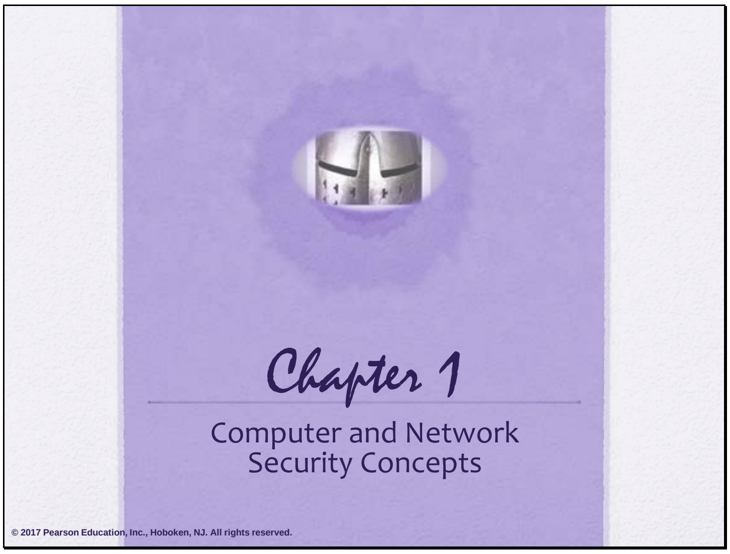
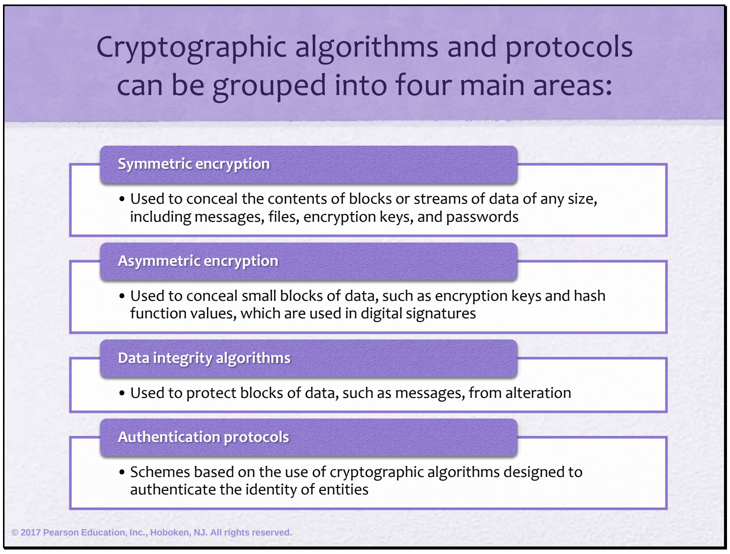
Slide 1



This book focuses on two broad areas: cryptographic algorithms and protocols, which
have a broad range of applications; and network and Internet security, which rely heavily on cryptographic techniques.

Slide 2



Cryptographic algorithms and protocols can be grouped into four main areas:

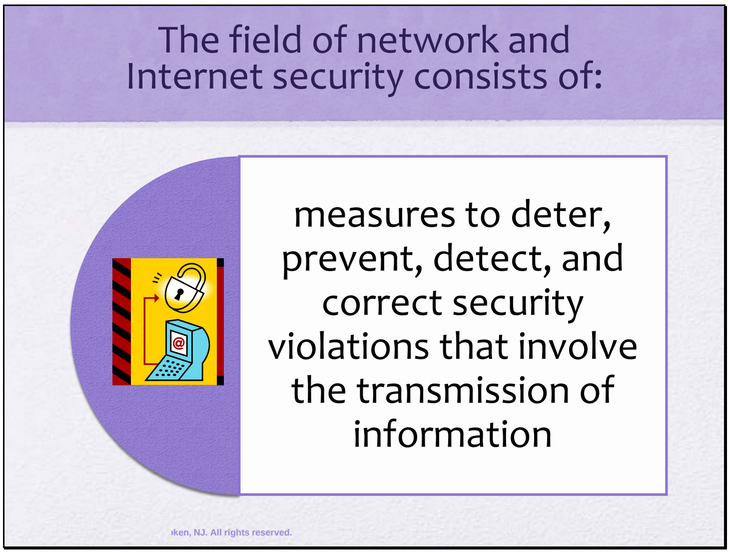
• Symmetric encryption: Used to conceal the contents of blocks or streams of data of any size, including messages, files, encryption keys, and passwords.

• Asymmetric encryption: Used to conceal small blocks of data, such as encryption keys and hash function values, which are used in digital signatures.

• Data integrity algorithms: Used to protect blocks of data, such as messages, from alteration.

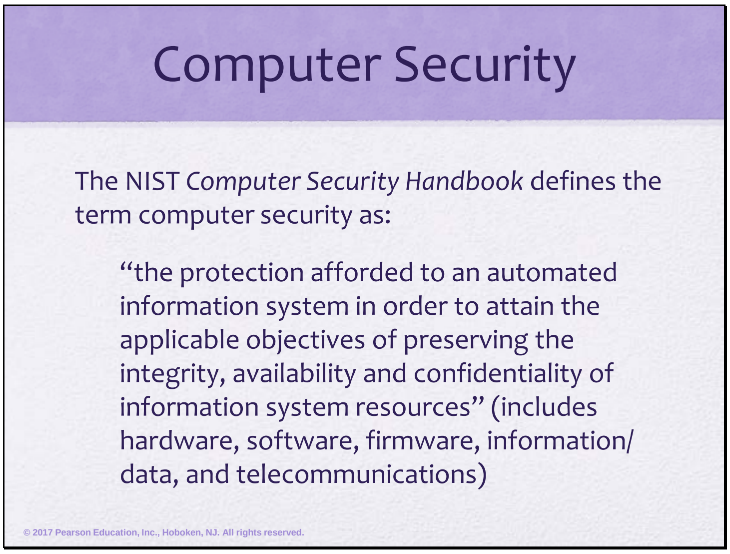
• Authentication protocols: These are schemes based on the use of cryptographic algorithms designed to authenticate the identity of entities.

Slide 3



The field of network and Internet security consists of measures to deter, prevent, detect, and correct security violations that involve the transmission of information. That is a broad statement that covers a host of possibilities.

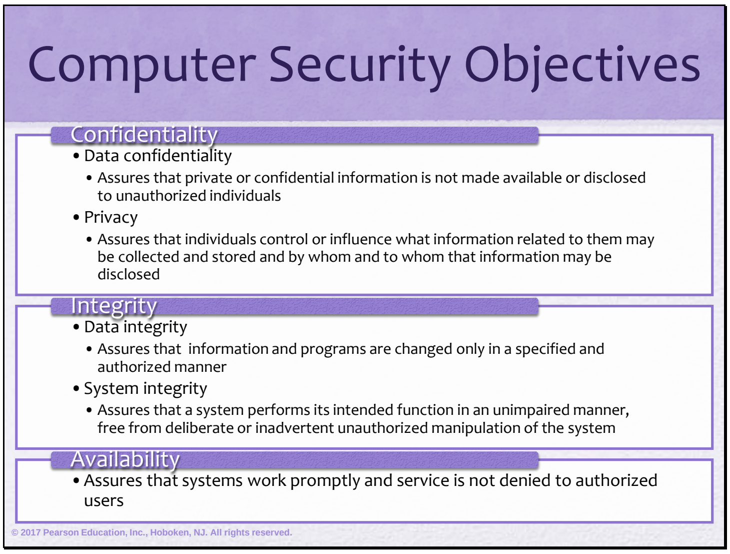
Slide 4



The NIST Computer Security Handbook [NIST95] defines the term computer security
as follows:

Computer Security: The protection afforded to an automated information system in order to attain the applicable objectives of preserving the integrity, availability, and confidentiality of information system resources (includes hardware, software, firmware, information/data, and telecommunications).

Slide 5



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

• Confidentiality: This term covers two related concepts:

Data confidentiality: 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.

Slide 6

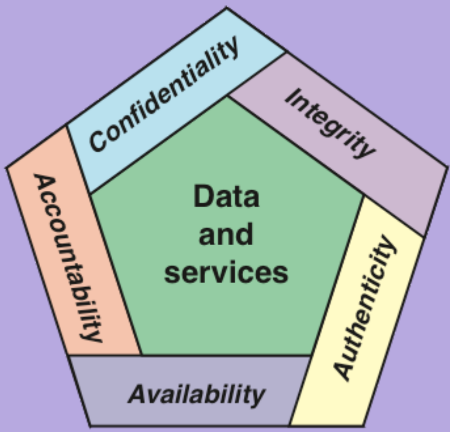


Figure 1.1 Essential Network and Computer Security Requirements

These three concepts form what is often referred to as the CIA triad . The three concepts embody the fundamental security objectives for both data and for information
and computing services. For example, the NIST standard FIPS 199 (Standards for Security Categorization of Federal Information and Information Systems) lists confidentiality, integrity, and availability as the three security objectives for information
and for information systems. FIPS 199 provides a useful characterization of these three objectives in terms of requirements and the definition of a loss of security in each category:

• 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.

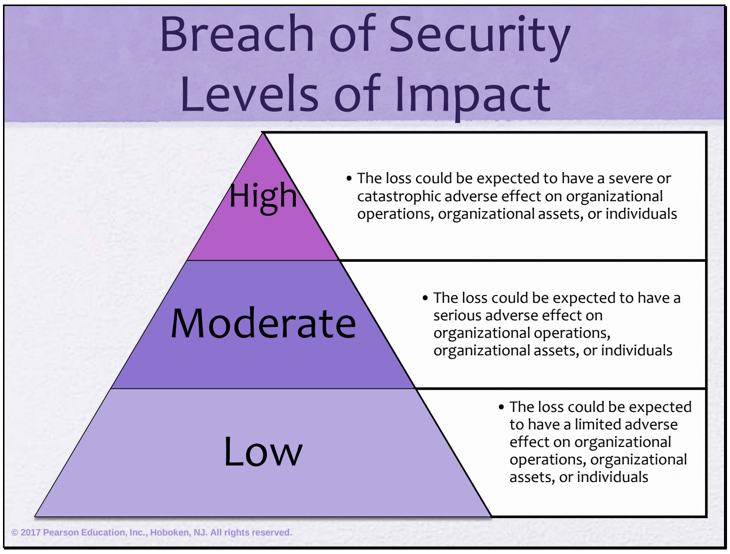
• Availability: Ensuring timely and reliable access to and use of information. A loss of availability is the disruption of access to or use of information or an information system.

Although the use of the CIA triad to define security objectives is well established, some in the security field feel that additional concepts are needed to present a complete picture. Two of the most commonly mentioned are as follows:

• 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.

Slide 7



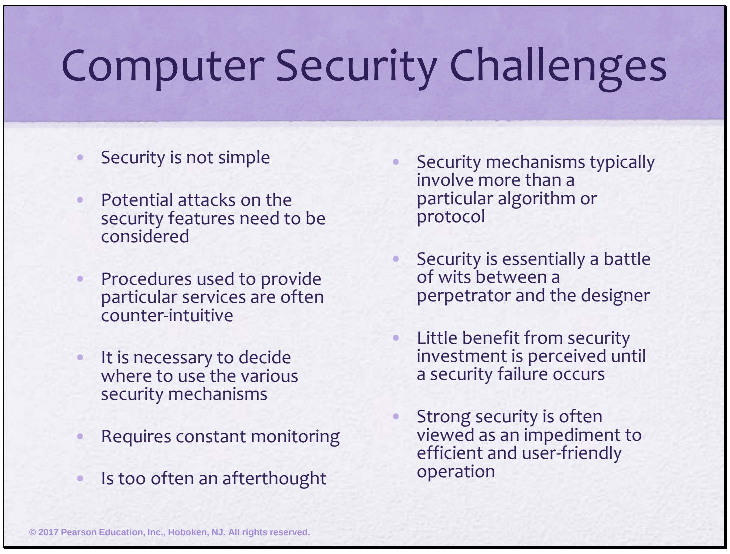
We use three levels of impact on organizations or individuals should there be a breach of security (i.e., a loss of confidentiality, integrity, or availability). These levels are defined in FIPS PUB 199:

• Low: The loss could be expected to have a limited adverse effect on organizational operations, organizational assets, or individuals. A limited adverse effect means that, for example, the loss of confidentiality, integrity, or availability might (i) cause a degradation in mission capability to an extent and duration that the organization is able to perform its primary functions, but the effectiveness of the functions is noticeably reduced; (ii) result in minor damage to organizational assets; (iii) result in minor financial loss; or (iv) result in minor harm to individuals.

• Moderate: The loss could be expected to have a serious adverse effect on organizational operations, organizational assets, or individuals. A serious adverse effect means that, for example, the loss might (i) cause a significant degradation in mission capability to an extent and duration that the organization is able to perform its primary functions, but the effectiveness of the functions is significantly reduced; (ii) result in significant damage to organizational assets; (iii) result in significant financial loss; or (iv) result in significant harm to individuals that does not involve loss of life or serious, life-threatening injuries.

• High: The loss could be expected to have a severe or catastrophic adverse effect on organizational operations, organizational assets, or individuals. A severe or catastrophic adverse effect means that, for example, the loss might (i) cause a severe degradation in or loss of mission capability to an extent and duration that the organization is not able to perform one or more of its primary functions; (ii) result in major damage to organizational assets; (iii) result in major financial loss; or (iv) result in severe or catastrophic harm to individuals involving loss of life or serious, life-threatening injuries.

Slide 8



Computer and network security is both fascinating and complex. Some of the reasons follow:

1. 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, or 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. It is only when the various aspects of the threat are considered that 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 also may 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 and network 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.

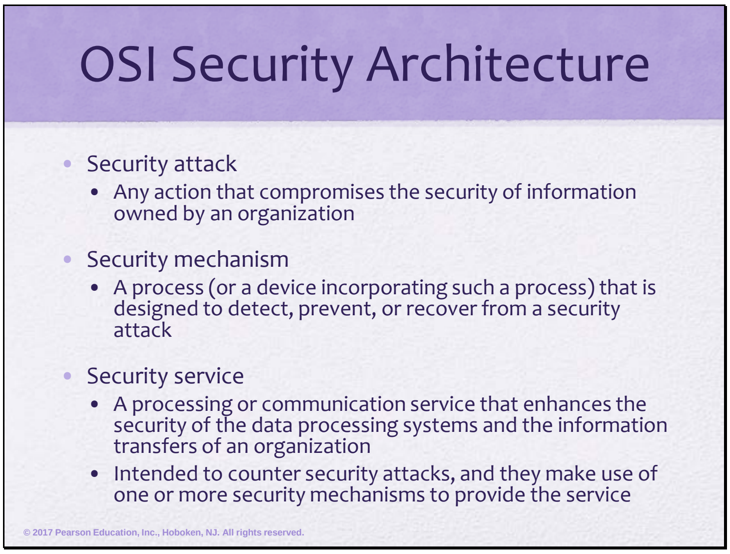
7. There is a natural tendency on the part of users and system managers to perceive little benefit from security investment until a security failure occurs.

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.

Slide 9



To assess effectively the security needs of an organization and to evaluate and choose various security products and policies, the manager responsible for security needs some systematic way of defining the requirements for security and characterizing
the approaches to satisfying those requirements. This is difficult enough in a centralized data processing environment; with the use of local and wide area networks,
the problems are compounded.

ITU-T Recommendation X.800, Security Architecture for OSI , defines such a systematic approach. The OSI security architecture is useful to managers as a way of organizing the task of providing security. Furthermore, because this architecture was developed as an international standard, computer and communications vendors have developed security features for their products and services that relate to this structured definition of services and mechanisms.

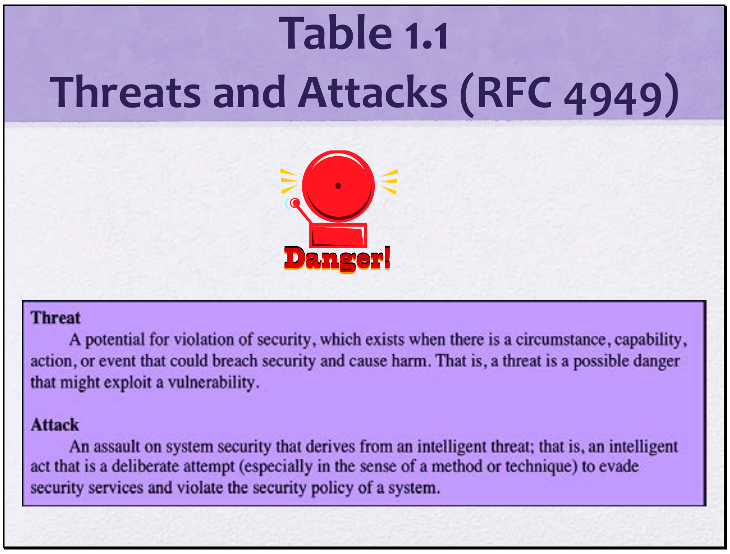
For our purposes, the OSI security architecture provides a useful, if abstract, overview of many of the concepts that this book deals with. The OSI security architecture
focuses on security attacks, mechanisms, and services. These can be defined briefly as

• Security attack: Any action that compromises the security of information owned by an organization.

• Security mechanism: A process (or a device incorporating such a process) that is designed to detect, prevent, or recover from a security attack.

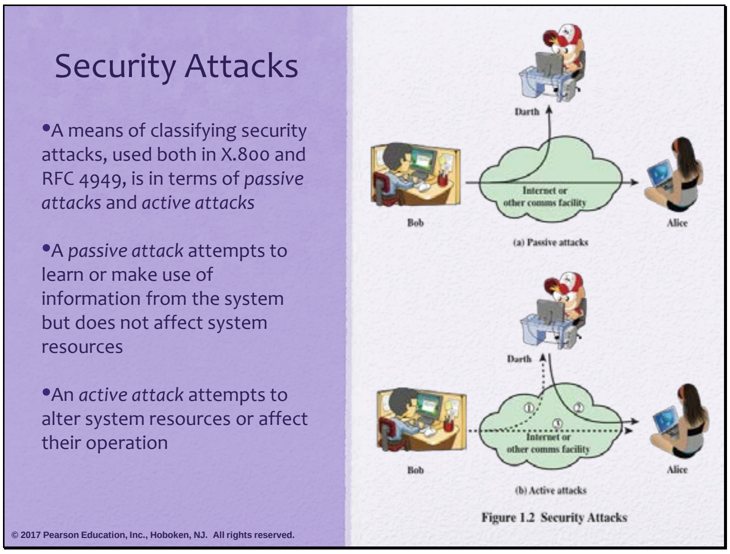
• Security service: A processing or communication service that enhances the security of the data processing systems and the information transfers of an organization. The services are intended to counter security attacks, and they make use of one or more security mechanisms to provide the service.

Slide 10



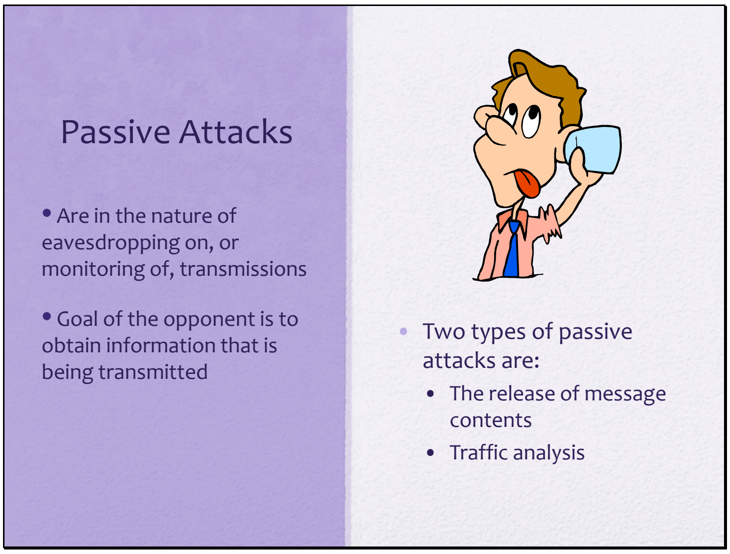
In the literature, the terms threat and attack are commonly used to mean more or less the same thing. Table 1.1 provides definitions taken from RFC 4949, Internet Security Glossary.

Slide 11



A useful means of classifying security attacks, used both in X.800 and RFC 4949, is in
terms of passive attacks and active attacks (Figure 1.2). A passive attack attempts to
learn or make use of information from the system but does not affect system resources.
An active attack attempts to alter system resources or affect their operation.

Slide 12



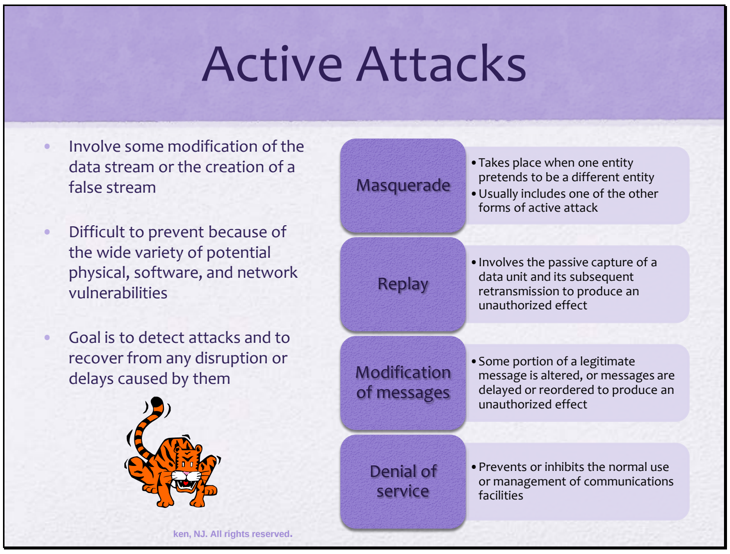
Passive attacks (Figure 1.2a) are in the nature of eavesdropping on, or monitoring of, transmissions. The goal of the opponent is to obtain information that is being transmitted. Two types of passive attacks are the release of message contents and traffic analysis.

The release of message contents is easily understood. A telephone conversation, an electronic mail message, and a transferred file may contain sensitive or confidential information. We would like to prevent an opponent from learning the contents of these transmissions.

A second type of passive attack, traffic analysis , is subtler. Suppose that we had a way of masking the contents of messages or other information traffic so that opponents, even if they captured the message, could not extract the information from the message. The common technique for masking contents is encryption. If we had encryption protection in place, an opponent might still be able to observe the pattern of these messages. The opponent could determine the location and identity of communicating hosts and could observe the frequency and length of messages being exchanged. This information might be useful in guessing the nature of the communication that was taking place.

Passive attacks are very difficult to detect, because they do not involve any alteration of the data. Typically, the message traffic is sent and received in an apparently normal fashion, and neither the sender nor receiver is aware that a third party has read the messages or observed the traffic pattern. However, it is feasible to prevent the success of these attacks, usually by means of encryption. Thus, the emphasis in dealing with passive attacks is on prevention rather than detection.

Slide 13



Active attacks (Figure 1.2b) involve some modification of the data stream or the creation of a false stream and can be subdivided into four categories: masquerade, replay, modification of messages, and denial of service.

A masquerade takes place when one entity pretends to be a different entity (path 2 of Figure 1.2b is active). A masquerade attack usually includes one of the other forms of active attack. For example, authentication sequences can be captured and replayed after a valid authentication sequence has taken place, thus enabling an authorized entity with few privileges to obtain extra privileges by impersonating an entity that has those privileges.

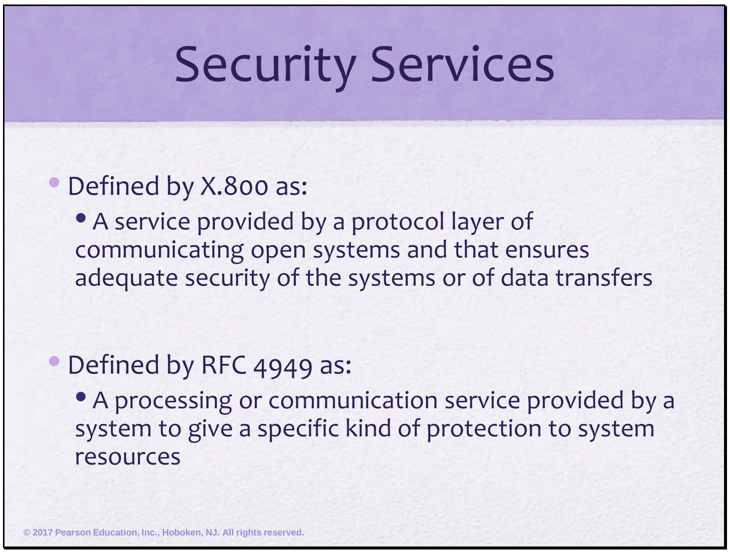
Replay involves the passive capture of a data unit and its subsequent retransmission
to produce an unauthorized effect (paths 1, 2, and 3 active).

Modification of messages simply means that some portion of a legitimate message is altered, or that messages are delayed or reordered, to produce an unauthorized effect (paths 1 and 2 active). For example, a message meaning “Allow John Smith to read confidential file accounts ” is modified to mean “Allow Fred Brown to read confidential file accounts. ”

The denial of service prevents or inhibits the normal use or management of communications facilities (path 3 active). This attack may have a specific target; for example, an entity may suppress all messages directed to a particular destination (e.g., the security audit service). Another form of service denial is the disruption of an entire network, either by disabling the network or by overloading it with messages so as to degrade performance.

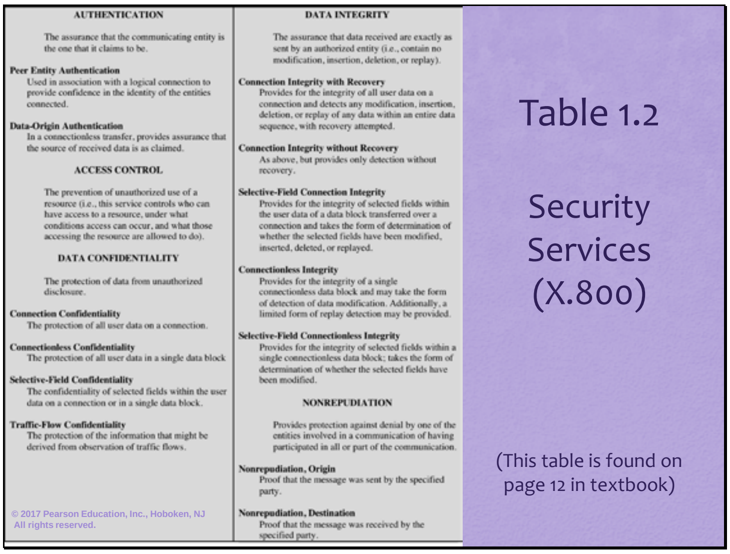
Active attacks present the opposite characteristics of passive attacks. Whereas passive attacks are difficult to detect, measures are available to prevent their success.
On the other hand, it is quite difficult to prevent active attacks absolutely because of the wide variety of potential physical, software, and network vulnerabilities.
Instead, the goal is to detect active attacks and to recover from any disruption or delays caused by them. If the detection has a deterrent effect, it may also contribute to prevention.

Slide 14



X.800 defines a security service as a service that is provided by a protocol layer of communicating open systems and that ensures adequate security of the systems or of data transfers. Perhaps a clearer definition is found in RFC 4949, which provides the following definition: a processing or communication service that is provided by a system to give a specific kind of protection to system resources; security services implement security policies and are implemented by security mechanisms.

Slide 15



AUTHENTICATION

The assurance that the communicating entity is the one that it claims to be.

Peer Entity Authentication
Used in association with a logical connection to provide confidence in the identity of the entities connected.

Data-Origin Authentication
In a connectionless transfer, provides assurance that the source of received data is as claimed.

ACCESS CONTROL

The prevention of unauthorized use of a resource (i.e., this service controls who can have access to a resource, under what conditions access can occur, and what those accessing the resource are allowed to do).

DATA CONFIDENTIALITY

The protection of data from unauthorized disclosure.

Connection Confidentiality
The protection of all user data on a connection.

Connectionless Confidentiality
The protection of all user data in a single data block

Selective-Field Confidentiality
The confidentiality of selected fields within the user data on a connection or in a single data block.

Traffic-Flow Confidentiality
The protection of the information that might be derived from observation of traffic flows.

DATA INTEGRITY

The assurance that data received are exactly as sent by an authorized entity (i.e., contain no modification, insertion, deletion, or replay).

Connection Integrity with Recovery
Provides for the integrity of all user data on a connection and detects any modification, insertion, deletion, or replay of any data within an entire data sequence, with recovery attempted.

Connection Integrity without Recovery
As above, but provides only detection without recovery.

Selective-Field Connection Integrity
Provides for the integrity of selected fields within the user data of a data block transferred over a connection and takes the form of determination of whether the selected fields have been modified, inserted, deleted, or replayed.

Connectionless Integrity
Provides for the integrity of a single connectionless data block and may take the form of detection of data modification. Additionally, a limited form of replay detection may be provided.

Selective-Field Connectionless Integrity
Provides for the integrity of selected fields within a single connectionless data block; takes the form of determination of whether the selected fields have been modified.

NONREPUDIATION

Provides protection against denial by one of the entities involved in a communication of having participated in all or part of the communication.

Nonrepudiation, Origin
Proof that the message was sent by the specified party.

Nonrepudiation, Destination
Proof that the message was received by the specified party.

Table 1.2

Security Services (X.800)

(This table is found on page 12 in textbook)

X.800 divides these services into five categories and fourteen specific services (Table 1.2).

Slide 16

The authentication service is concerned with assuring that a communication is authentic. In the case of a single message, such as a warning or alarm signal, the function of the authentication service is to assure the recipient that the message is from the source that it claims to be from. In the case of an ongoing interaction, such as the connection of a terminal to a host, two aspects are involved. First, at the time of connection initiation, the service assures that the two entities are authentic, that is, that each is the entity that it claims to be. Second, the service must assure that the connection is not interfered with in such a way that a third party can masquerade as one of the two legitimate parties for the purposes of unauthorized transmission or reception.

Two specific authentication services are defined in X.800:

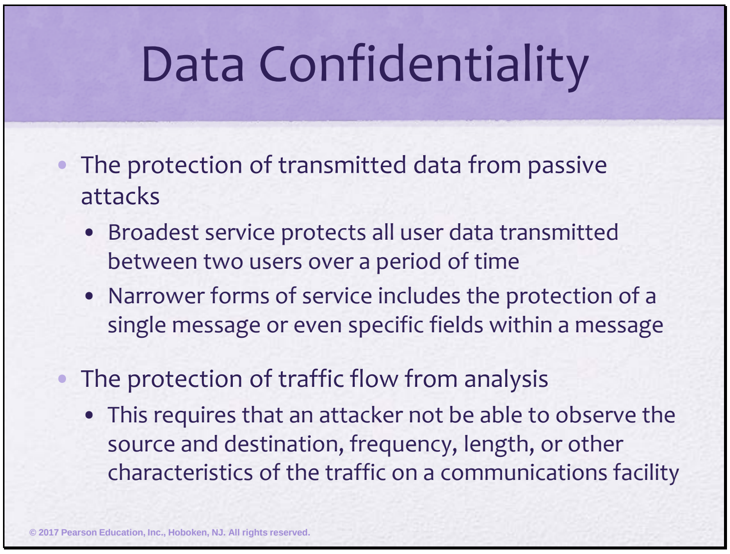
• Peer entity authentication: Provides for the corroboration of the identity of a peer entity in an association. Two entities are considered peers if they implement to same protocol in different systems; for example two TCP modules in two communicating systems. Peer entity authentication is provided for use at the establishment of, or at times during the data transfer phase of, a connection. It attempts to provide confidence that an entity is not performing either a masquerade or an unauthorized replay of a previous connection.

• Data origin authentication: Provides for the corroboration of the source of a data unit. It does not provide protection against the duplication or modification of data units. This type of service supports applications like electronic mail, where there are no prior interactions between the communicating entities.

Slide 17

In the context of network security, access control is the ability to limit and control the access to host systems and applications via communications links. To achieve this, each entity trying to gain access must first be identified, or authenticated, so that access rights can be tailored to the individual.

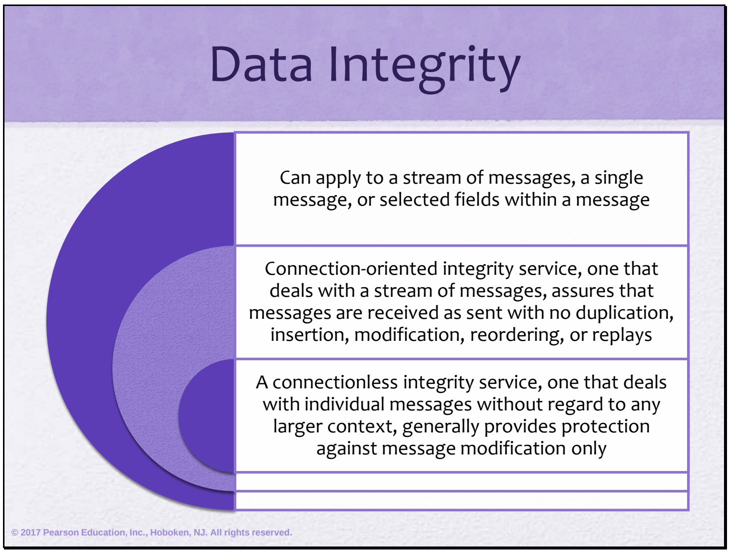
Slide 18



Confidentiality is the protection of transmitted data from passive attacks. With respect to the content of a data transmission, several levels of protection can be identified. The broadest service protects all user data transmitted between two users over a period of time. For example, when a TCP connection is set up between two systems, this broad protection prevents the release of any user data transmitted over the TCP connection. Narrower forms of this service can also be defined, including the protection of a single message or even specific fields within a message. These refinements are less useful than the broad approach and may even be more complex and expensive to implement.

The other aspect of confidentiality is the protection of traffic flow from analysis. This requires that an attacker not be able to observe the source and destination, frequency,
length, or other characteristics of the traffic on a communications facility.

Slide 19

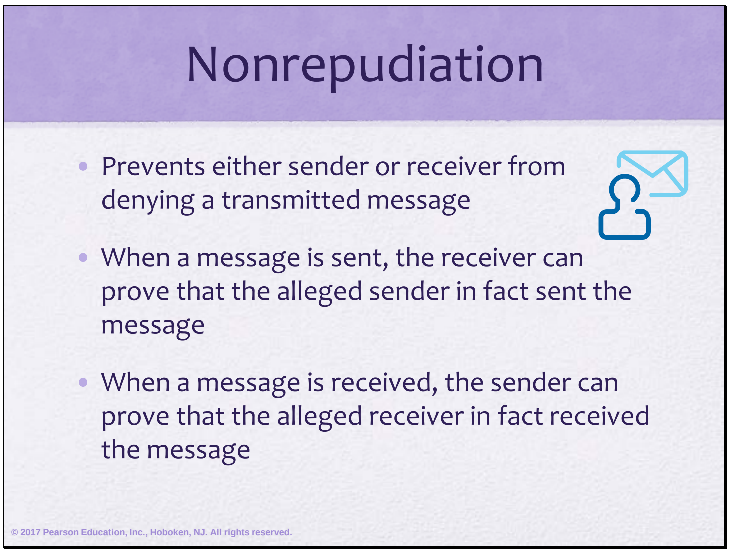


As with confidentiality, integrity can apply to a stream of messages, a single message,
or selected fields within a message. Again, the most useful and straightforward approach is total stream protection.

A connection-oriented integrity service, one that deals with a stream of messages, assures that messages are received as sent with no duplication, insertion, modification, reordering, or replays. The destruction of data is also covered under this service. Thus, the connection-oriented integrity service addresses both message stream modification and denial of service. On the other hand, a connectionless integrity
service, one that deals with individual messages without regard to any larger context, generally provides protection against message modification only.

We can make a distinction between service with and without recovery.
Because the integrity service relates to active attacks, we are concerned with detection
rather than prevention. If a violation of integrity is detected, then the service may simply report this violation, and some other portion of software or human intervention is required to recover from the violation. Alternatively, there are mechanisms available to recover from the loss of integrity of data, as we will review subsequently. The incorporation of automated recovery mechanisms is, in general, the more attractive alternative.

Slide 20



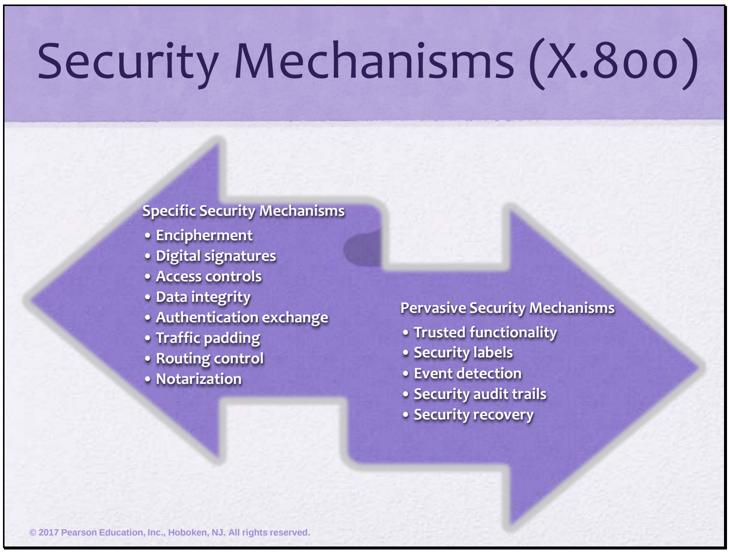
Nonrepudiation prevents either sender or receiver from denying a transmitted message.
Thus, when a message is sent, the receiver can prove that the alleged sender in fact sent the message. Similarly, when a message is received, the sender can prove that the alleged receiver in fact received the message.

Slide 21

Both X.800 and RFC 4949 define availability to be the property of a system or a system resource being accessible and usable upon demand by an authorized system entity, according to performance specifications for the system (i.e., a system is available if it provides services according to the system design whenever users request them). A variety of attacks can result in the loss of or reduction in availability. Some of these attacks are amenable to automated countermeasures, such as authentication and encryption, whereas others require some sort of physical action to prevent or recover from loss of availability of elements of a distributed system.

X.800 treats availability as a property to be associated with various security services. However, it makes sense to call out specifically an availability service. An availability service is one that protects a system to ensure its availability. This service addresses the security concerns raised by denial-of-service attacks. It depends on proper management and control of system resources and thus depends on access control service and other security services.

Slide 22



X.800 security mechanisms.

Slide 23

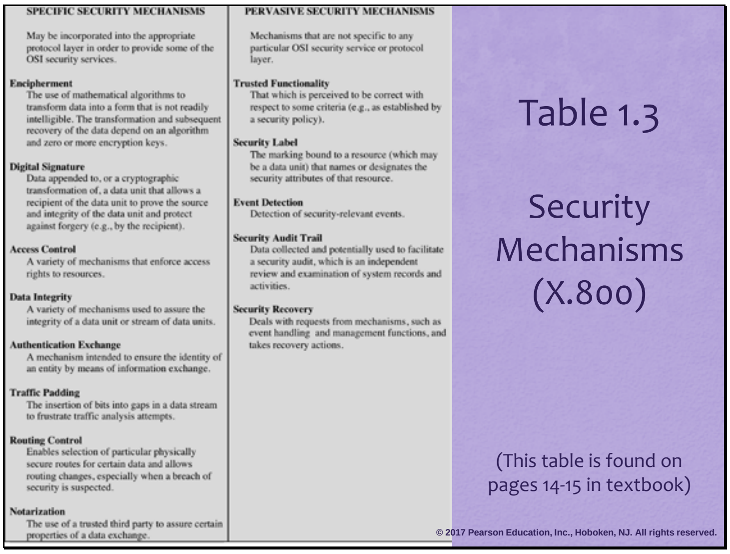
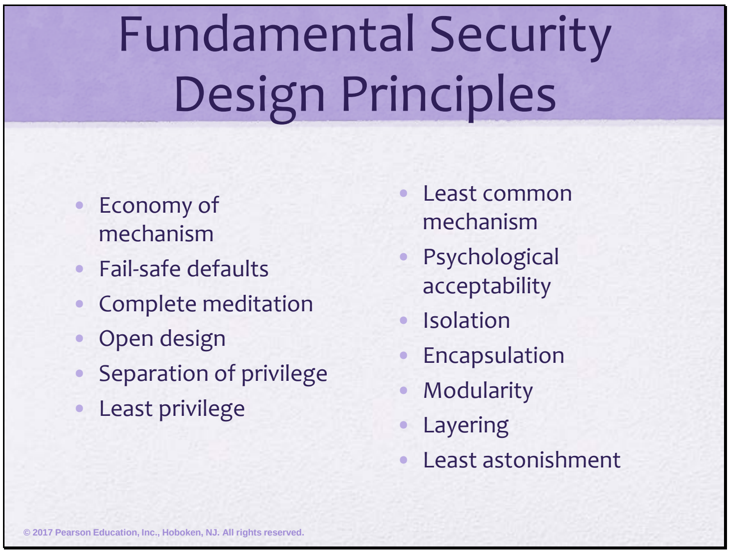


Table 1.3 lists the security mechanisms defined in X.800. The mechanisms are divided into those that are implemented in a specific protocol layer, such as TCP or an application-layer protocol, and those that are not specific to any particular protocol layer or security service. These mechanisms will be covered in the appropriate places in the book. So we do not elaborate now, except to comment on the definition of encipherment. X.800 distinguishes between reversible encipherment

mechanisms and irreversible encipherment mechanisms. A reversible encipherment mechanism is simply an encryption algorithm that allows data to be encrypted and subsequently decrypted. Irreversible encipherment mechanisms include hash algorithms and message authentication codes, which are used in digital signature and message authentication applications.

Slide 24



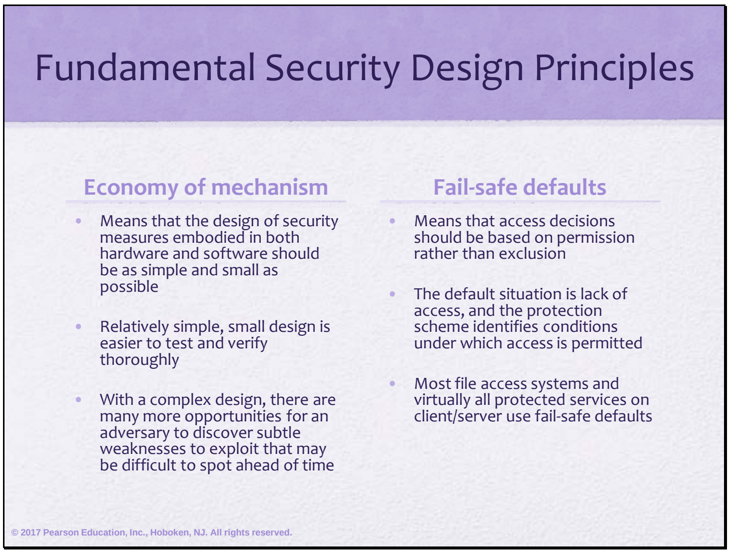
Despite years of research and development, it has not been possible to develop security design and implementation techniques that systematically exclude security flaws and prevent all unauthorized actions. In the absence of such foolproof techniques, it is useful to have a set of widely agreed design 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
- Fail-safe defaults
- Complete mediation
- Open design
- Separation of privilege
- Least privilege
- Least common mechanism
- Psychological acceptability
- Isolation
- Encapsulation

- Modularity
- Layering
- Least astonishment

The first eight listed principles were first proposed in [SALT75] and have withstood
the test of time.

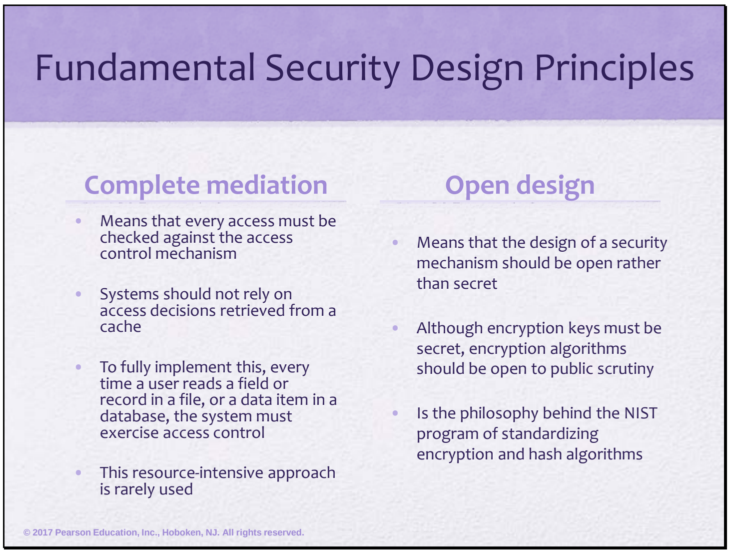
Slide 25



Economy of mechanism means that the design of security measures embodied in both hardware and software should be as simple and small as possible. The motivation for this principle is that relatively simple, small design is easier to test and verify thoroughly. With a complex design, there are many more opportunities for an adversary to discover subtle weaknesses to exploit that may be difficult to spot ahead of time. The more complex the mechanism, the more likely it is to possess exploitable flaws. Simple mechanisms tend to have fewer exploitable flaws and require less maintenance. Further, because configuration management issues are simplified, updating or replacing a simple mechanism becomes a less intensive process. In practice, this is perhaps the most difficult principle to honor. There is a constant demand for new features in both hardware and software, complicating the security design task. The best that can be done is to keep this principle in mind during system design to try to eliminate unnecessary complexity.

Fail-safe defaults means that access decisions should be based on permission rather than exclusion. That is, the default situation is lack of access, and the protection scheme identifies conditions under which access is permitted. This approach exhibits a better failure mode than the alternative approach, where the default is to permit access. A design or implementation mistake in a mechanism that gives explicit permission tends to fail by refusing permission, a safe situation that can be quickly detected. On the other hand, a design or implementation mistake in a mechanism that explicitly excludes access tends to fail by allowing access, a failure that may long go unnoticed in normal use. Most file access systems and virtually all protected services on client/server systems use fail-safe defaults.

Slide 26

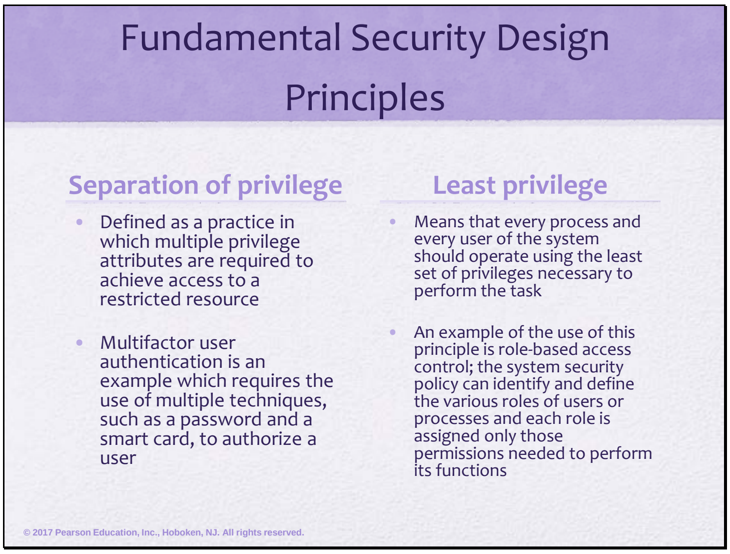


Complete mediation means that every access must be checked against the
Access control mechanism. Systems should not rely on access decisions retrieved
from a cache. In a system designed to operate continuously, this principle requires
that, if access decisions are remembered for future use, careful consideration be
given to how changes in authority are propagated into such local memories. File
access systems appear to provide an example of a system that complies with this
principle. However, typically, once a user has opened a file, no check is made to see
if permissions change. To fully implement complete mediation, every time a user
reads a field or record in a file, or a data item in a database, the system must exercise
access control. This resource-intensive approach is rarely used.

Open design means that the design of a security mechanism should be open
rather than secret. For example, although encryption keys must be secret, encryption
algorithms should be open to public scrutiny. The algorithms can then be reviewed
by many experts, and users can therefore have high confidence in them. This is the

philosophy behind the National Institute of Standards and Technology (NIST) Program of standardizing encryption and hash algorithms, and has led to the widespread adoption of NIST-approved algorithms.

Slide 27

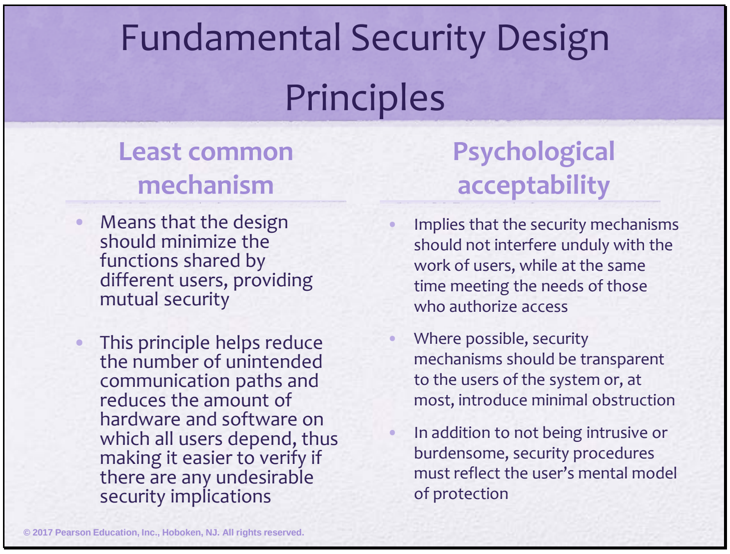


Separation of privilege is defined in [SALT75] as a practice in which multiple privilege attributes are required to achieve access to a restricted resource. A good example of this is multifactor user authentication, which requires the use of multiple techniques, such as a password and a smart card, to authorize a user. The term is also now applied to any technique in which a program is divided into parts that are limited to the specific privileges they require in order to perform a specific task. This is used to mitigate the potential damage of a computer security attack. One example of this latter interpretation of the principle is removing high privilege operations to another process and running that process with the higher privileges required to perform its tasks. Day-to-day interfaces are executed in a lower privileged process.

Least privilege means that every process and every user of the system should operate using the least set of privileges necessary to perform the task. A good example of the use of this principle is role-based access control. The system security

policy can identify and define the various roles of users or processes. Each role is
assigned only those permissions needed to perform its functions. Each permission
specifies a permitted access to a particular resource (such as read and write access
to a specified file or directory, connect access to a given host and port). Unless a
permission is granted explicitly, the user or process should not be able to access the
protected resource. More generally, any access control system should allow each
user only the privileges that are authorized for that user. There is also a temporal
aspect to the least privilege principle. For example, system programs or administrators
who have special privileges should have those privileges only when necessary;
when they are doing ordinary activities the privileges should be withdrawn. Leaving
them in place just opens the door to accidents.

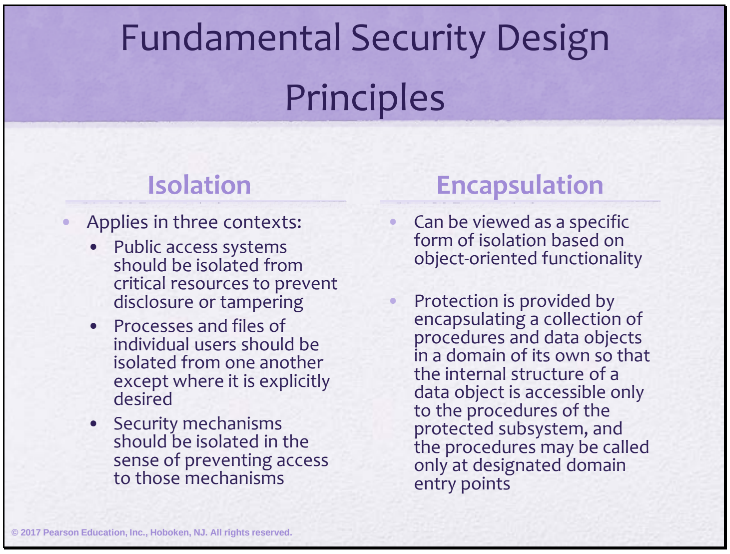
Slide 28



Least common mechanism means that the design should minimize the functions
shared by different users, providing mutual security. This principle helps reduce the number of unintended communication paths and reduces the amount of
hardware and software on which all users depend, thus making it easier to verify if
there are any undesirable security implications.

Psychological acceptability implies that the security mechanisms should not interfere unduly with the work of users, while at the same time meeting the needs of
those who authorize access. If security mechanisms hinder the usability or accessibility
of resources, then users may opt to turn off those mechanisms. Where possible,
security mechanisms should be transparent to the users of the system or at most
introduce minimal obstruction. In addition to not being intrusive or burdensome,
security procedures must reflect the user's mental model of protection. If the protection
procedures do not make sense to the user or if the user must translate his image
of protection into a substantially different protocol, the user is likely to make errors.

Slide 29

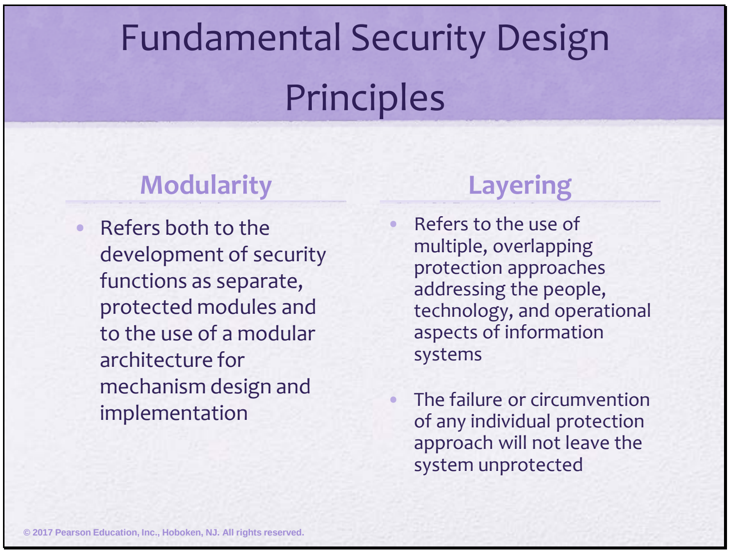


Isolation is a principle that applies in three contexts. First, public access systems
should be isolated from critical resources (data, processes, etc.) to prevent disclosure
or tampering. In cases where the sensitivity or criticality of the information
is high, organizations may want to limit the number of systems on which that data is
stored and isolate them, either physically or logically. Physical isolation may include
ensuring that no physical connection exists between an organization's public access
information resources and an organization's critical information. When implementing
logical isolation solutions, layers of security services and mechanisms should be
established between public systems and secure systems responsible for protecting
critical resources. Second, the processes and files of individual users should be isolated
from one another except where it is explicitly desired. All modern operating
systems provide facilities for such isolation, so that individual users have separate,
isolated process space, memory space, and file space, with protections for preventing
unauthorized access. And finally, security mechanisms should be isolated in the

sense of preventing access to those mechanisms. For example, logical access control
may provide a means of isolating cryptographic software from other parts of the
host system and for protecting cryptographic software from tampering and the keys
from replacement or disclosure.

Encapsulation can be viewed as a specific form of isolation based on object-oriented
functionality. Protection is provided by encapsulating a collection of procedures
and data objects in a domain of its own so that the internal structure of a
data object is accessible only to the procedures of the protected subsystem, and the
procedures may be called only at designated domain entry points.

Slide 30

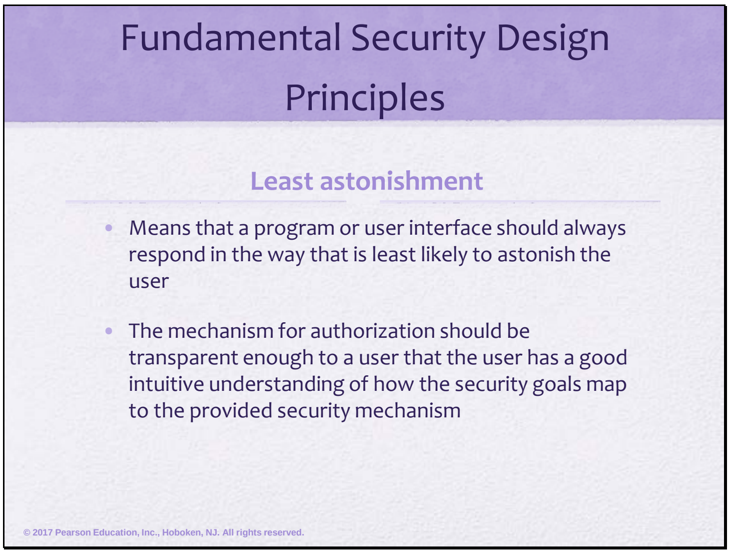


Modularity in the context of security refers both to the development of security functions as separate, protected modules and to the use of a modular architecture for mechanism design and implementation. With respect to the use of separate security modules, the design goal here is to provide common security functions and services, such as cryptographic functions, as common modules. For example, numerous protocols and applications make use of cryptographic functions. Rather than implementing such functions in each protocol or application, a more secure design is provided by developing a common cryptographic module that can be invoked by numerous protocols and applications. The design and implementation effort can then focus on the secure design and implementation of a single cryptographic module and including mechanisms to protect the module from tampering. With respect to the use of a modular architecture, each security mechanism should be able to support migration to new technology or upgrade of new features without requiring an entire system

redesign. The security design should be modular so that individual parts of the security
design can be upgraded without the requirement to modify the entire system.

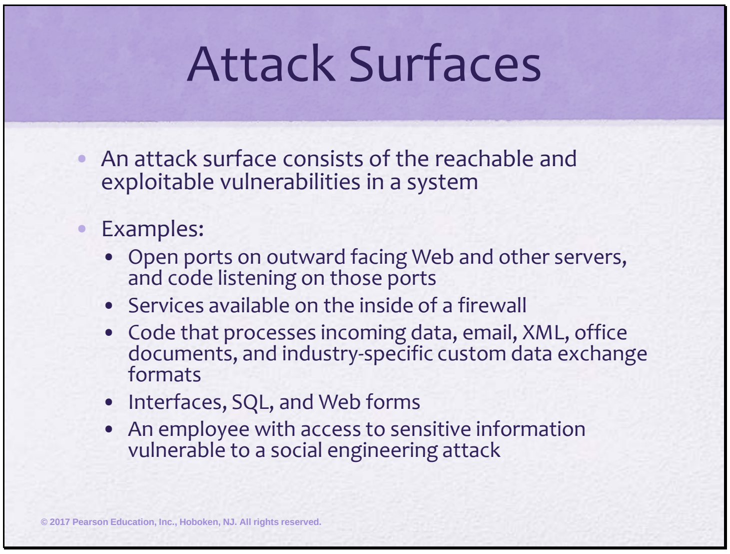
Layering refers to the use of multiple, overlapping protection approaches addressing the people, technology, and operational aspects of information systems.
By using multiple, overlapping protection approaches, the failure or circumvention
of any individual protection approach will not leave the system unprotected.
We will see throughout this book that a layering approach is often used to provide
multiple barriers between an adversary and protected information or services. This
technique is often referred to as defense in depth .

Slide 31



Least astonishment means that a program or user interface should always respond in the way that is least likely to astonish the user. For example, the mechanism
for authorization should be transparent enough to a user that the user has a good intuitive
understanding of how the security goals map to the provided security mechanism.

Slide 32



An attack surface consists of the reachable and exploitable vulnerabilities in a system [MANA11, HOWA03]. Examples of attack surfaces are the following:

■ Open ports on outward facing Web and other servers, and code listening on those ports

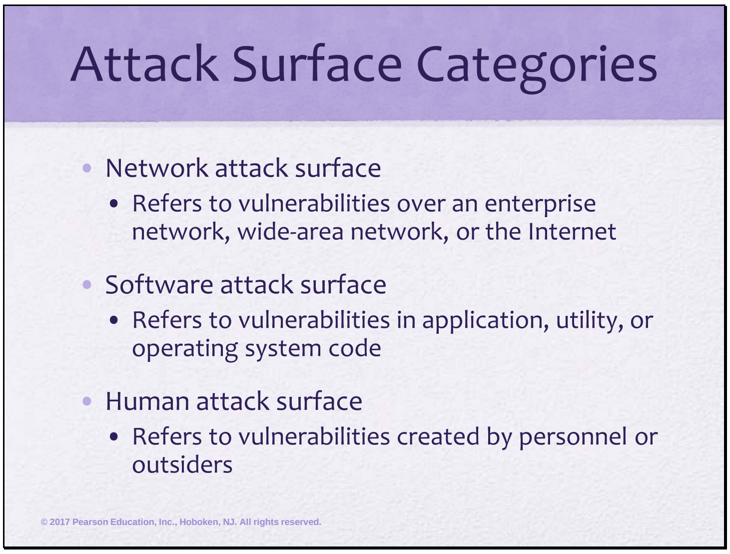
■ Services available on the inside of a firewall

■ Code that processes incoming data, email, XML, office documents, and industry-specific custom data exchange formats

■ Interfaces, SQL, and Web forms

■ An employee with access to sensitive information vulnerable to a social Engineering attack

Slide 33



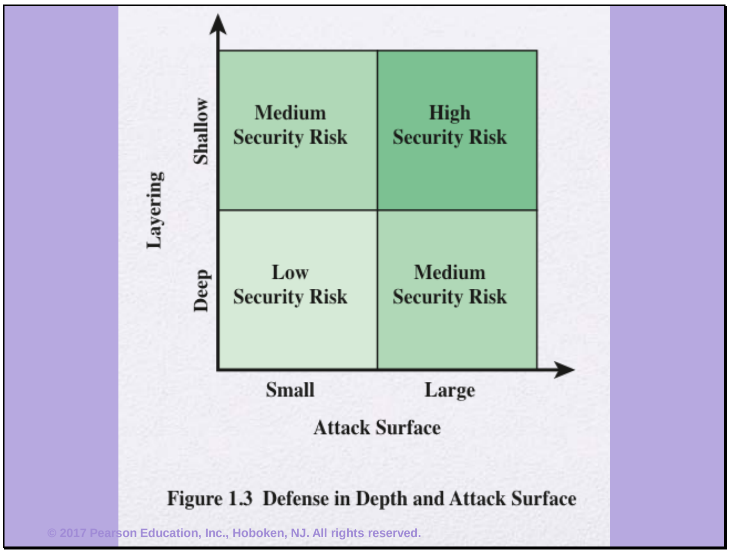
Attack surfaces can be categorized as follows:

■ 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.

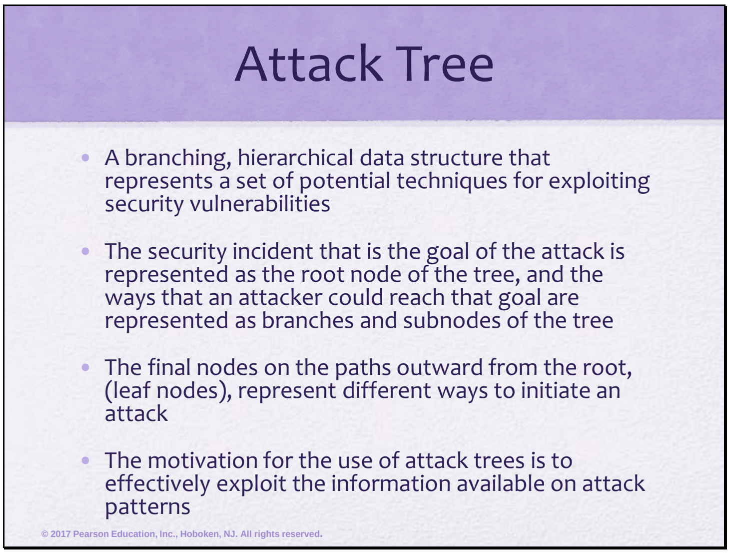
Slide 34



An attack surface analysis is a useful technique for assessing the scale and severity of threats to a system. A systematic analysis of points of vulnerability makes developers and security analysts aware of where security mechanisms are
required. Once an attack surface is defined, designers may be able to find ways to
make the surface smaller, thus making the task of the adversary more difficult. The
attack surface also provides guidance on setting priorities for testing, strengthening
security measures, and modifying the service or application.

As illustrated in Figure 1.3, the use of layering, or defense in depth, and attack surface reduction complement each other in mitigating security risk.

Slide 35



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 that 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. Each node other than a leaf is either an AND-node or an
OR-node. To achieve the goal represented by an AND-node, the subgoals represented
by all of that node’s subnodes must be achieved; and for an OR-node, at least one of
the subgoals must be achieved. Branches can be labeled with values representing difficulty,
cost, or other attack attributes, so that alternative attacks can be compared.

The motivation for the use of attack trees is to effectively exploit the information
available on attack patterns. Organizations such as CERT publish security

advisories that have enabled the development of a body of knowledge about both
general attack strategies and specific attack patterns. Security analysts can use the
attack tree to document security attacks in a structured form that reveals key vulnerabilities.
The attack tree can guide both the design of systems and applications,
and the choice and strength of countermeasures.

Slide 36

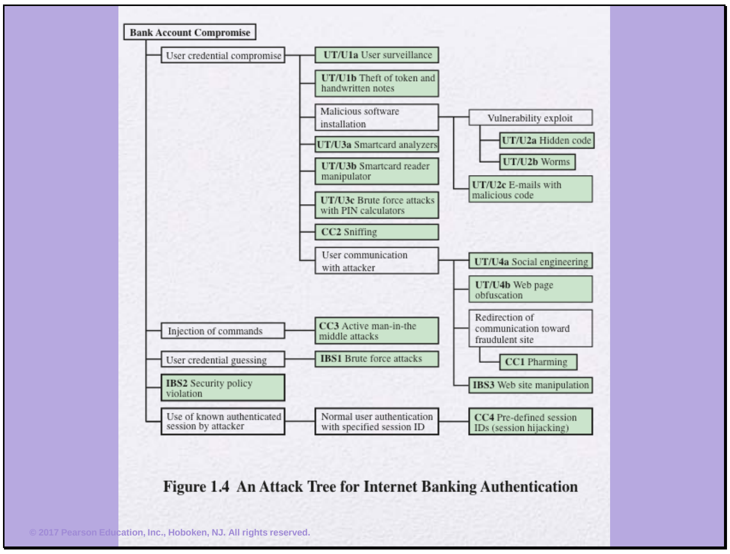


Figure 1.4 An Attack Tree for Internet Banking Authentication

Figure 1.4, based on a figure in [DIMI07], is an example of an attack tree analysis for an Internet banking authentication application. The root of the tree is
the objective of the attacker, which is to compromise a user’s account. The shaded
boxes on the tree are the leaf nodes, which represent events that comprise the
attacks. Note that in this tree, all the nodes other than leaf nodes are OR-nodes.
The analysis to generate this tree considered the three components involved in
authentication:

■ User terminal and user (UT/U): These attacks target the user equipment, including the tokens that may be involved, such as smartcards or other password
generators, as well as the actions of the user.

■ Communications channel (CC): This type of attack focuses on communication
links.

■ Internet banking server (IBS): These types of attacks are offline attacks against
the servers that host the Internet banking application.

Five overall attack strategies can be identified, each of which exploits one or more of the three components. The five strategies are as follows:

■ User credential compromise: This strategy can be used against many elements of the attack surface. There are procedural attacks, such as monitoring a user's action to observe a PIN or other credential, or theft of the user's token or handwritten notes. An adversary may also compromise token information using a variety of token attack tools, such as hacking the smartcard or using a brute force approach to guess the PIN. Another possible strategy is to embed malicious software to compromise the user's login and password. An adversary may also attempt to obtain credential information via the communication channel (sniffing). Finally, an adversary may use various means to engage in communication with the target user, as shown in Figure 1.4.

■ Injection of commands: In this type of attack, the attacker is able to intercept communication between the UT and the IBS. Various schemes can be used to be able to impersonate the valid user and so gain access to the banking system.

■ User credential guessing: It is reported in [HILT06] that brute force attacks against some banking authentication schemes are feasible by sending random usernames and passwords. The attack mechanism is based on distributed zombie personal computers, hosting automated programs for username- or password-based calculation.

■ Security policy violation: For example, violating the bank's security policy in combination with weak access control and logging mechanisms, an employee may cause an internal security incident and expose a customer's account.

■ Use of known authenticated session: This type of attack persuades or forces the user to connect to the IBS with a preset session ID. Once the user authenticates to the server, the attacker may utilize the known session ID to send packets to the IBS, spoofing the user's identity.

Figure 1.4 provides a thorough view of the different types of attacks on an

Internet banking authentication application. Using this tree as a starting point, security analysts can assess the risk of each attack and, using the design principles outlined in the preceding section, design a comprehensive security facility. [DIMO07] provides a good account of the results of this design effort.

Slide 37

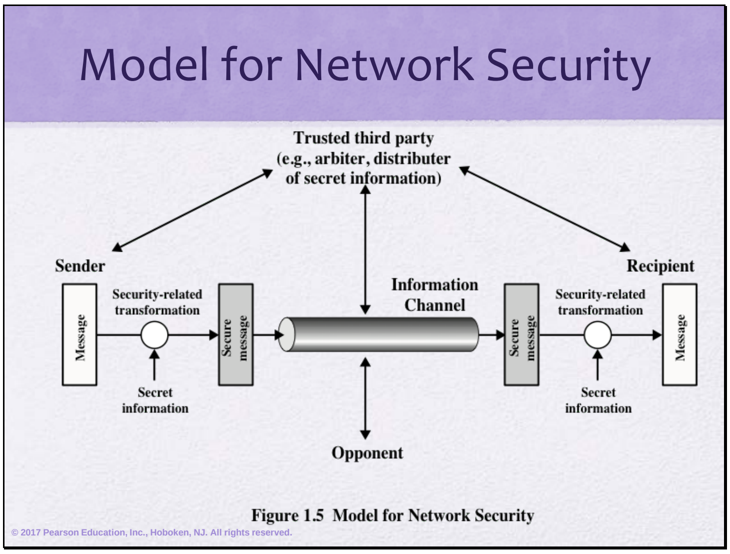


Figure 1.5 Model for Network Security

A model for much of what we will be discussing is captured, in very general terms, in
Figure 1.5. A message is to be transferred from one party to another across some sort
of Internet service. The two parties, who are the principals in this transaction, must
cooperate for the exchange to take place. A logical information channel is established
by defining a route through the Internet from source to destination and by the cooperative
use of communication protocols (e.g., TCP/IP) by the two principals.

Security aspects come into play when it is necessary or desirable to protect the
information transmission from an opponent who may present a threat to confidentiality,
authenticity, and so on. All the techniques for providing security have two components:

■ A security-related transformation on the information to be sent. Examples
include the encryption of the message, which scrambles the message so that it
is unreadable by the opponent, and the addition of a code based on the contents
of the message, which can be used to verify the identity of the sender.

■ Some secret information shared by the two principals and, it is hoped, Unknown to the opponent. An example is an encryption key used in conjunction with the transformation to scramble the message before transmission and unscramble it on reception.

A trusted third party may be needed to achieve secure transmission. For example, a third party may be responsible for distributing the secret information to the two principals while keeping it from any opponent. Or a third party may be needed to arbitrate disputes between the two principals concerning the authenticity of a message transmission.

This general model shows that there are four basic tasks in designing a particular security service:

1. Design an algorithm for performing the security-related transformation. The algorithm should be such that an opponent cannot defeat its purpose.

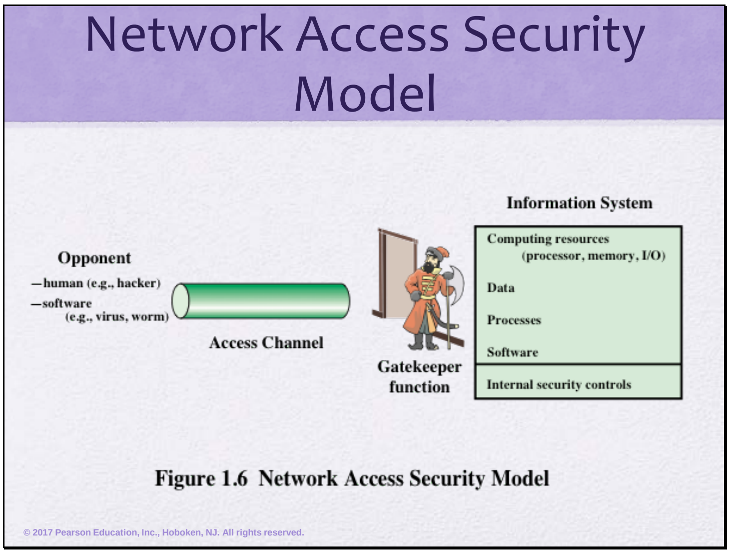
2. Generate the secret information to be used with the algorithm.

3. Develop methods for the distribution and sharing of the secret information.

- Specify a protocol to be used by the two principals that makes use of the

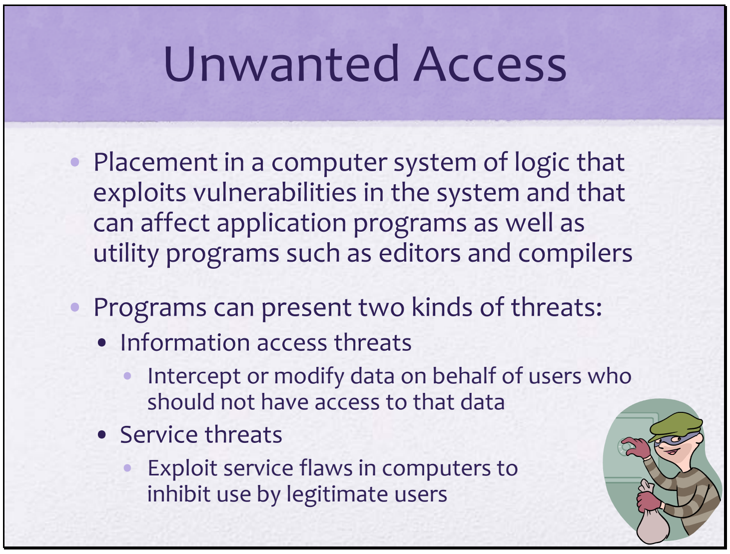
- security algorithm and the secret information to achieve a particular security service.

Slide 38



Parts One through Five of this book concentrate on the types of security mechanisms
and services that fit into the model shown in Figure 1.5. However, there are other security-related situations of interest that do not neatly fit this model but are considered in this book. A general model of these other situations is illustrated in Figure 1.6, which reflects a concern for protecting an information system from unwanted
access. Most readers are familiar with the concerns caused by the existence of hackers, who attempt to penetrate systems that can be accessed over a network. The hacker can be someone who, with no malign intent, simply gets satisfaction from breaking and entering a computer system. The intruder can be a disgruntled employee who wishes to do damage or a criminal who seeks to exploit computer assets for financial gain (e.g., obtaining credit card numbers or performing illegal money transfers).

Slide 39



Another type of unwanted access is the placement in a computer system of logic that exploits vulnerabilities in the system and that can affect application programs as well as utility programs, such as editors and compilers. Programs can present two kinds of threats:

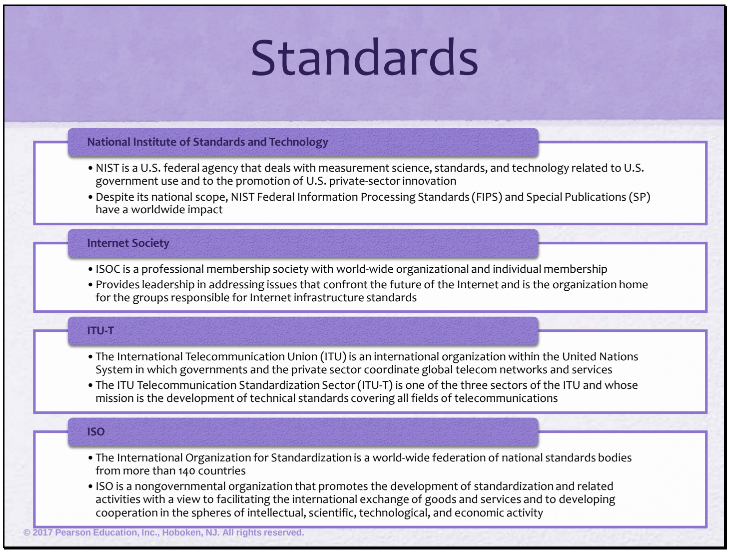
• Information access threats: Intercept or modify data on behalf of users who should not have access to that data.

• Service threats: Exploit service flaws in computers to inhibit use by legitimate users.

Viruses and worms are two examples of software attacks. Such attacks can be introduced into a system by means of a disk that contains the unwanted logic concealed in otherwise useful software. They can also be inserted into a system across a network; this latter mechanism is of more concern in network security.

The security mechanisms needed to cope with unwanted access fall into two broad categories (see Figure 1.6). The first category might be termed a gatekeeper function. It includes password-based login procedures that are designed to deny access to all but authorized users and screening logic that is designed to detect and reject worms, viruses, and other similar attacks. Once either an unwanted user or unwanted software gains access, the second line of defense consists of a variety of internal controls that monitor activity and analyze stored information in an attempt to detect the presence of unwanted intruders. These issues are explored in Part Six.

Slide 40



Many of the security techniques and applications described in this book have been
specified as standards. Additionally, standards have been developed to cover management
practices and the overall architecture of security mechanisms and services.
Throughout this book, we describe the most important standards in use or that are
being developed for various aspects of cryptography and network security. Various
organizations have been involved in the development or promotion of these standards.
The most important (in the current context) of these organizations are as follows:

■ 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.

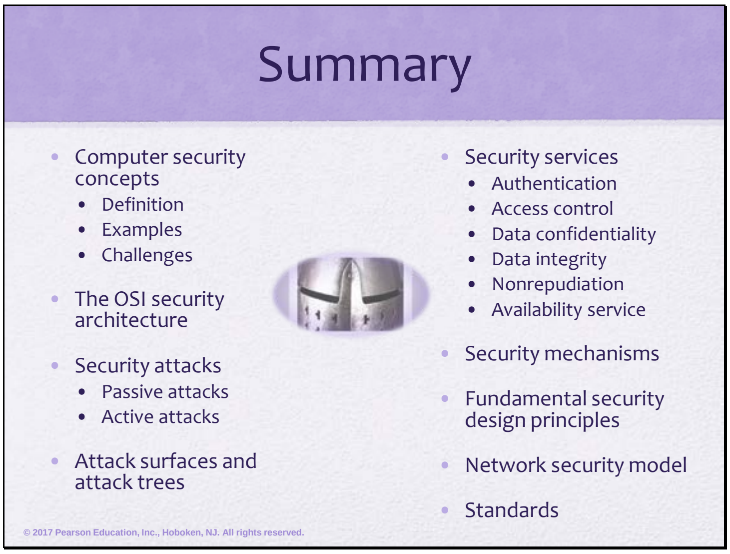
■ Internet Society: ISOC is a professional membership society with worldwide
organizational and individual membership. It provides leadership in
addressing issues that confront the future of the Internet and is the organization

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).

■ ITU-T: The International Telecommunication Union (ITU) is an international organization within the United Nations System 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 development of technical standards covering all fields of telecommunications. ITU-T standards are referred to as Recommendations.

■ ISO: The International Organization for Standardization (ISO) is a worldwide federation of national standards bodies from more than 140 countries, one from each country. 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.

Slide 41



Chapter 1 summary.