



**Secure Software Design and Engineering
(CY-321)**

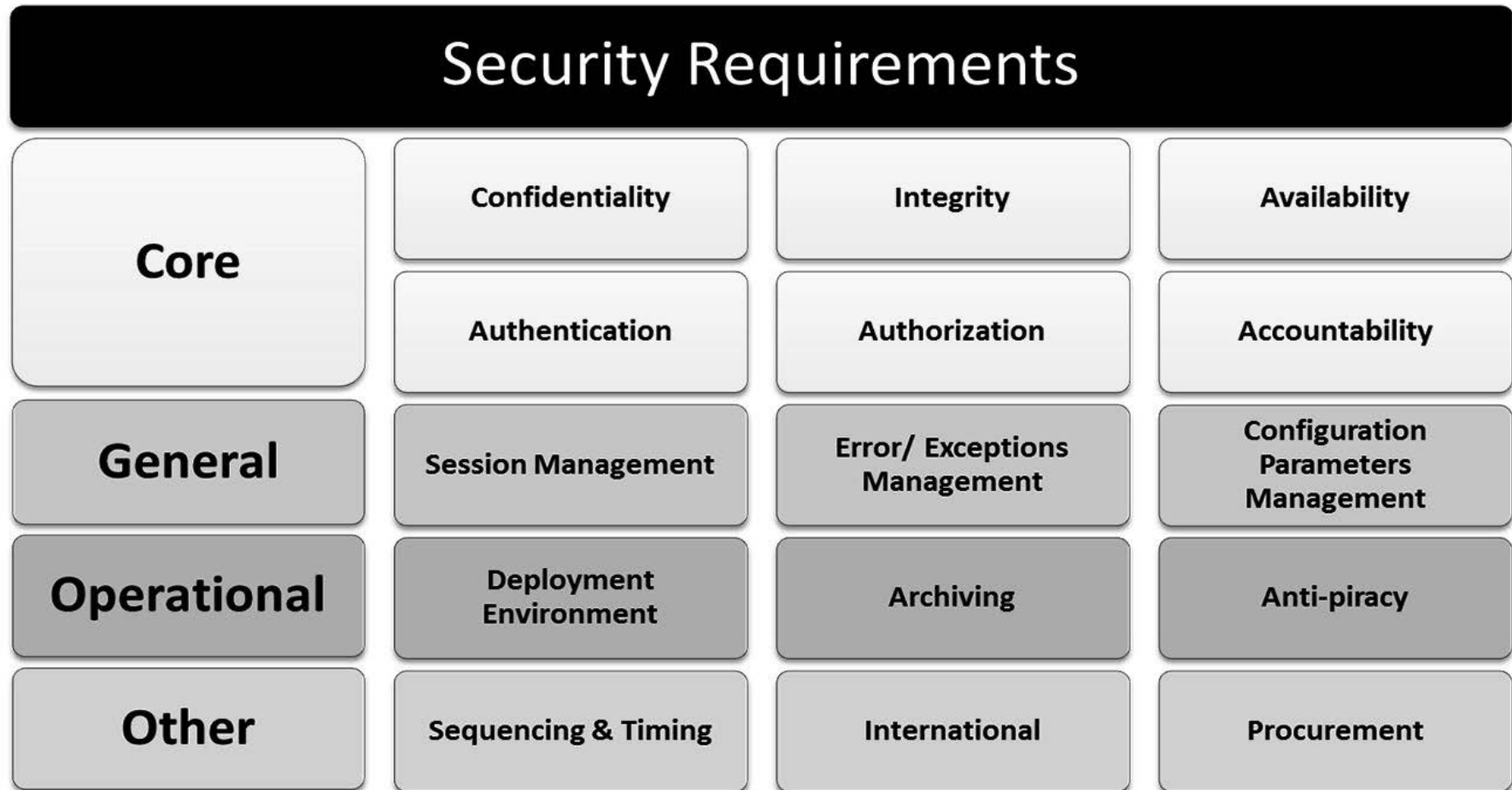
Security in SDLC (Planning and Requirement Phase)

Dr. Zubair Ahmad

- **Attendance?**

- Active Attendance
- **Dead Bodies.**
- **Active Minds**
- Mobiles in hands -> Mark as absent
- 80% mandatory

Planning and Requirement Stage



Core Requirements (1)

Authorization Requirements

Layered upon authentication, authorization requirements are those that confirm that an authenticated entity has the needed rights and privileges to access and perform actions on a requested resource

- CRUD operations, which stand for **Create, Read, Update or Delete** data

Core Requirements (2)

Authorization Requirements

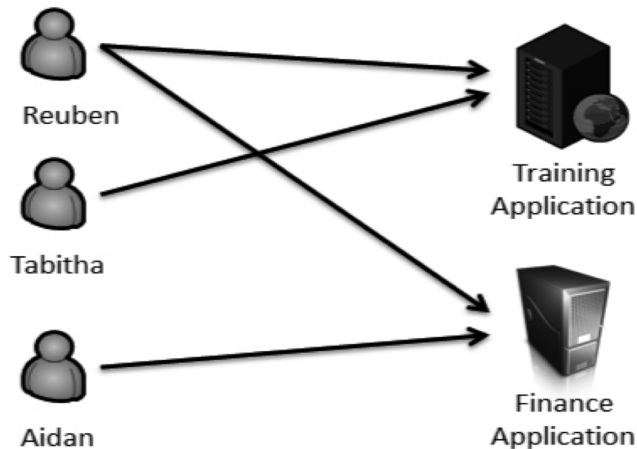
Access control models are primarily of the following types:

- Discretionary Access Control (DAC)
- Non-Discretionary Access Control (NDAC)
- Mandatory Access Control (MAC)
- Role-Based Access Control (RBAC)
- Resource-Based Access Control

Authorization Requirements (1)

Discretionary Access Control (DAC)

- Means of restricting access to objects based on the identity of subjects and/or groups to which they belong
- Subject with a certain access permission is capable of passing that permission (perhaps indirectly) on to any other subject



User	Training Application	Finance Application
Reuben	Yes	Yes
Tabitha	Yes	No
Aidan	No	Yes

Authorization Requirements (2)

Non-Discretionary Access Control (DAC)

- System enforcing the security policies
- Does not rely on the subject compliance with security policies. it is unavoidably imposed on all subjects
- The system security policies and mechanisms configured by the systems or security administrators are enforced and tamperproof

Authorization Requirements (3)

Mandatory Access Control (MAC)

- Access to objects is restricted to subjects based on the sensitivity of the information contained in the object
- The sensitivity is represented by a label
- MAC requires sensitivity labels for all the objects and clearance levels for all subjects and access is determined based on matching a subject clearance level with the object sensitivity level

Rule-based access control.

the access decision is based on a list of rules that are created or authorized by system owners who specify the privileges (i.e., read, write, execute, etc.) that the subjects (users) have on the objects (resources).

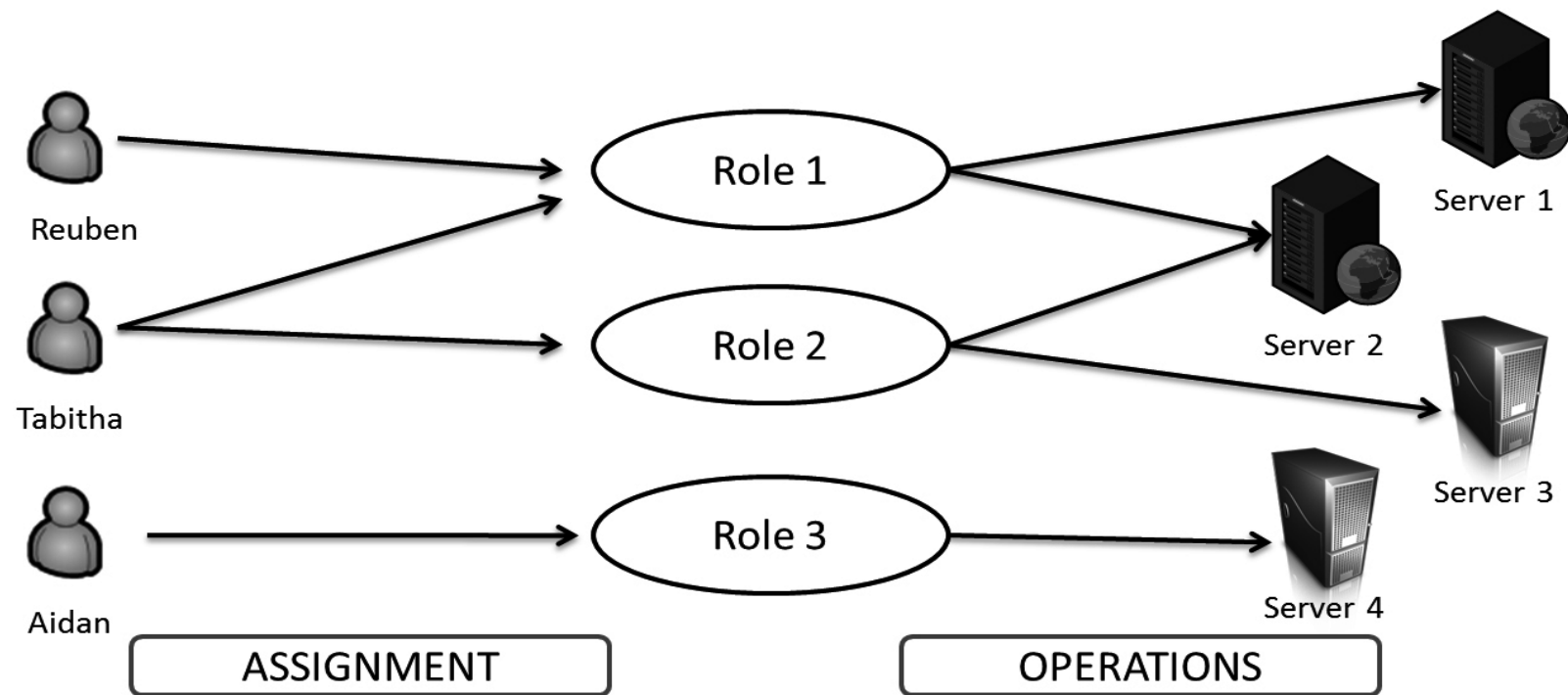
Authorization Requirements (4)

Role-Based Access Control (RBAC)

- Individuals (subjects) have access to a resource (object) based on their assigned role
- Roles are defined by job function which can be used for authorization decisions.
- Roles define the trust levels of entities to perform desired operations. These roles may be user roles or service roles.

Authorization Requirements (5)

Role-Based Access Control (RBAC)



Authorization Requirements (6)

RBAC in Relation to Least Privilege and Separation of Duties

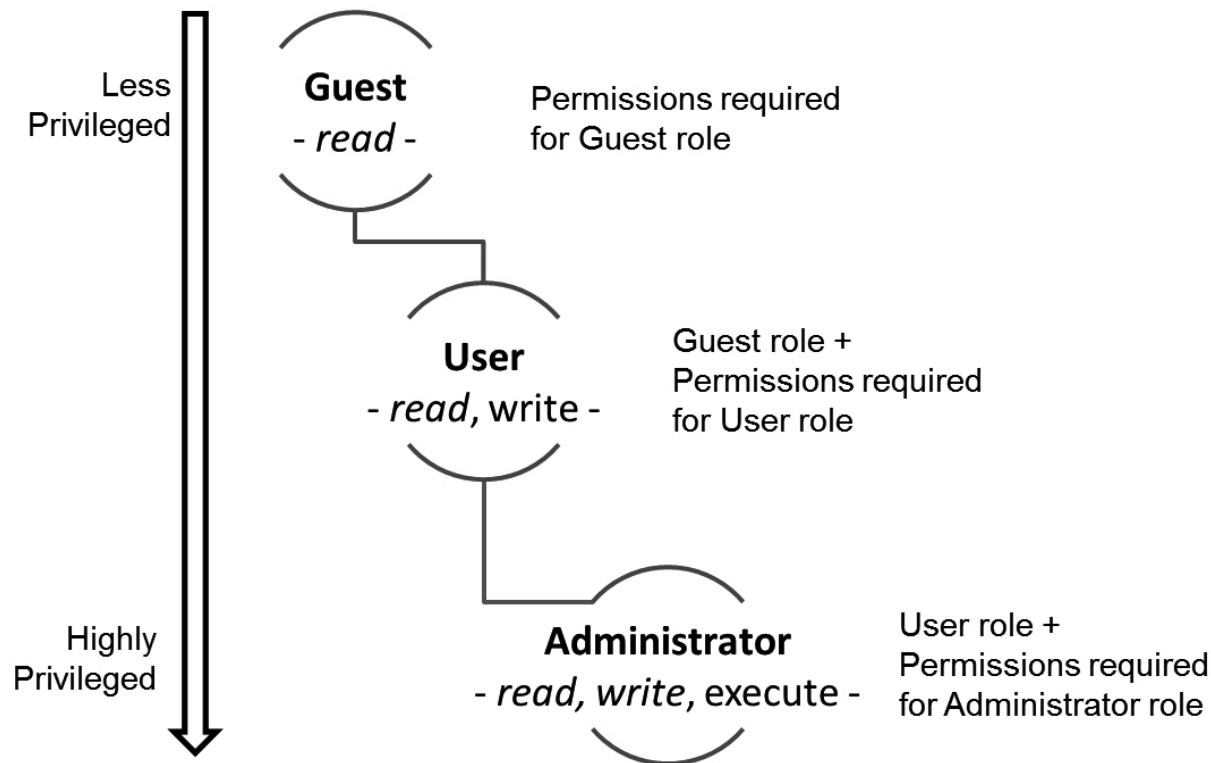
- Roles support the principle of least privilege, since roles are given just the needed privileges to undertake an operation against a resource
- No individual can be assigned to two roles that are mutually exclusive in their permissions to perform operations

The real benefit of RBAC over other access control methods includes the following:

- Simplified subjects and objects access rights administration
- Ability to represent the organizational structure
- Force enterprise compliance with control policies more easily and effectively.

Authorization Requirements (7)

Role Hierarchies



Authorization Requirements (8)

Resource-Based Access Control

- When the list of all users of your software are not known in advance, access can also be granted based on the resources
 - Impersonation and Delegation Model
 - Trusted Subsystem Model

Authorization Requirements (9)

Impersonation and Delegation Model

- Allowing a secondary entity to act on one's behalf is the principle of delegation.
- The secondary entity is considered to impersonate the identity of the primary entity when the complete sets of permissions of the primary entity are assigned to it
- **Kerberos** uses the delegation and impersonation model where the user upon successful authentication is granted a Kerberos ticket and the ticket is delegated the privileges and rights (sets of permission

Trusted Subsystem Model

- Access request decisions are granted based on the identity of a resource that is trusted instead of user identities
- A user logs into their bank account using a web browser to transfer funds from one account to another. The web application identity calls the database to first authenticate the user supplied credentials. It is not the user identity that is checked but the web application identity that is trusted and that can invoke the call to the database

Authorization Requirements

To achieve authorization requirements:

“Access to highly sensitive secret files will be restricted to users with secret or top secret clearance levels only.”

“User should not be required to send their credentials each and every time once they have authenticated themselves successfully.”

“All unauthenticated users will inherit read-only permissions that are part of guest user role while authenticated users will default to having read and write permissions as part of the general user role. Only members of the administrator role will have all rights as a general user in addition to having permissions to execute operations.”

Accountability Requirements

Accountability Requirements

- The identity of the subject (user or process) performing an action (who)
- The action (what)
- The object on which the action was performed (where)
- The timestamp of the action (when)

Accountability Requirements

To achieve accountability requirements:

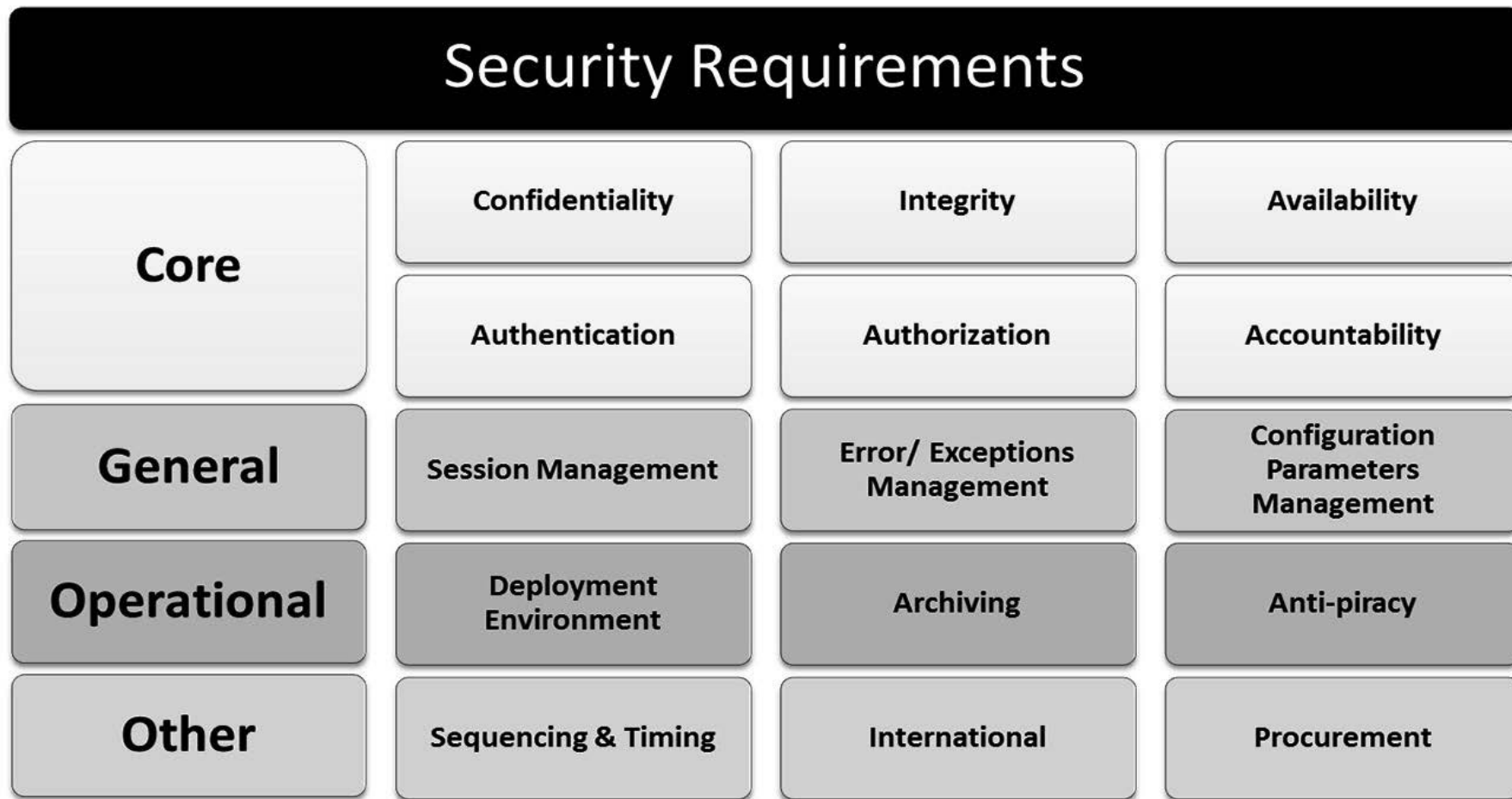
“All failed logon/in attempts will be logged along with the timestamp and the Internet Protocol address where the request originated.”

“A before and an after snapshot of the pricing data that changed when a user updates the pricing of a product must be tracked with the following auditable fields – identity, action, object and timestamp.”

“The audit logs must be securely retained for a period of 3 years.”

“Audit logs should always append and never be overwritten.”

Planning and Requirement Stage



General Requirements (1)

Session Management Requirements

Related to
which Secure
Design
concepts?

Why we need
session?

- Complete Mediation
- psychological acceptability

Since valid sessions can be potentially hijacked where an attacker takes control over an established session, it is necessary to plan for secure session management.

General Requirements (2)

“Each user activity will need to be uniquely tracked.”

“Session identifiers used to identify user sessions must not be passed in clear text or be easily guessable.”

Requirements For Secure Session Management

Sessions must be explicitly abandoned when the user logs off or closes the browser window

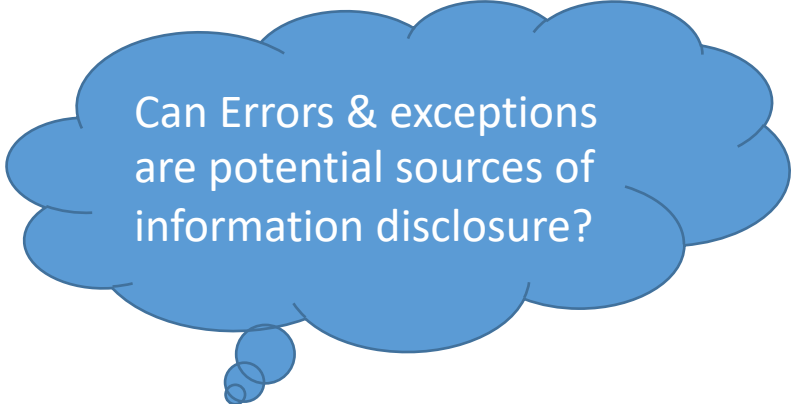
user should not be required to provide user credential once authenticated

General Requirements (3)



Errors & Exception Management Requirements

All about how you handle and structure the errors and exception



Can Errors & exceptions
are potential sources of
information disclosure?

General Requirements (4)

Errors & Exception Management Requirements

All exceptions are to be explicitly handled using try, catch and finally blocks

Security exception details are to be audited and monitored periodically.”

Error messages that are displayed to the end user will reveal only the needed information without disclosing any internal system error details

General Requirements (5)

Configuration Parameters Management Requirements

parameters and code usually need to be initialized before the software can run.

Identifying and capturing configuration settings is vital to ensure that an appropriate level of protection is considered when the software is designed, developed and more importantly when it is deployed.

General Requirements (6)

Passwords must not be hard-coded in line code

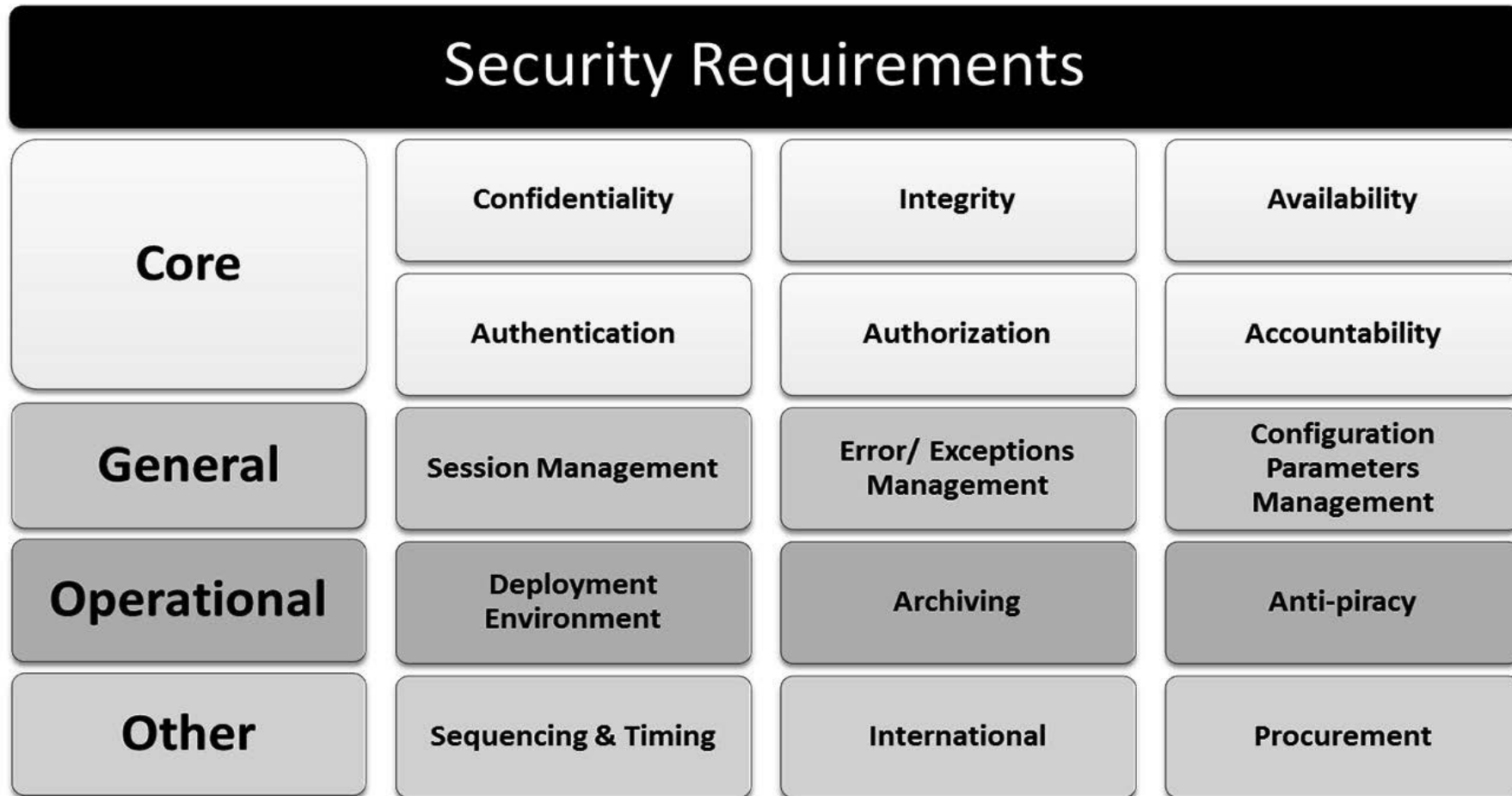
configuration file must encrypt sensitive database connections settings and other sensitive application settings."

Configuration Parameters Management Requirements

Initialization and disposal of global variables need to be carefully and explicitly monitored

Application and/or Session OnStart and OnEnd events must include protection of configuration information

Planning and Requirement Stage



Operational Requirements (1)

Operational Requirements

Concept of Operations
(CONOPS)

Operational Requirements (2)

What we need for Operational Requirements?

Incident management process should be followed to handle security incidents and root cause of the incidents must be identified

Data backups and replications must be protected in secure logs with least privilege implemented

“Discovered vulnerabilities in the software, that can impact the business and the brand, must be addressed and fixed as soon as possible, after being thoroughly tested in a simulated environment

The software must be continuously monitored to ensure that it is not susceptible to emerging threats.”

Operational Requirements (3)

Deployment Environment Requirements

Requirements about the environment in which the software will be deployed

Operational Requirements (3)

Deployment Environment Requirements

Will the software be deployed in an Internet, Extranet or intranet environment?

Can we leverage existing operating system event logging for auditing purposes?

Will the software be load balanced and how is clustering architected?

What privileges will be allowed in the production environment?

Will the software be transmitting sensitive or confidential information?

Will the software be hosted in a Demilitarized Zone (DMZ)?

Will the software need to support single sign-on (SSO) authentication?

What ports and protocols are available for use?

Will the software be deployed in a web farm environment?

Operational Requirements (4)

Archiving Requirements

the *location, duration* and *format* of archiving information must be determined

conflict between the organizational policy and a regulatory requirement, then follow?

Operational Requirements (5)

Archiving Requirements

Where will the data or information be stored?

How do we ensure that the media is not re-writable?

How long will we need to store the archives for?

Is there a regulatory requirement to store the data for a set period of time?

Is our archival retention policy contradictory to any compliance or regulatory requirements?

In what format will the data or information be stored? Clear text or cipher text?

If the data or information is stored in cipher text, how is this accomplished and are there management processes in place that will ensure proper retrieval?

How will these archives themselves be protected?

Operational Requirements (6)

Anti-Piracy Requirements

The software must be digitally signed to protect against tampering and reverse engineering

The code must be obfuscated, if feasible, to deter the duplication of code.

License keys must not be statically hard-coded in the software binaries as they can be disclosed by debugging and disassembly

License verification checks must be dynamic, preferably with phone-home mechanisms and not be dependent on factors that the end-user can change

Code obfuscation, code signing, anti-tampering, licensing and IP protection mechanisms should be included

Other Requirements (1)

Sequencing and Timing Requirements

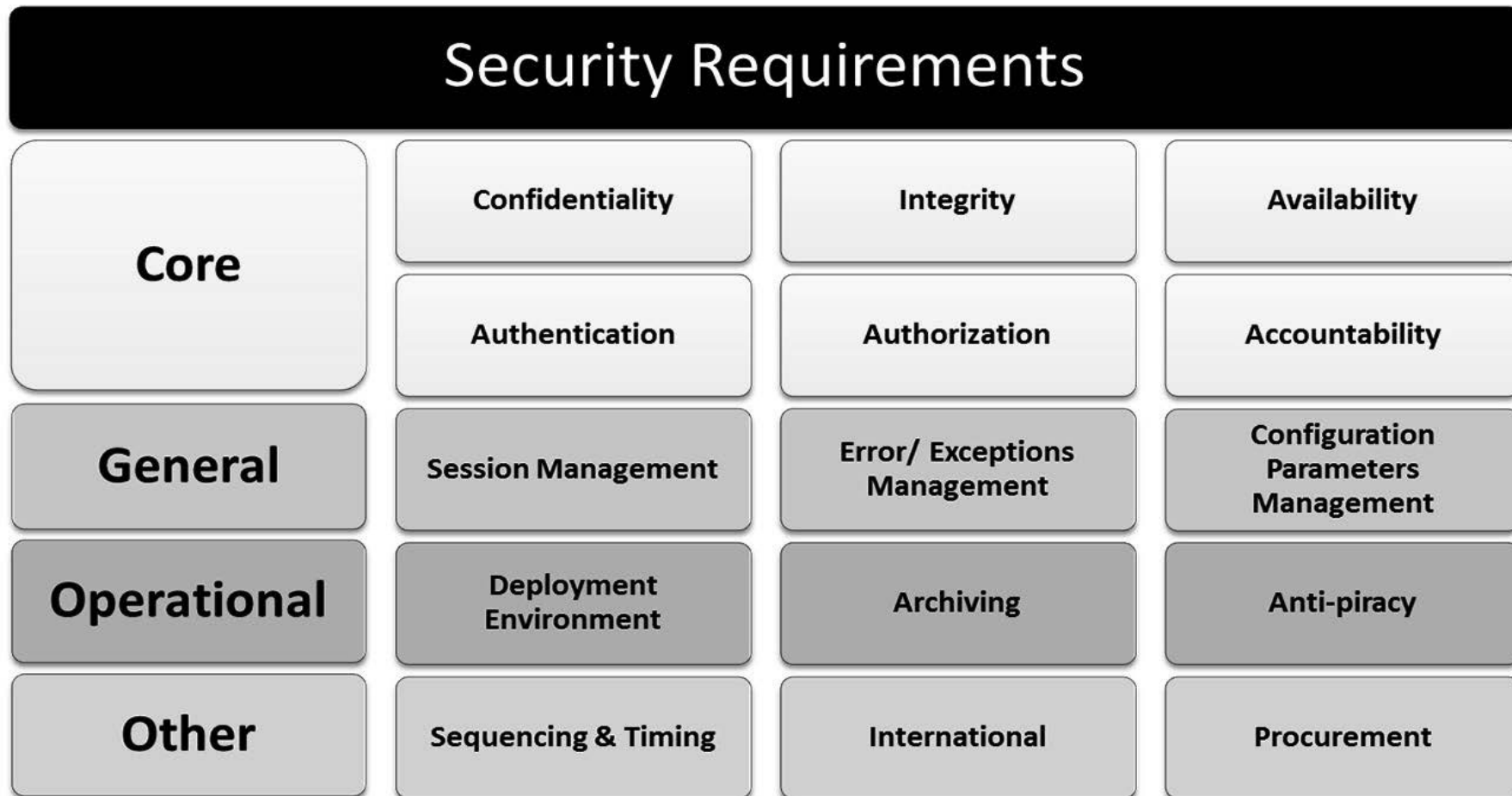
Undesirable sequence of events, where one event that is follow, in the program execution order attempts to supersedes its preceding event in its operations.

Multiple unsynchronized threads executing simultaneously for a process that needs to be completed atomically

Infinite loops that prevent a program from returning control to the normal flow of logic.

race conditions or
Time of Check/Time
of Use (TOC/TOU)
attacks.

Planning and Requirement Stage



Other Requirements (2)

International Requirements

No violation of any regulations

*legal and
technological.*

Character encoding and display direction

Software needs to support multi-lingual, multi-cultural and multi-regional needs

Other Requirements (3)

Procurement Requirements

secure software requirements must also be communicated and appropriately evaluated.

Additionally it is important to include software security requirements in legal protection mechanisms such as contracts and SLAs

Scenario:

A university is developing an **Online Exam System** for conducting remote assessments. The system allows students to log in, take exams, and receive scores automatically. However, during the system review, students must **identify missing security requirements** and suggest improvements.

Identified Security Features:

The initial design includes:

Authentication – Students log in using their university credentials.

Confidentiality – Exam questions are encrypted and stored securely.

Integrity – Answers are recorded without modification.

Session Management – Users are logged out after inactivity.

Identify at least 3 missing security requirements, impact and proposed Solution

Questions??

zubair.ahmad@giki.edu.pk

Office: G14 FCSE lobby