



**BITS Pilani**

Pilani | Dubai | Goa | Hyderabad

# Module 2-CS\_02

## L06 Security



SoftArch\_Session04.pdf

Harvinder S Jabbal  
SSZG653 Software Architectures



## Module 2-CS 02

### L06 Security

# Chapter Outline

---



What is Security?

Security General Scenario

Tactics for Security

A Design Checklist for Security

Summary

# What is Security?



Security is a measure of the system's ability to protect data and information from unauthorized access while still providing access to people and systems that are authorized.

An action taken against a computer system with the intention of doing harm is called an *attack* and can take a number of forms.

It may be an unauthorized attempt to access data or services or to modify data, or it may be intended to deny services to legitimate users.

# What is Security?



Security has three main characteristics, called CIA:

- Confidentiality is the property that data or services are protected from unauthorized access. For example, a hacker cannot access your income tax returns on a government computer.
- Integrity is the property that data or services are not subject to unauthorized manipulation. For example, your grade has not been changed since your instructor assigned it.
- Availability is the property that the system will be available for legitimate use. For example, a denial-of-service attack won't prevent you from ordering a book from an online bookstore.

Other characteristics that support CIA are

- Authentication verifies the identities of the parties to a transaction and checks if they are truly who they claim to be. For example, when you get an e-mail purporting to come from a bank, authentication guarantees that it actually comes from the bank.
- Nonrepudiation guarantees that the sender of a message cannot later deny having sent the message and that the recipient cannot deny having received the message. For example, you cannot deny ordering something from the Internet, or the merchant cannot disclaim getting your order.
- Authorization grants a user the privileges to perform a task. For example, an online banking system authorizes a legitimate user to access his account.

# Security General Scenario



Portion of Scenario	Possible Values
Source	Human or another system which may have been previously identified (either correctly or incorrectly) or may be currently unknown. A human attacker may be from outside the organization or from inside the organization.
Stimulus	Unauthorized attempt is made to display data, change or delete data, access system services, change the system's behavior, or reduce availability.
Artifact	System services; data within the system; a component or resources of the system; data produced or consumed by the system
Environment	The system is either online or offline, connected to or disconnected from a network, behind a firewall or open to a network, fully operational, partially operational, or not operational
Response	<p>Transactions are carried out in a fashion such that</p> <ul style="list-style-type: none"> <li>• data or services are protected from unauthorized access;</li> <li>• data or services are not being manipulated without authorization;</li> <li>• parties to a transaction are identified with assurance;</li> <li>• the parties to the transaction cannot repudiate their involvements;</li> <li>• the data, resources, and system services will be available for legitimate use.</li> </ul> <p>The system tracks activities within it by</p> <ul style="list-style-type: none"> <li>• recording access or modification,</li> <li>• recording attempts to access data, resources or services,</li> <li>• notifying appropriate entities (people or systems) when an apparent attack is occurring.</li> </ul>
Response Measure	<p>One or more of the following</p> <ul style="list-style-type: none"> <li>• how much of a system is compromised when a particular component or data value is compromised,</li> <li>• how much time passed before an attack was detected,</li> <li>• how many attacks were resisted,</li> <li>• how long does it take to recover from a successful attack,</li> <li>• how much data is vulnerable to a particular attack</li> </ul>

# Sample Concrete Security Scenario



---

A disgruntled employee from a remote location attempts to modify the pay rate table during normal operations. The system maintains an audit trail and the correct data is restored within a day.

# Goal of Security Tactics



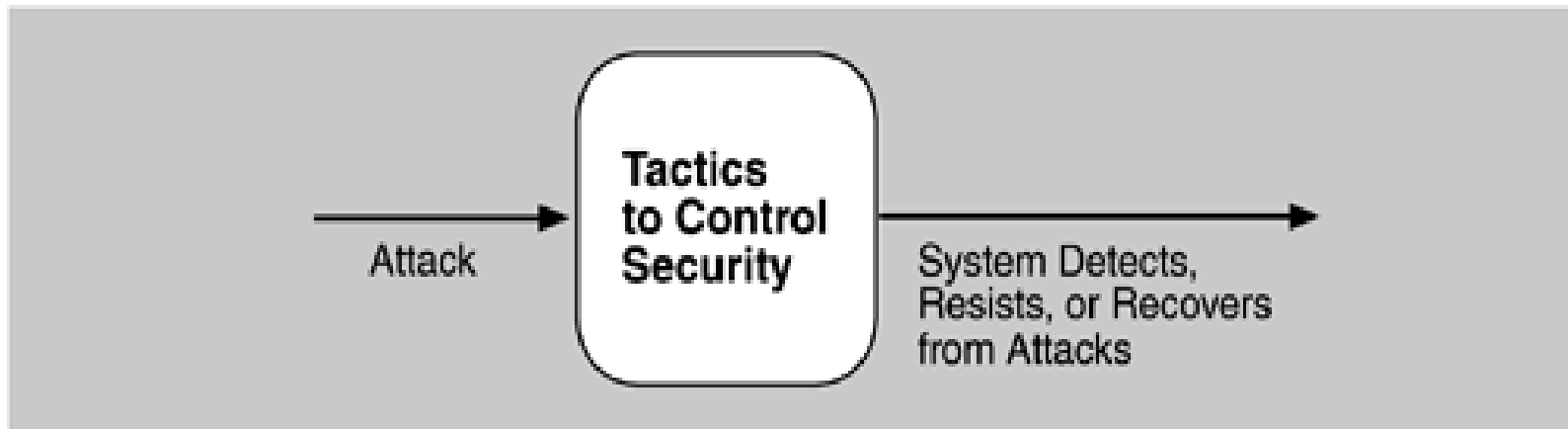
One method for thinking about system security is to think about physical security.

Secure installations have limited access to them (e.g., by using security checkpoints), have means of detecting intruders (e.g., by requiring legitimate visitors to wear badges), have deterrence mechanisms such as armed guards, have reaction mechanisms such as automatic locking of doors and have recovery mechanisms such as off-site back up.

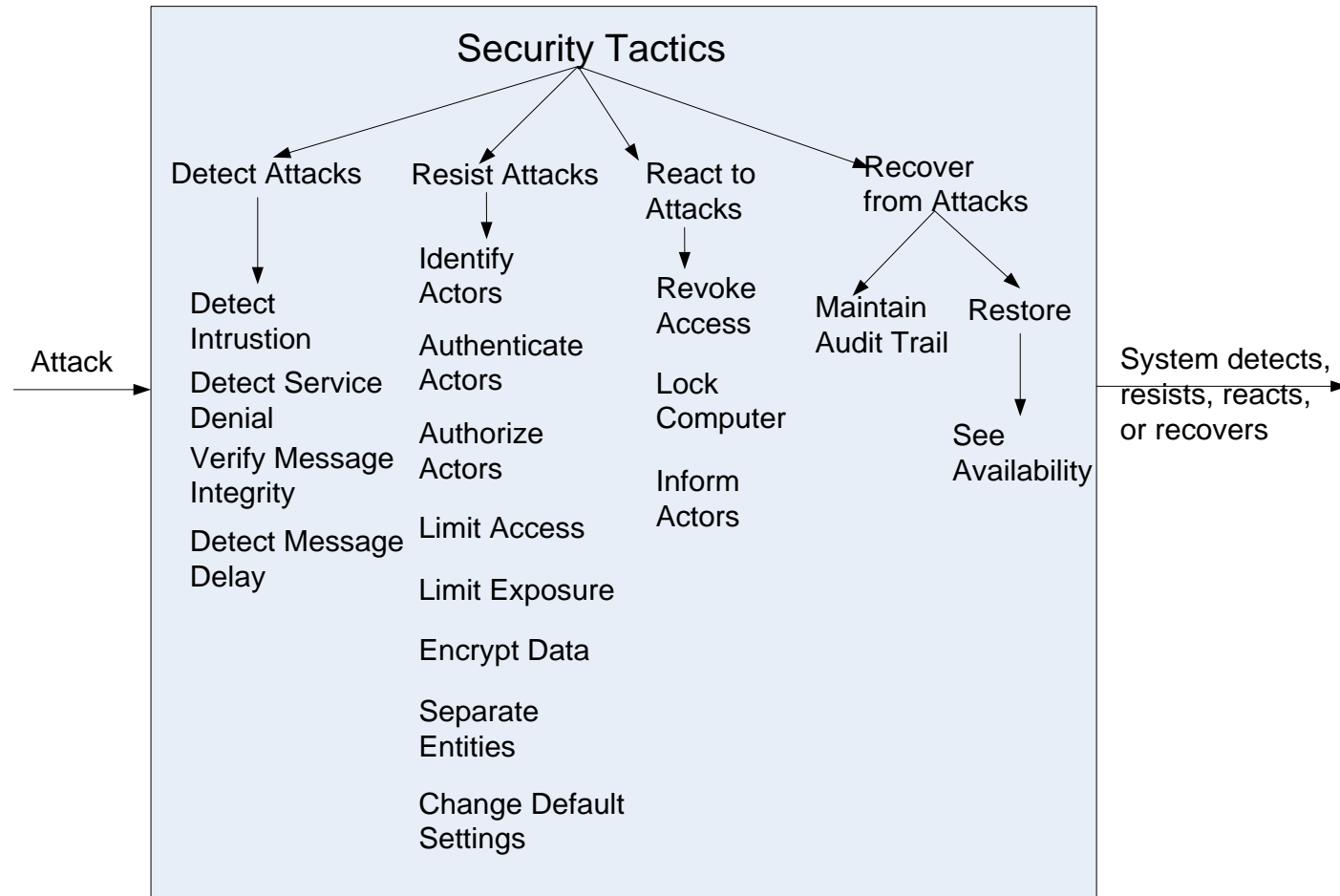
This leads to our four categories of tactics: detect, resist, react, and recover.



# Goal of Security Tactics



# Security Tactics



# Detect Attacks



Detect Intrusion: compare network traffic or service request patterns *within* a system to a set of signatures or known patterns of malicious behavior stored in a database.

Detect Service Denial: comparison of the pattern or signature of network traffic *coming into* a system to historic profiles of known Denial of Service (DoS) attacks.

Verify Message Integrity: use techniques such as checksums or hash values to verify the integrity of messages, resource files, deployment files, and configuration files.

Detect Message Delay: checking the time that it takes to

# Resist Attacks



Identify Actors: identify the source of any external input to the system.

Authenticate Actors: ensure that an actor (user or a remote computer) is actually who or what it purports to be.

Authorize Actors: ensuring that an authenticated actor has the rights to access and modify either data or services.

Limit Access: limiting access to resources such as memory, network connections, or access points.

# Resist Attacks



Limit Exposure: minimize the attack surface of a system by having the fewest possible number of access points.

Encrypt Data: apply some form of encryption to data and to communication.

Separate Entities: can be done through physical separation on different servers attached to different networks, the use of virtual machines, or an “air gap”.

Change Default Settings: Force the user to change settings assigned by default.

# React to Attacks



Revoke Access: limit access to sensitive resources, even for normally legitimate users and uses, if an attack is suspected.

Lock Computer: limit access to a resource if there are repeated failed attempts to access it.

Inform Actors: notify operators, other personnel, or cooperating systems when an attack is suspected or detected.

# Recover From Attacks



In addition to the Availability tactics for recovery of failed resources there is Audit.

Audit: keep a record of user and system actions and their effects, to help trace the actions of, and to identify, an attacker.

# Design Checklist for Security

Allocation of Responsibilities	<p>Determine which system responsibilities need to be secure. For each of these responsibilities ensure that additional responsibilities have been allocated to:</p> <ul style="list-style-type: none"><li>• identify the actor</li><li>• authenticate the actor</li><li>• authorize actors</li><li>• grant or deny access to data or services</li><li>• record attempts to access or modify data or services</li><li>• encrypt data</li><li>• recognize reduced availability for resources or services and inform appropriate personnel and restrict access</li><li>• recover from an attack</li><li>• verify checksums and hash values</li></ul>
--------------------------------	--



# Design Checklist for Security

<b>Coordination Model</b>	<p>Determine mechanisms required to communicate and coordinate with other systems or individuals. For these communications, ensure that mechanisms for authenticating and authorizing the actor or system, and encrypting data for transmission across the connection are in place.</p> <p>Ensure also that mechanisms exist for monitoring and recognizing unexpectedly high demands for resources or services as well as mechanisms for restricting or terminating the connection.</p>
---------------------------	--

# Design Checklist for Security

## Data Model

Determine the sensitivity of different data fields. For each data abstraction

- Ensure that data of different sensitivity is separated.
- Ensure that data of different sensitivity has different access rights and that access rights are checked prior to access.
- Ensure that access to sensitive data is logged and that the log file is suitably protected.
- Ensure that data is suitably encrypted and that keys are separated from the encrypted data.
- Ensure that data can be restored if it is inappropriately modified.

# Design Checklist for Security

## Mapping Among Architectural Elements

Determine how alternative mappings of architectural elements may change how an individual or system may read, write, or modify data, access system services or resources, or reduce their availability. Determine how alternative mappings may affect the recording of access to data, services or resources and the recognition of high demands for resources.

For each such mapping, ensure that there are responsibilities to

- identify an actor
- authenticate an actor
- authorize actors
- grant or deny access to data or services
- record attempts to access or modify data or services
- encrypt data
- recognize reduced availability for resources or services, inform appropriate personnel, and restrict access
- recover from an attack

# Design Checklist for Security

## Resource Management

Determine the system resources required to identify and monitor a system or an individual who is internal or external, authorized or not authorized, with access to specific resources or all resources.

Determine the resources required to authenticate the actor, grant or deny access to data or resources, notify appropriate entities, record attempts to access data or resources, encrypt data, recognize high demand for resources, inform users or systems, and restrict access.

For these resources consider whether an external entity can access or exhaust a critical resource; how to monitor the resource; how to manage resource utilization; how to log resource utilization and ensure that there are sufficient resources to perform necessary security operations.

Ensure that a contaminated element can be prevented from contaminating other elements.

Ensure that shared resources are not used for passing sensitive data from an actor with access rights to that data to an actor without access rights.

# Design Checklist for Security

<b>Binding Time</b>	<p>Determine cases where an instance of a late bound component may be untrusted.</p> <p>For such cases ensure that late bound components can be qualified, that is, if ownership certificates for late bound components are required, there are appropriate mechanisms to manage and validate them; that access to late bound data and services can be managed; that access by late bound components to data and services can be blocked; that mechanisms to record the access, modification, and attempts to access data or services by late bound components are in place; and that system data is encrypted where the keys are intentionally withheld for late bound components</p>
---------------------	--

# Design Checklist for Security

## Choice of Technology

Determine what technologies are available to help user authentication, data access rights, resource protection, data encryption.

Ensure that your chosen technologies support the tactics relevant for your security needs.

# Summary



Attacks against a system can be characterized as attacks against the confidentiality, integrity, or availability of a system or its data.

This leads to many of the tactics used to achieve security. Identifying, authenticating, and authorizing actors are tactics intended to determine which users or systems are entitled to what kind of access to a system.

No security tactic is foolproof and systems *will* be compromised. Hence, tactics exist to detect an attack, limit the spread of any attack, and to react and recover from an attack.