Class5: Security Engineering Chapter 13

Adeeb Noor, PhD
IT Department
Faculty of Computing and Information Technology
King Abdulaziz University
Fall 2025

Today lecture topic

Lecture 4

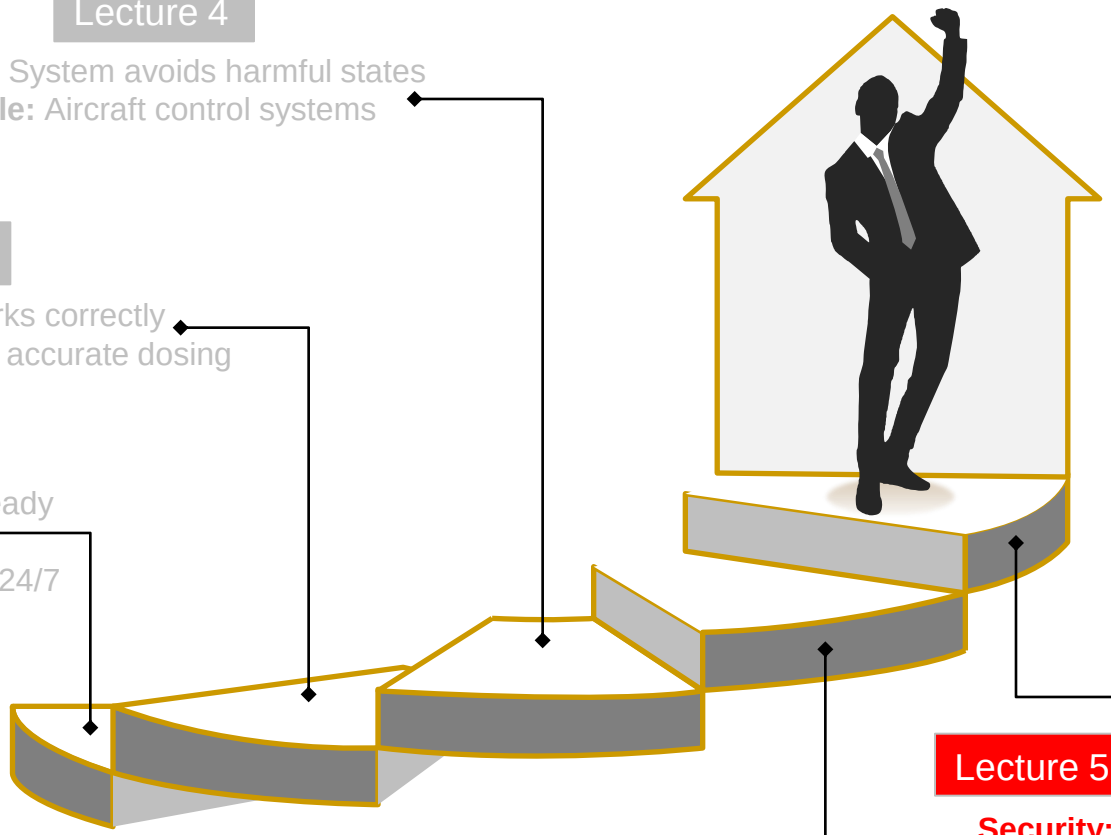
Safety: System avoids harmful states
Example: Aircraft control systems

Lecture 3

Reliability: System works correctly
Example: Insulin pump accurate dosing

Lecture 2

Availability: System is ready when needed
Example: ATM available 24/7



Lecture 6

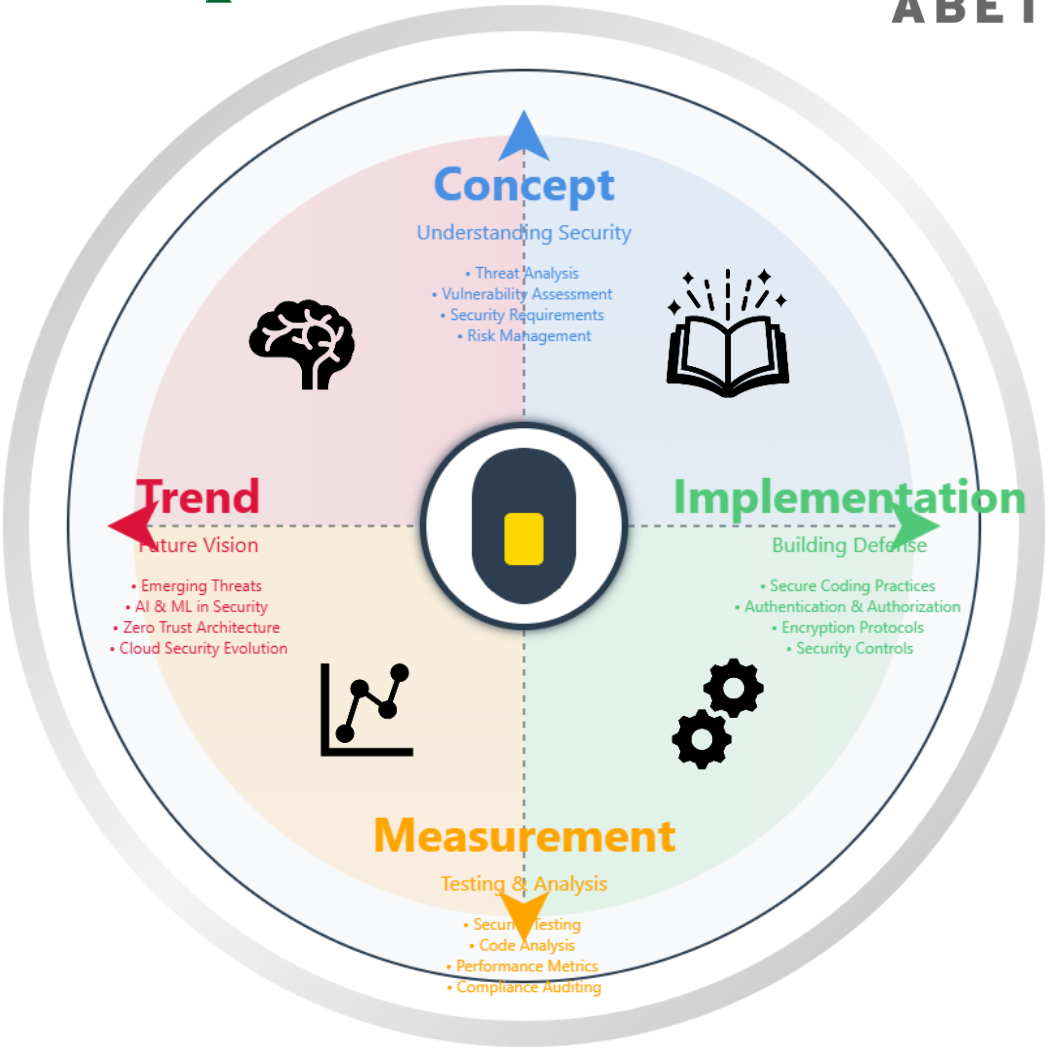
Resilience: System recovers from failures
Example: Backup power systems

Lecture 5

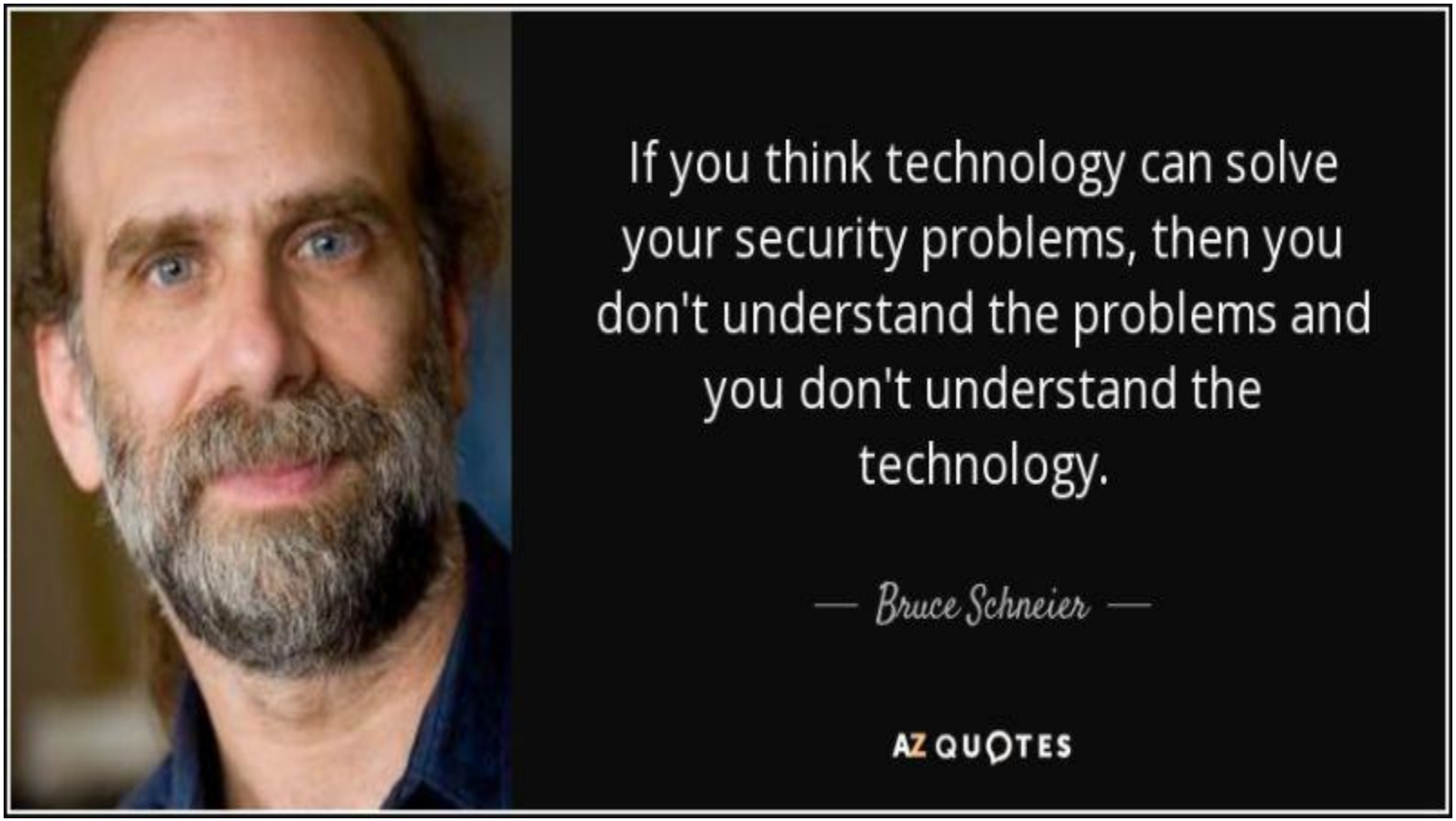
Security: System resists attacks
Example: Secure against fraud

The CIMT Compass

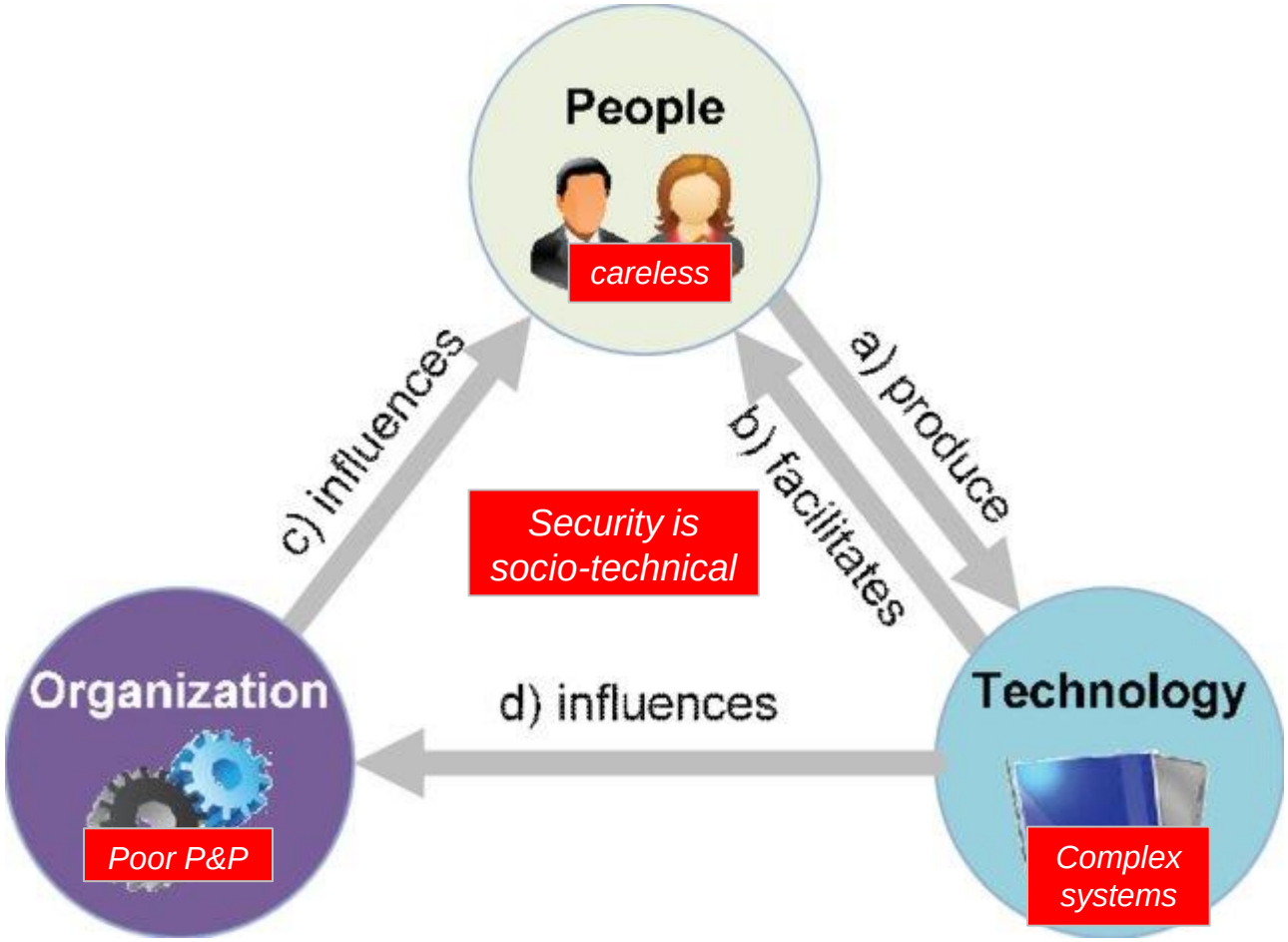
Where Academic Meets Business Reality



The Big Picture

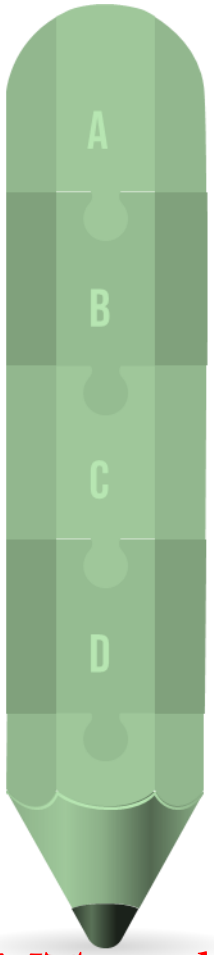


People are the weakest link



Take-home points

- The Key CLO: Design and test secure systems
- Security and dependability (13.1): Focus on confidentiality, integrity, and availability
- Security and organizations (13.2): Requires infrastructure, application, and operational security
- Security requirements (13.3): Define constraints the system must respect to maintain security
- Secure systems design (13.4): Must protect key assets and minimize potential losses from attacks
- Security testing (13.5): Focus on validating system can resist attackers
- LLMs enhance security engineering by automating hazard detection, documentation, and risk assessment to strengthen system security



CIMT model

Chapter Summary

Security engineering focuses on developing and maintaining systems that can protect against malicious attacks through systematic consideration of security requirements, architectural design, and comprehensive testing at all stages of development

Software Security: why matters

Security Critical Failure:

Unauthorized access exposed 50M+ customers' personal data through unsecured network access point

Root Causes:

Weak infrastructure security (13.2), missing security requirements (13.3), poor security design (13.4)

How to solve Causes:

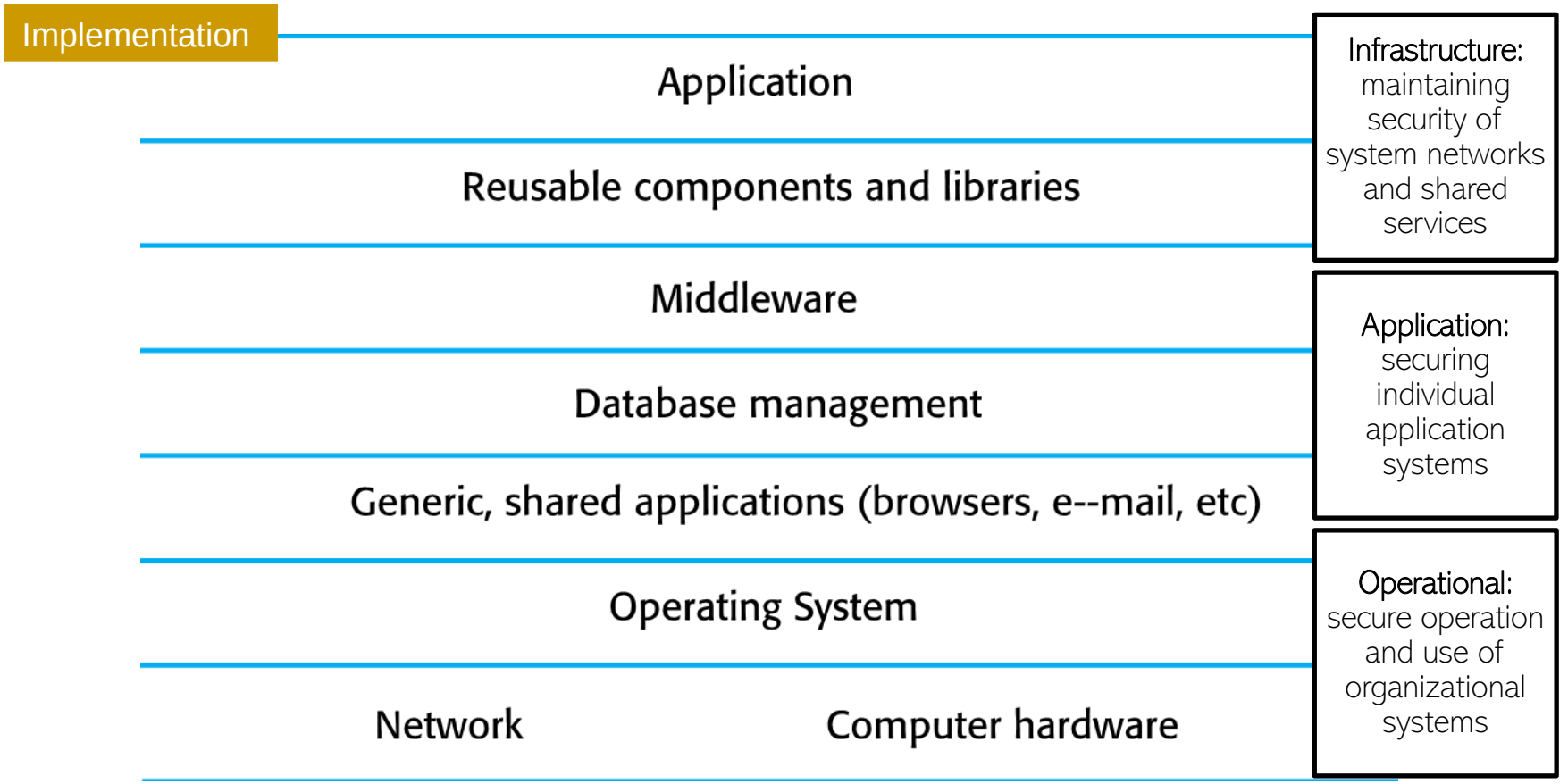
Apply CIA principles (13.1) through network segmentation, access control, monitoring, and comprehensive security testing (13.5)

13.1 Security & dependability

Concept

- What is it: System's ability to protect itself from external attacks
- Key Focus:
  - Protecting computer-based systems and data
  - Preventing unauthorized access and damage
- Essential for: Availability – Reliability – Safety
- Three security dimensions (CIA):
  - Confidentiality - Preventing unauthorized access
  - Integrity - Preventing unauthorized modification
  - Availability - Ensuring system access when needed

System layers where security may be compromised



Security terminology

Term	Definition
Asset	Something of value which has to be protected. The asset may be the software system itself or data used by that system.
Attack	An exploitation of a system's vulnerability. Generally, this is from outside the system and is a deliberate attempt to cause some damage.
Control	A protective measure that reduces a system's vulnerability. Encryption is an example of a control that reduces a vulnerability of a weak access control system
Exposure	Possible loss or harm to a computing system. This can be loss or damage to data, or can be a loss of time and effort if recovery is necessary after a security breach.
Threat	Circumstances that have potential to cause loss or harm. You can think of these as a system vulnerability that is subjected to an attack.
Vulnerability	A weakness in a computer-based system that may be exploited to cause loss or harm.

Security & dependability

Concept

Security is the foundation for reliable, available, safe, and resilient systems. Without it, everything can crumble

A. Security & Reliability

A hacker messes with a hospital's patient database. Now, doctors can't trust the information, leading to wrong treatments and potential harm

B. Security & Availability

Attackers flood a bank's website with requests, preventing customers from accessing their accounts online

C. Security & Safety

A malicious actor manipulates the software of a power plant, causing a malfunction that leads to a power outage or even an explosion

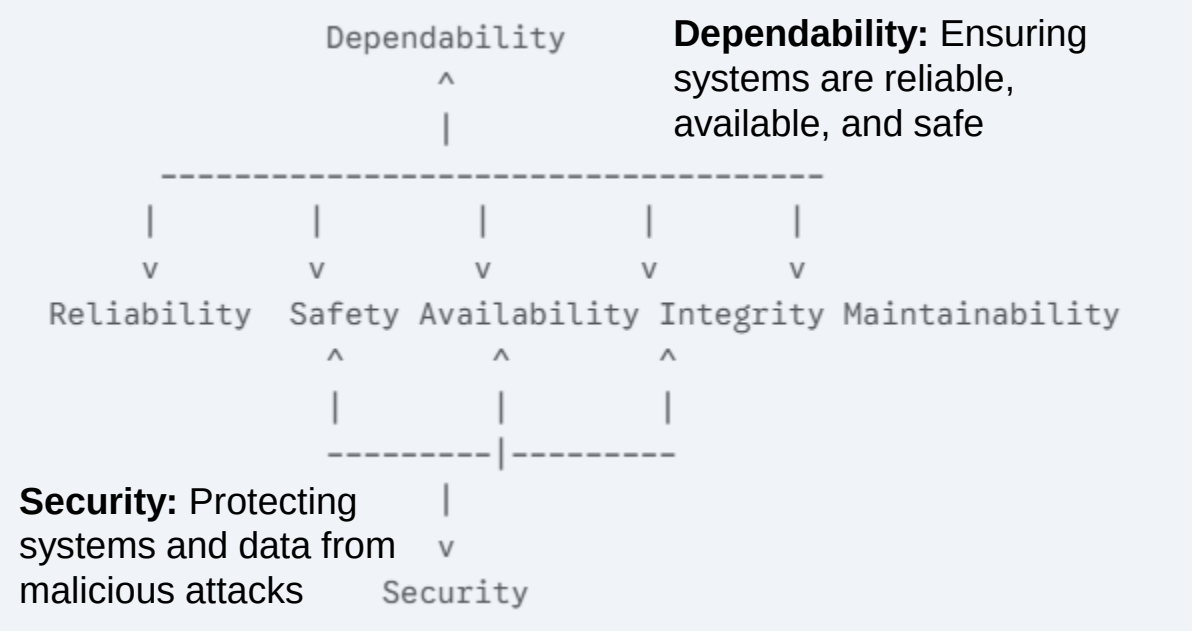
D. Security & Resilience

A company experiences ransomware attack. Their strong security and backup systems allow them to restore data and resume operations with minimal downtime

13.2 Security & organizations

Concept

- Security is an investment that requires a cost-effective approach
- Organizations need a clear security policy to guide decision-making



Dependability: Ensuring systems are reliable, available, and safe

Security: Protecting systems and data from malicious attacks

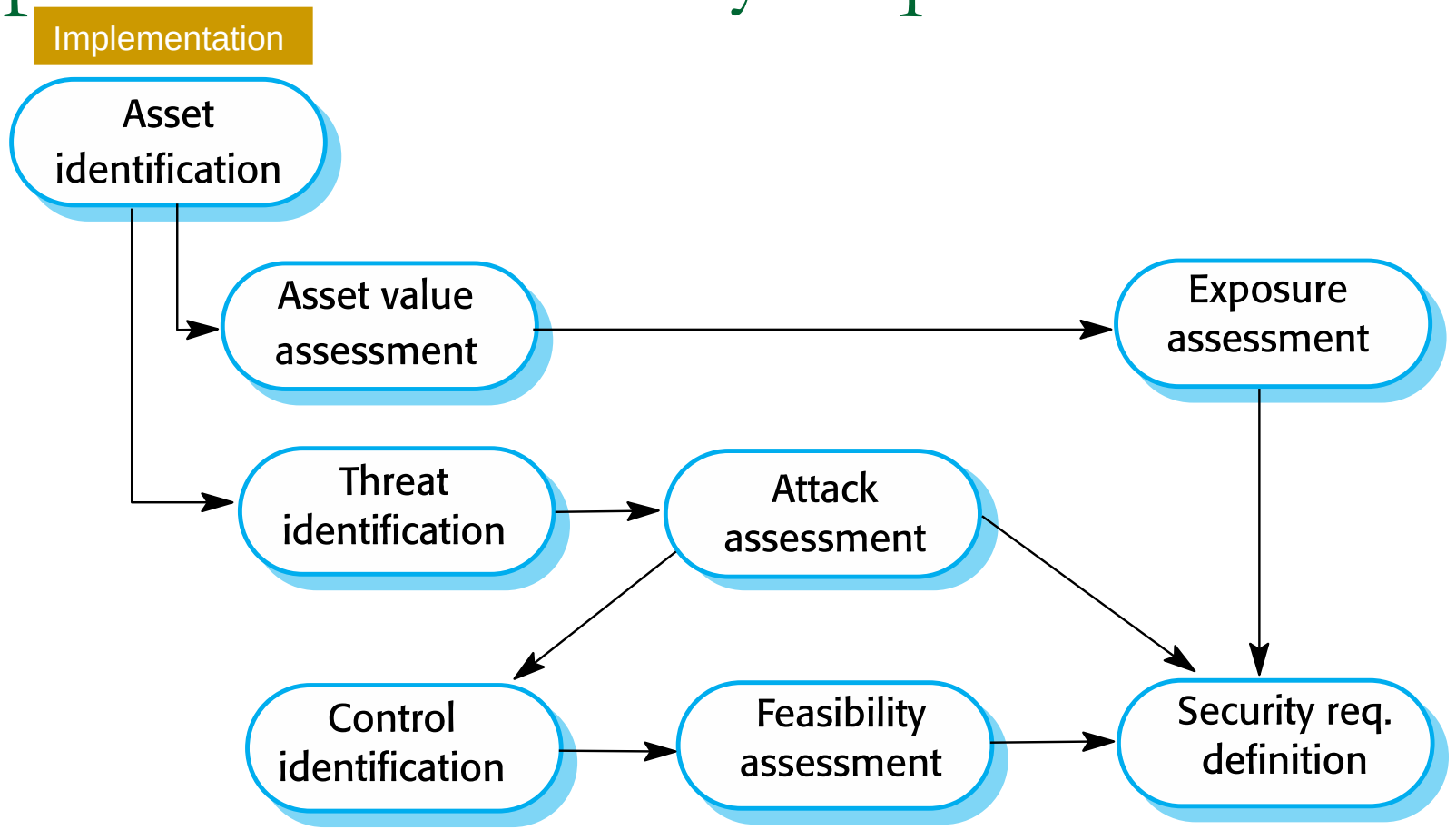
- Security is essential for dependability in networked systems
- Attacks can compromise reliability and safety by altering the system or its data

Risk assessment

Concept

- Security Policy: Organization-wide rules for protecting valuable assets
- Risk Management: Identifying and mitigating potential threats
- Types of Risk Assessment:
  - Preliminary: Initial assessment of generic risks
  - Design: Assessing risks during system development
  - Operational: Assessing risks during system use

Preliminary risk assessment process for security requirements



13.3 Security requirements: what needs to be protected

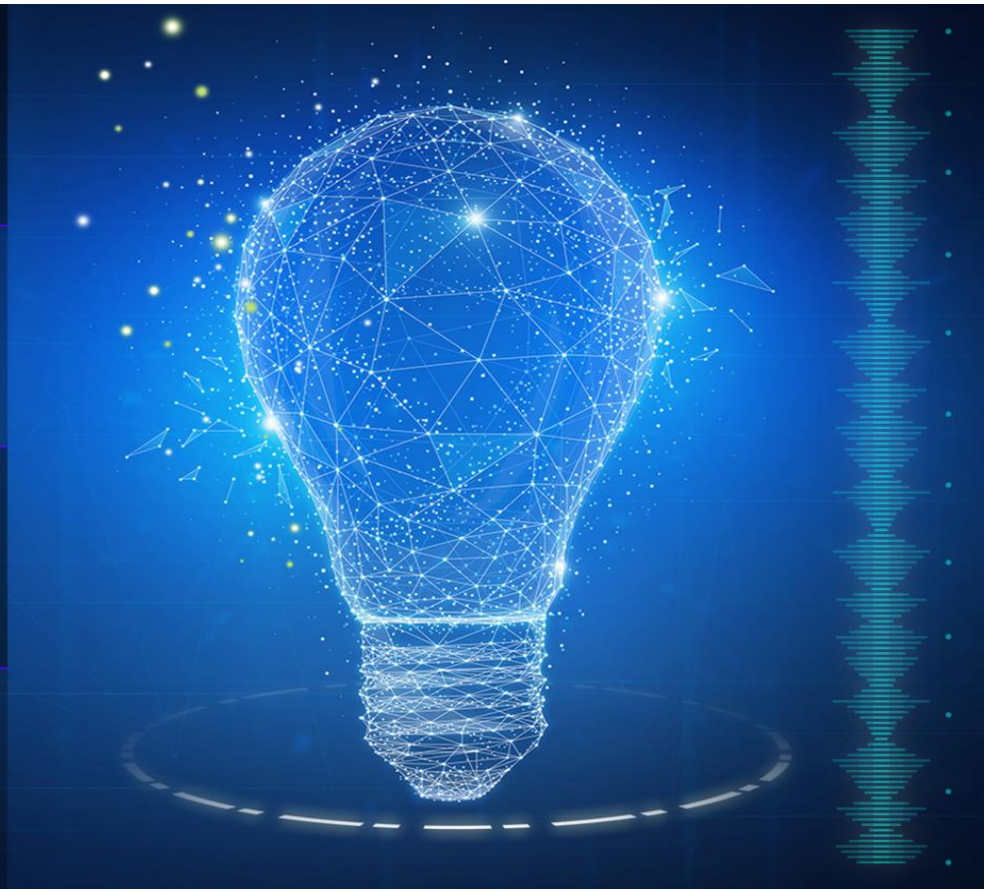
Implementation

Assets: Valuable resources that need protection (e.g., software, data)

Threats: Circumstances that can cause harm (e.g., unauthorized access)

Vulnerabilities: Weaknesses in a system that can be exploited (e.g., weak passwords)

Attacks: Exploiting a vulnerability to cause damage (e.g., stealing data)



Types of security requirements

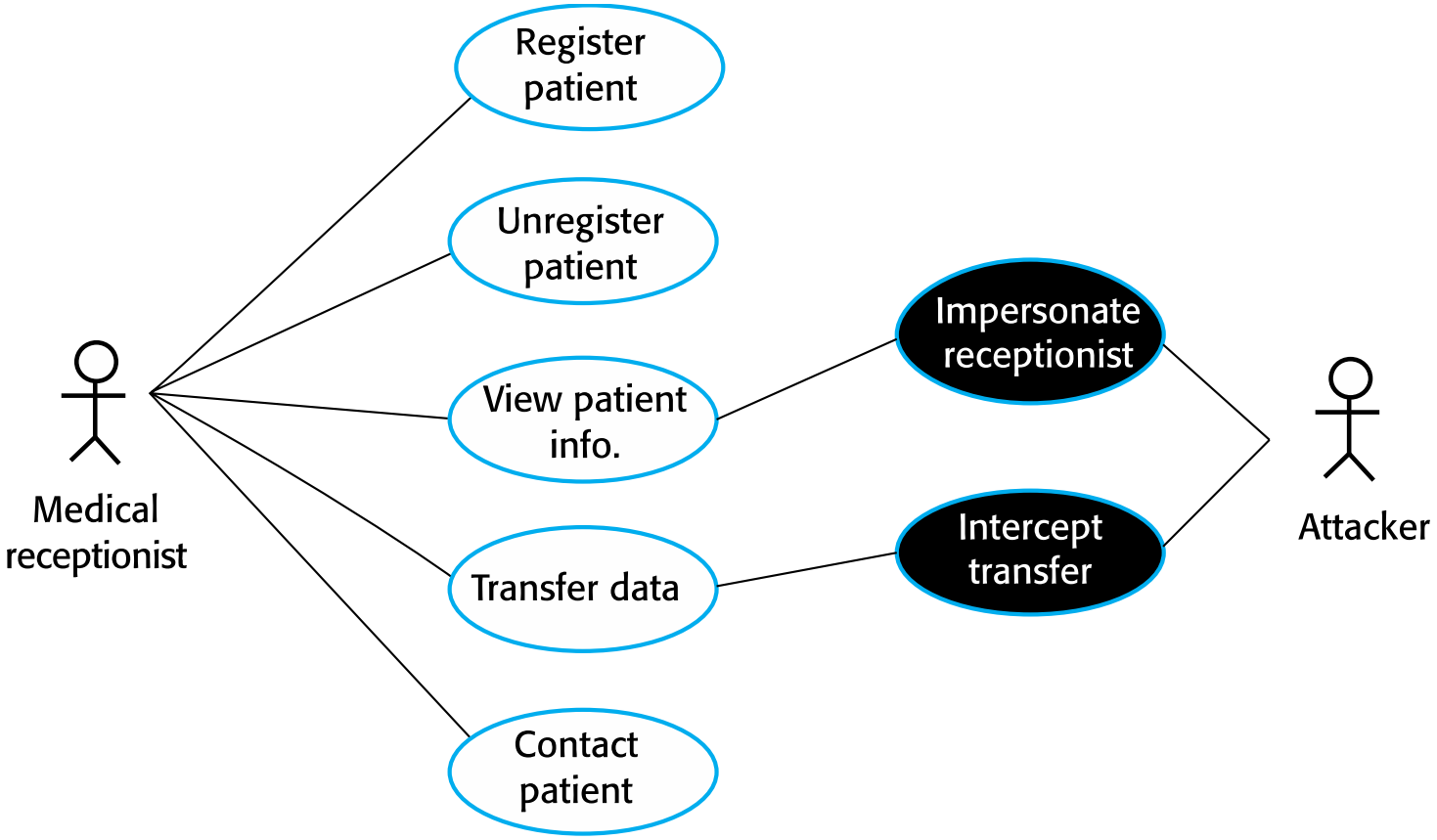
Implementation

1. Identification: Verifying user identity
2. Authentication: Confirming user identity
3. Authorization: Granting access based on permissions
4. Immunity: Protecting against malicious code (e.g., viruses)
5. Integrity: Ensuring data accuracy and reliability
6. Intrusion Detection: Identifying attacks
7. Non-repudiation: Preventing denial of actions
8. Privacy: Protecting sensitive information
9. Security Auditing: Monitoring and reviewing security logs
10. System Maintenance: Ensuring secure updates and maintenance procedures

Misuse cases – Insider Attacks

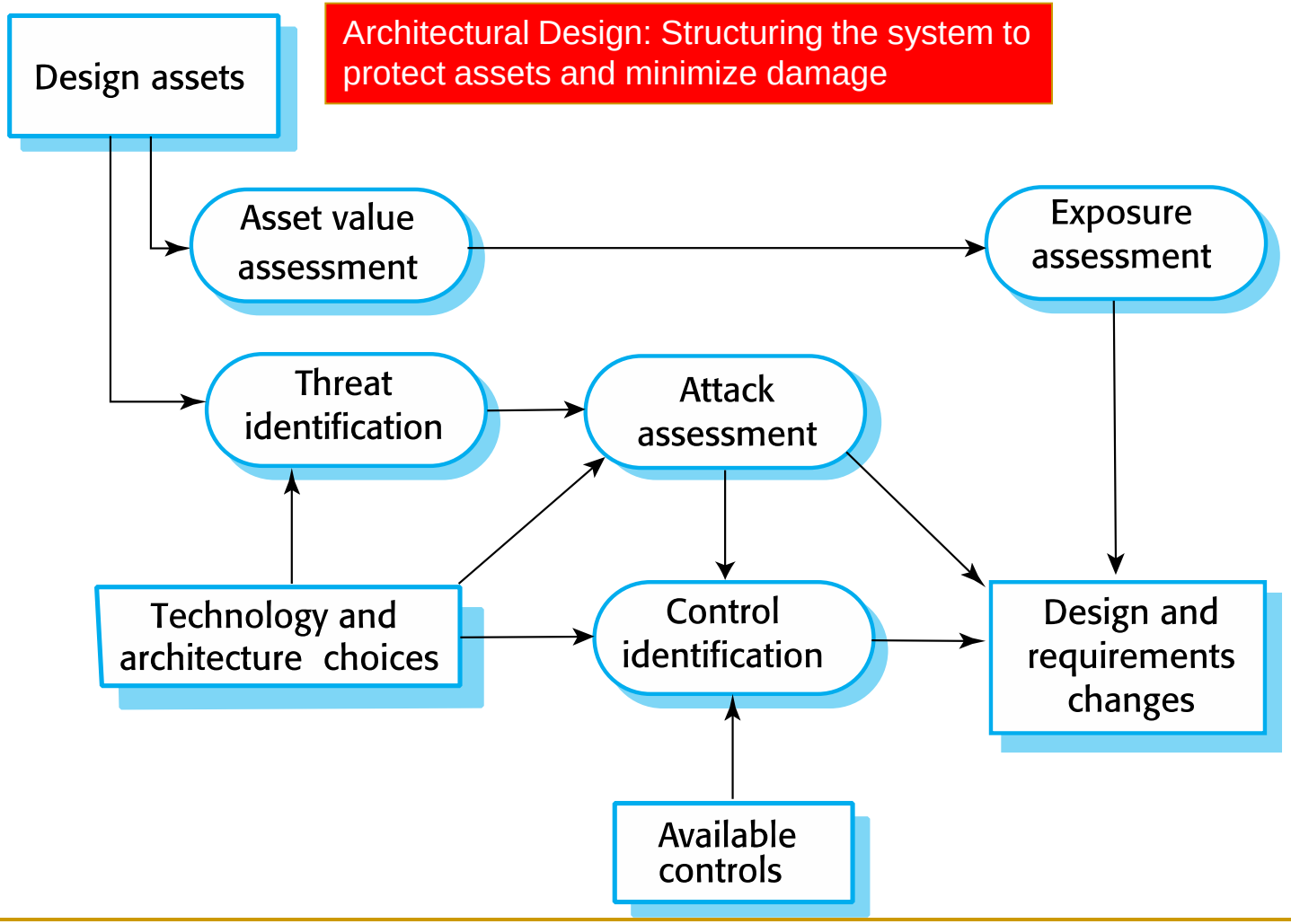
Concept

- Misuse cases are instances of threats to a system



13.4 Secure systems design

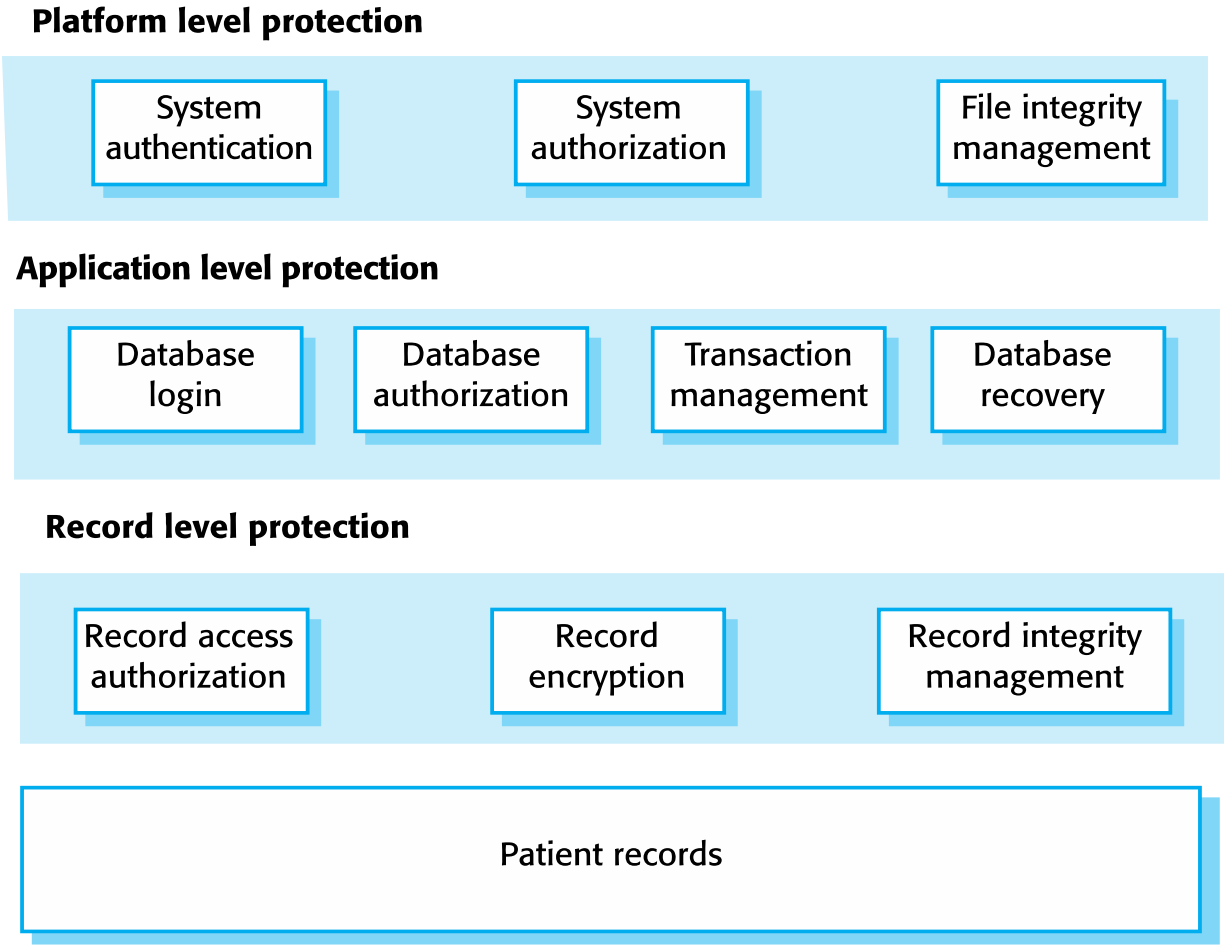
Implementation



Secure systems design, continued

Layered Protection: Implementing security measures at different levels (platform, application, record)

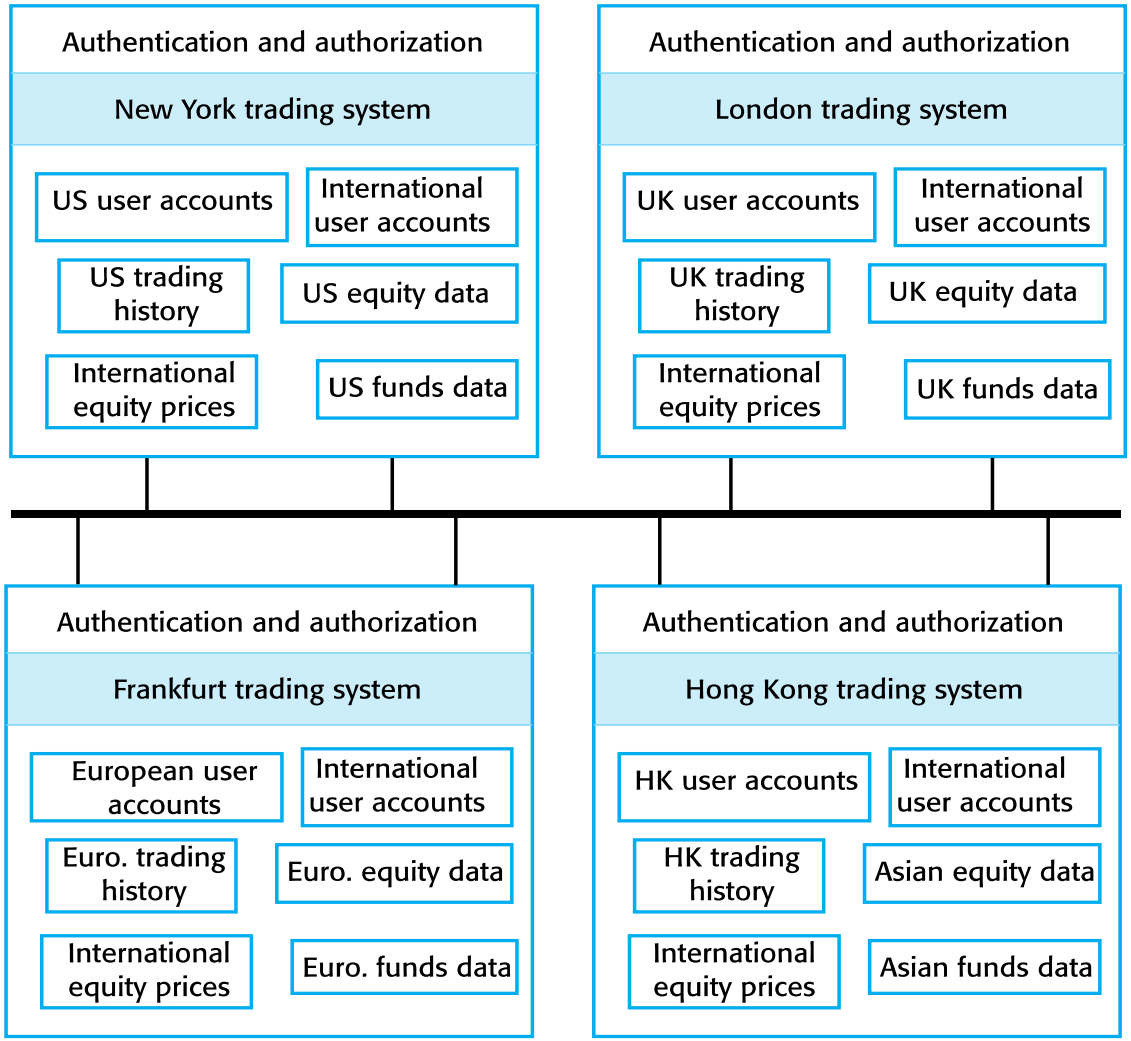
Implementation



Secure systems design, continued

Implementation

Distribution: Spreading assets to reduce the impact of attacks



Secure systems design, continued

Implementation

Design Guidelines: Following best practices for secure development (e.g., avoid single point of failure)

1. Limit the visibility of information in a program
2. Check all inputs for validity
3. Provide a handler for all exceptions
4. Minimize the use of error-prone constructs
5. Provide restart capabilities
6. Check array bounds
7. Include timeouts when calling external components
8. Name all constants that represent real-world values

13.5 Security testing & assurance

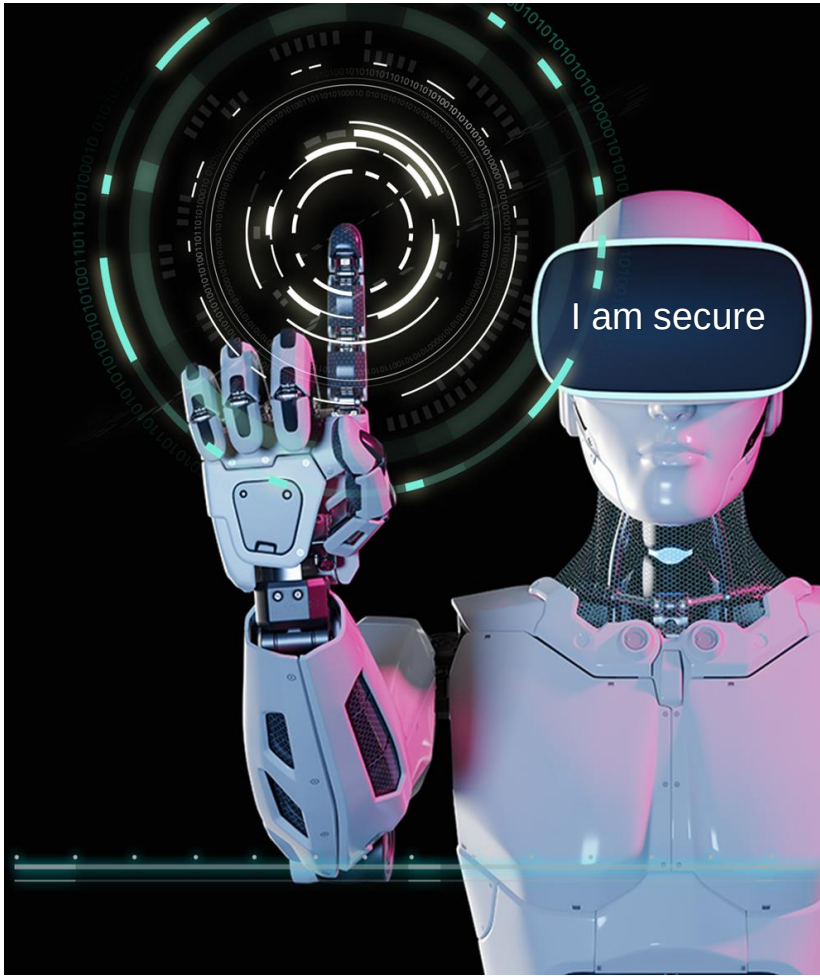
Measuremen

- Security testing is hard because it tests what *shouldn't happen*, attackers are smart and adaptive, and they experiment to find weaknesses

Technique	Strengths	Weaknesses	Applicability
Experience-based Testing	Leverages tester expertise & knowledge of attacks	Can miss vulnerabilities outside known patterns	Early stages, quick checks, targeted testing
Penetration Testing	Simulates real-world attacks, finds vulnerabilities	Can be disruptive, requires skilled testers	Later stages, comprehensive vulnerability checks
Tool-based Analysis	Automated, efficient, can cover wide range of issues	May produce false positives/negatives, limited scope	All stages, specific vulnerability checks
Formal Verification	Mathematically rigorous, proves correctness	Complex, time-consuming, requires formal specs	Critical systems, high assurance requirements

Security engineering in AI era

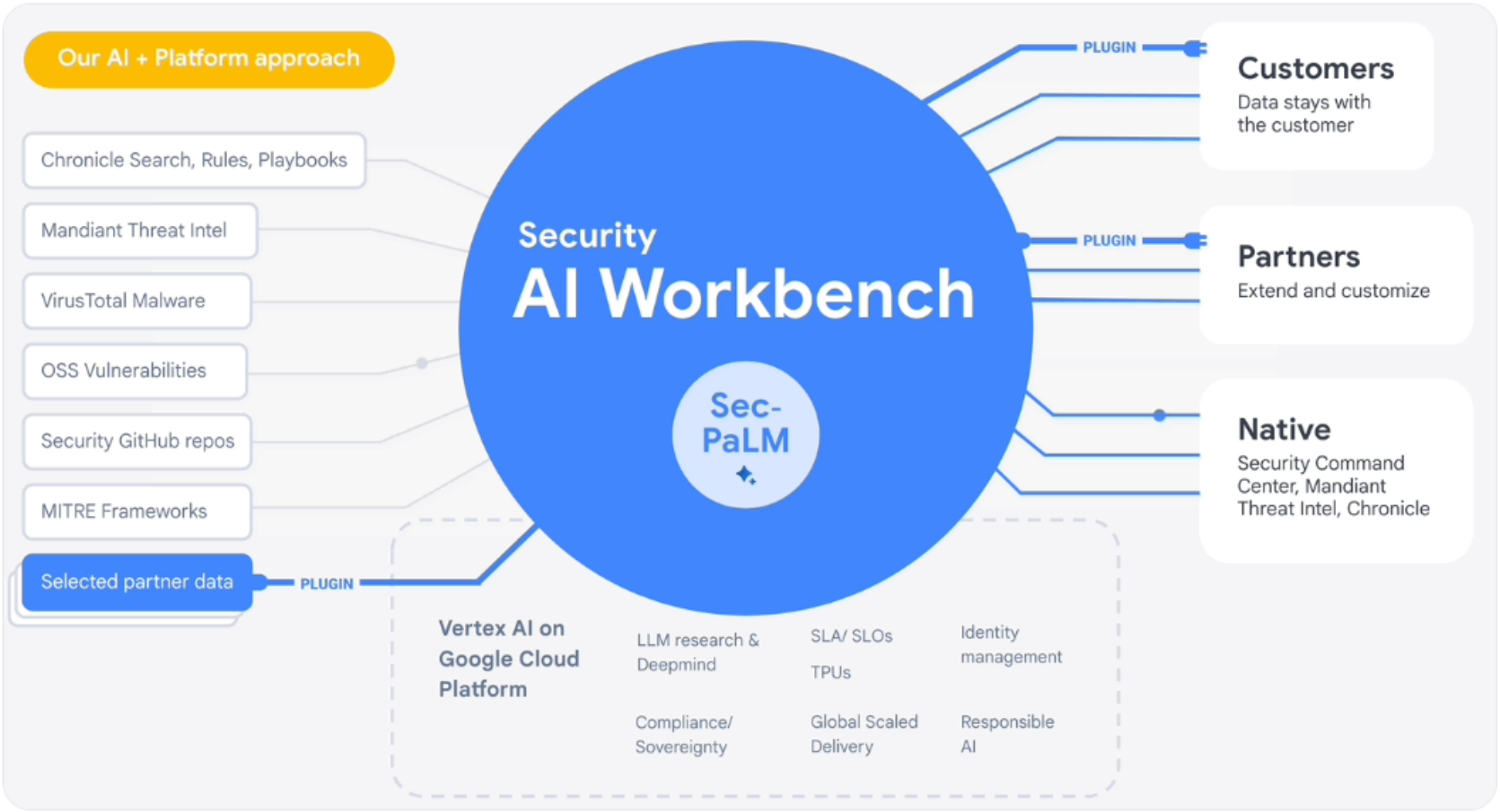
Trend



- **Threat Detection and Response**: LLMs can analyse vast amounts of data to identify and respond to threats in real-time
- **Vulnerability Assessment**: LLMs can analyse code and systems to identify potential vulnerabilities and suggest fixes
- **Security Automation**: LLMs can automate routine security tasks, such as patching and configuration management
- **Personalized Security**: LLMs can tailor security measures to individual users and contexts

Ex; Google's AI for security

Trend



Aligning with industry trends

Trend

- **Shift-Left Security:** Integrating security practices early in the software development lifecycle (SDLC)
- **DevSecOps:** Automating security practices and integrating them into DevOps workflows
- **Zero Trust Security:** Assuming no user or device can be trusted by default and implementing strict verification measures
- **Cloud Security:** Implementing security measures specifically for cloud-based systems and applications
- **AI and Machine Learning for Security:** Utilizing AI and ML to automate threat detection, vulnerability assessment, and incident response

To master today's lesson

- Assignment 4 is designed to assess your mastery of fifth (CLO): Design and test secure systems
- Total Marks: 5
- Submission Deadline: 8:00 PM every Saturday. No late submissions accepted
- You can use any LLM (ChatGPT, Claude, etc.), but you must follow the IT department's LLM ethics guidelines
- Collaboration allowed but submit individual work