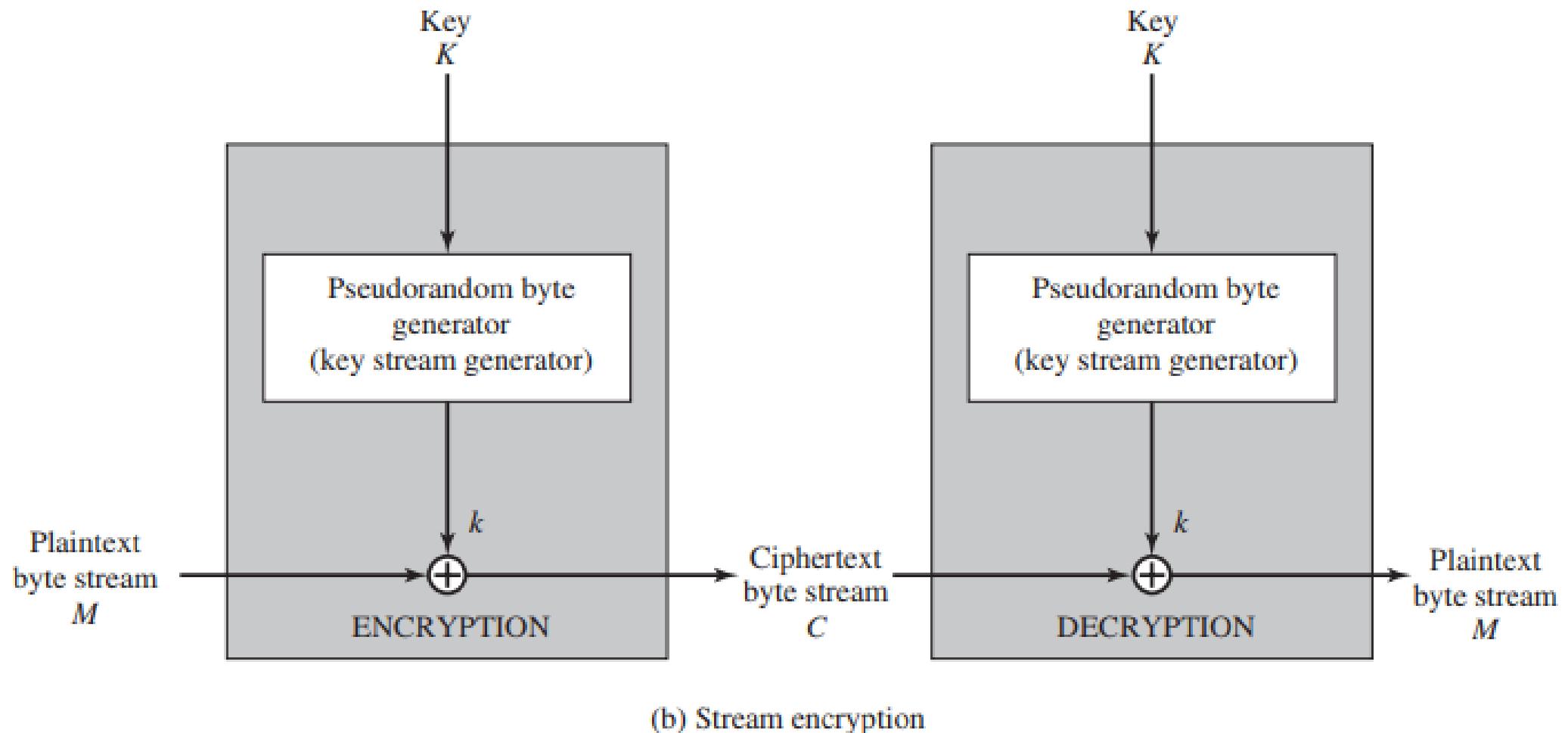


Figure 2.2 Types of Symmetric Encryption

Symmetric Block Encryption Algorithms

The most important symmetric algorithms, all of which are block ciphers, are the Data Encryption Standard (DES), triple DES, and the Advanced Encryption Standard (AES); see Table 2.1. This subsection provides an overview of these algorithms.

Table 2.1 Comparison of Three Popular Symmetric Encryption Algorithms

	DES	Triple DES	AES
Plaintext block size (bits)	64	64	128
Ciphertext block size (bits)	64	64	128
Key size (bits)	56	112 or 168	128, 192, or 256

DES = Data Encryption Standard

AES = Advanced Encryption Standard