System Security Principles

CYBR371: System and Network Security, (2024/T1)

Arman Khouzani (course coordinator), Mohammad Nekooei

Slides modified from "Masood Mansoori"

26 February, 2024

Victoria University of Wellington – School of Engineering and Computer Science

“System” Security: Definition of a “system”

In the context of system security, the term “system” refers to any combination of hardware, software, infrastructure, and processes that work together to perform specific functions or tasks.

This can encompass a wide range of entities, such as a single computer, a network of computers, databases, applications, operating systems, and even the entire IT infrastructure of an organisation.

“System” Security: Scope

The scope of a “system” in system security is defined by the boundaries of what is being protected. e.g.:

- **A single device:** e.g. a computer, smartphone, smart card.
- **A network:** e.g. a local area networks (LANs), wide area networks (WANs), the internet, or intranets.
- **An application:** Software applications or services that run on devices or over networks, including web applications, databases, and enterprise systems.
- **An IT infrastructure:** The comprehensive framework that supports an organisation’s computing needs, including its network, hardware, software, and policies, procedures, staff.

System security components

Policy (specification)

- What security properties to focus on;
- What is and is not allowed.

Mechanism (implementation, controls)

- The mechanism enforces the policy.
- Includes procedural controls, not just technical ones.
  - E.g., who may enter the room where backups are stored;
  - How new accounts are established.
- Prevention, Detection, Recovery.

Assurance

- Verifying that the mechanism implements the policy.

Security

System Security: The process of protecting a system's resources from access, modification or corruption by unauthorised entities.

Information security: a "well-informed sense of assurance that the information risks and controls are in balance." – Jim Anderson, Inovant (2002)

Security should be considered a balance between ...and ...

Why is Security Hard?

- Only one attack needs to succeed.
- Need to place defence at the right point in the system.
- May rely on keeping secrets but also sharing them.
- People don’t realise its value until a security failure.
- Security requires constant monitoring.
- Often added as an after-thought.
- Impossible to obtain perfect security.
    - it is a process, not an absolute

Security Principles

General principles

- Eight principles underlying design and implementation of security mechanisms
- These are guidelines, not hard and fast rules
- Not exhaustive

Principle 1: “Least Privilege”

Principle of least privilege:

Only the minimum amount of access necessary to perform an operation should be granted, and that access should be granted only for the minimum amount of time necessary.

- The *function* of a subject (not its identity) should determine this.
- If reduced privileges are sufficient for a given task, the subject should request only those privileges.

“Least Privilege”: In practice

- There is a limit to how much granularity a system can handle
- Systems are often not designed with the necessary granularity
    - e.g. Employee information stored in a file vs. Relational Database

Full name	Office	Emp. Type	Salary
Masood Mansoori	CO130	Academic	20,000$
David Fox	AM405	Professional	18,000$
Ben Anderson	CO134	Academic/HoS	34,000$

Employees.csv
Full name, Office, Emp. Type, Salary
Masood Mansoori, CO130, Academic, 20,000$
David Fox, AM405, Professional, 18,000$
Ben Anderson, CO134, Academic/HoS, 34,000$

Principle 2: “Separation of Privilege”

Separation of Privilege:

(As much as is feasible…) a system should not grant permission based on a single condition

- E.g., require more than one system admin to issue a critical command, or more than one teller to issue an ATM card

Principle 3: “Fail-Safe Defaults”

Fail-Safe Defaults:

Unless a subject is given explicit access to an object, it should be denied access, that is, the default is no access.

```
# Squid web proxy example
# host and network definitions
acl localhost src 127.0.0.1/255.255.255.255
acl mynetworks 10.10.10.0/28
acl badwebsite www.facebook.com
## ports and protocols allowed
acl Safe_ports port 80 443
http_access deny !Safe_ports
http_access allow mynetworks safe_ports
http_access deny localhost badwebsite
http_access deny all
```

Principle 3: “Fail-Safe Defaults”

More generally, in case of ambiguity the system should default to the more restrictive case.

Need to argue why a user *should have* access. Do not argue why a user *should not have* access.

Principle 4: “Economy of Mechanism”

Economy of Mechanism:

- Security mechanisms should be as simple as possible.
- Offering too much functionality can be dangerous.
  - e.g., Macros in Excel, Word
  - e.g., Postscript can execute arbitrary code

Principle 5: “Complete Mediation”

Complete Mediation:

- All accesses to objects should be checked to ensure they are allowed.
- A system should mediate any request to read an object – even on the second such request by the same subject!
    - Don’t cache authorisation results.
    - Don’t rely on authentication/authorisation performed by another module.

“Complete Mediation”: Insecure Example

- (Insecure) Example:
  when a process tries to read a file, the system checks access rights
  - If allowed, it gives the process a file descriptor
  - File descriptor is presented to OS for access

  If permissions are subsequently revoked, the process still has a valid file descriptor!
- DNS example?

Principle 6: “Open Design”

Open Design:

- No “security through obscurity”
- Security of a system should not depend on the secrecy of its implementation.
    - Of course, secret keys do not violate this principle!

Principle 7: “Least Common Mechanism”

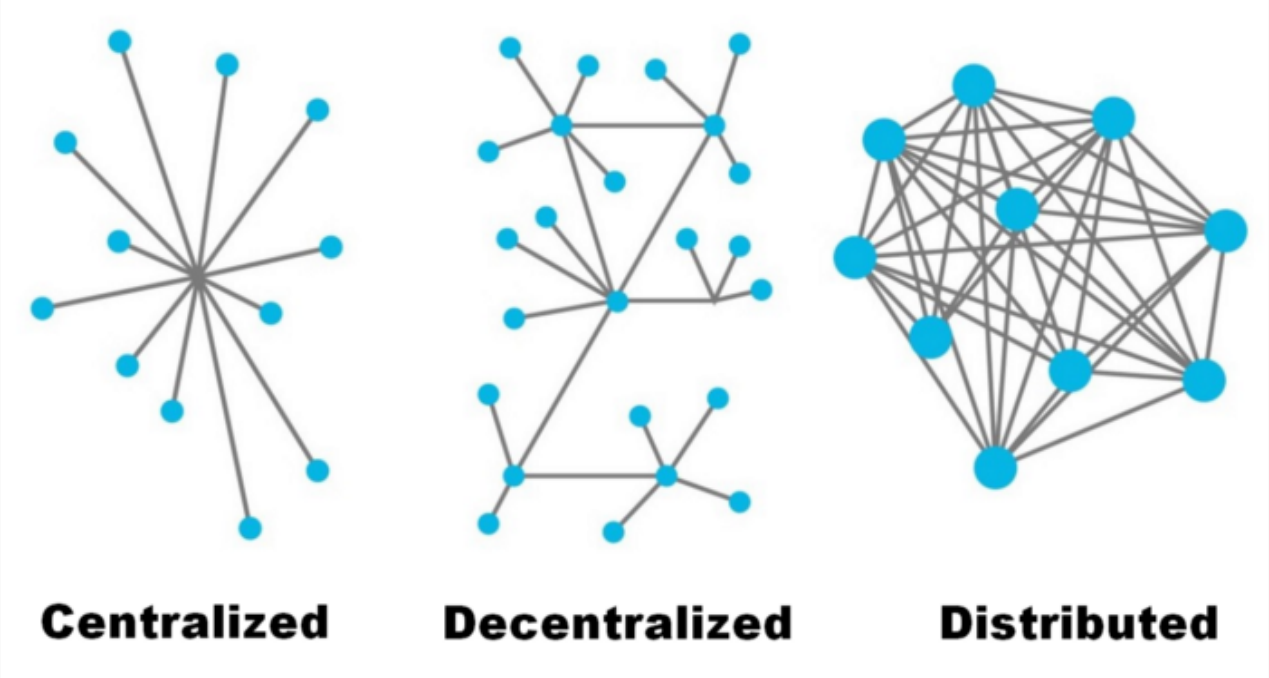
Least Common Mechanism:

- Minimize mechanisms depended upon by all users.
    - Minimize the impact of a security flaw.
- Shared mechanisms are a potential information path, and so may be used to compromise security.
- Shared mechanisms also expose the system to potential DoS attacks.

Principle 7: “Centralised or Decentralised?”

Centralised or Decentralised?

- A centralised system may be more secure
  - Policy will always be enforced consistently
  - No propagation delays if policy changes
- A centralised system can lead to performance bottlenecks, and is less flexible



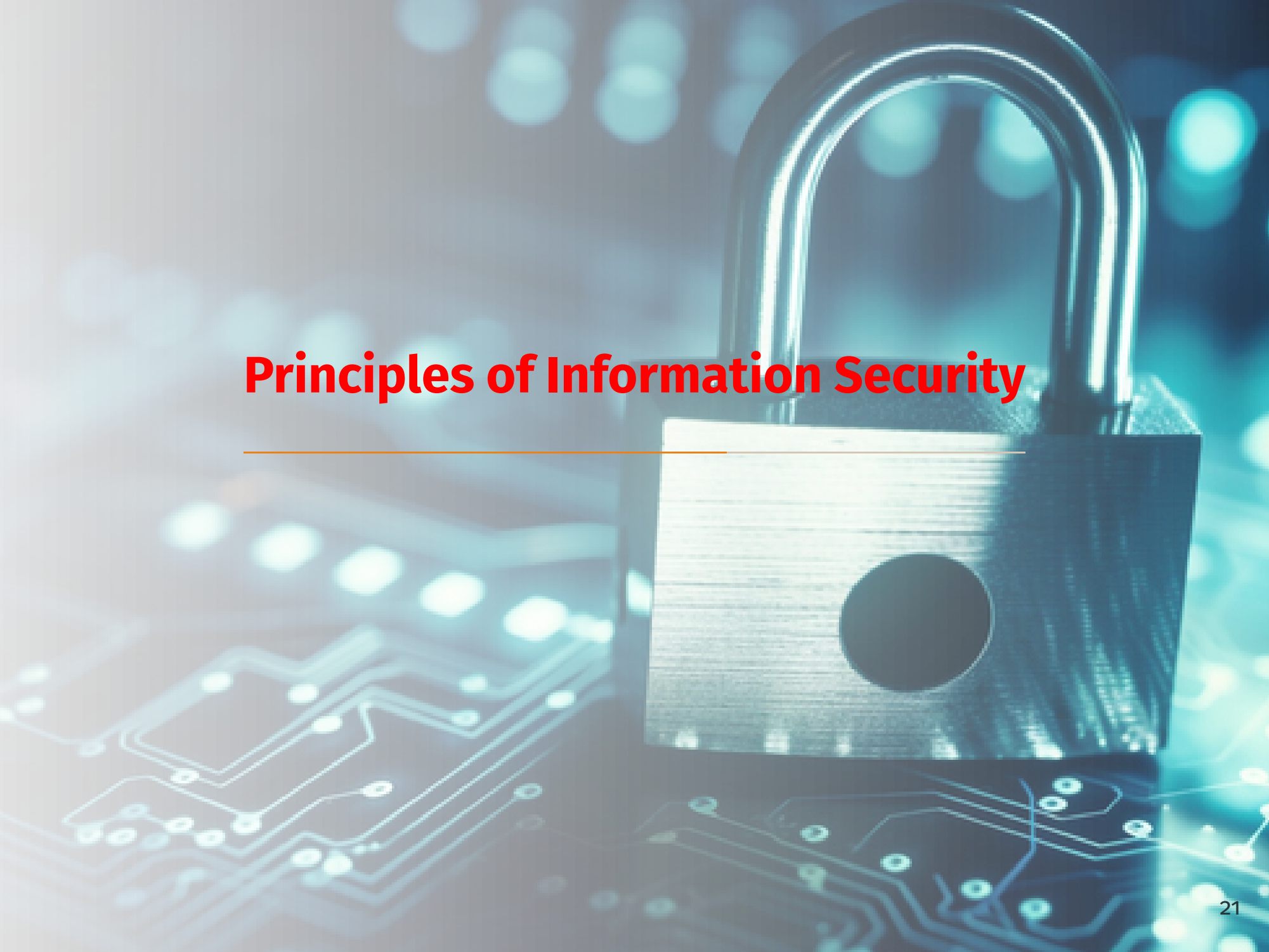
Principle 8: “Psychological Acceptability”

Psychological Acceptability:

- User interface must be easy to use, so that users routinely and automatically apply the mechanisms correctly. Otherwise, they will be bypassed.
- Security mechanisms should not make access to the resource more difficult.
- If mechanisms are too cumbersome, they will be circumvented.
    - Even if they are used, they may be used incorrectly.

+ 1 Key Point: “Secure the Weakest Link”

- A security system is only as strong as its weakest link.
- Attackers go after the easy targets.
    - e.g. they will go after endpoints rather than trying to crack encryption.
    - they will attempt to crack an application visible through the firewall rather than the firewall itself.
- Identify and strengthen weak links until an acceptable level of risk is achieved (i.e. “risk appetite”).



Principles of Information Security

The Big Questions

Protect what? why? against what? and how?

- **What and where?**
  - Identify the resources you have, *everything!*
- **Why?**
  - Justify why they need to be secured.
- **Against what?**
  - Why are we doing this again?
- **How?**
  - What measures can we take to protect them?

Characteristics of Information

Why?

How valuable are these resources to you/your organisation?

How can these resources be affected by a malicious entity?

C.I.A. triad: **C**onfidentiality, **I**ntegrity, and **A**vailability.

Now extended into list of critical characteristics of information:

- Availability
- Accuracy
- Authenticity
- Confidentiality
- Integrity
- Utility
- Possession

Threat identification and Controls

Against what?

Identify all possible threats

- their likelihood, severity, and its associated risk, for each threat, for each resource!

threat, vulnerability, risk?

attack, weakness, bug, exploit, hack...?

- we need some clarifying definitions!

Vulnerabilities, Threats, and Attacks

Against What?...Definitions

Threat: an object, person, or other entity that represents a danger to an asset.

Threat Agent: the specific instance or a component of a threat.

Vulnerability: a weakness or fault in a system or protection mechanism that opens it to attack or damage.

Attack: acts or actions that exploits a vulnerability (i.e., an identified weakness) in a system.

Vulnerabilities, Threats, and Attacks

Against What?...Definitions

Exploit a technique used to compromise a system.

Exposure a condition or state of being exposed.

Risk the probability of an unwanted occurrence.

Subject an entity being the subject of a threat or having a vulnerability exposed to threat agents and in risk of being exploited.

Examples of Threats

Category of Threat	Examples
Compromises to intellectual property	piracy, copyright infringement
Software attacks	viruses, worms, macros, DoS
Deviations in quality of service	ISP, power, WAN service issues from providers
Espionage or trespass	unauthorised access and/or data collection
Forces of nature	fire, flood, earthquake, lightning
Human error or failure	accidents, employee mistakes
Information extortion	blackmail, information disclosure
Missing/inadequate/incomplete backup	disk drive failure, without backup and recovery plan
Missing/inadequate/incomplete controls	no firewall, or permissive firewall rules
Sabotage or vandalism	destruction of systems or information
Theft	illegal confiscation of equipment or information
Technical hardware failure or errors	equipment failure
Technical software failure or errors	bugs, code problems, unknown loopholes

Vulnerabilities, Threats, and Attacks

Examples of software/network attacks

- Malware (e.g. ransomware, polymorphic)
- DDoS (e.g. Redirection attack, Amplification attacks)
- Password-based attack (e.g. Rainbow attacks)

Controls and Countermeasures

How?

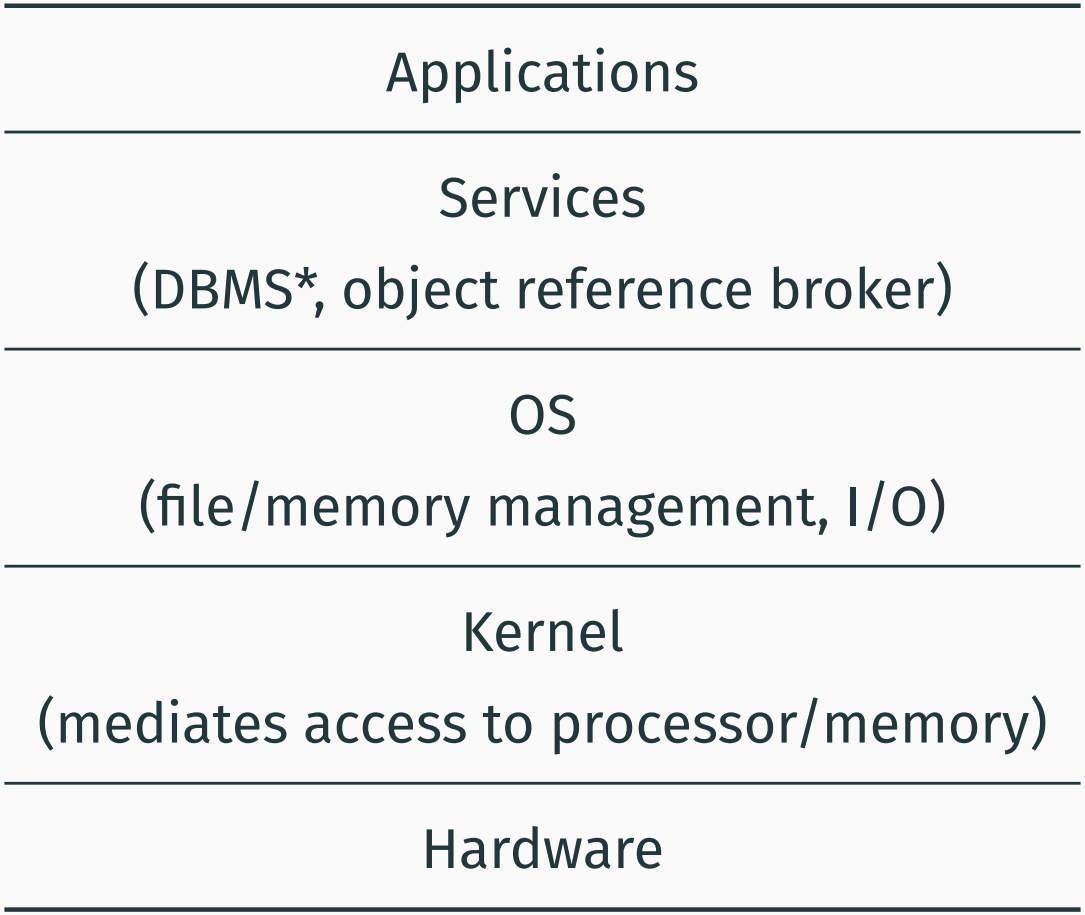
What **counter-measures** can we take against threats? let's call them security controls or just **controls**

(Security) Control: [NIST Glossary]:

- *"A safeguard or countermeasure prescribed for an information system or an organization designed to protect the confidentiality, integrity, and availability of its information and to meet a set of defined security requirements."*
- *"The management, operational, and technical controls [...] prescribed for a system to protect the confidentiality, integrity, and availability of the system, its components, processes, and data."*

System Security Design considerations

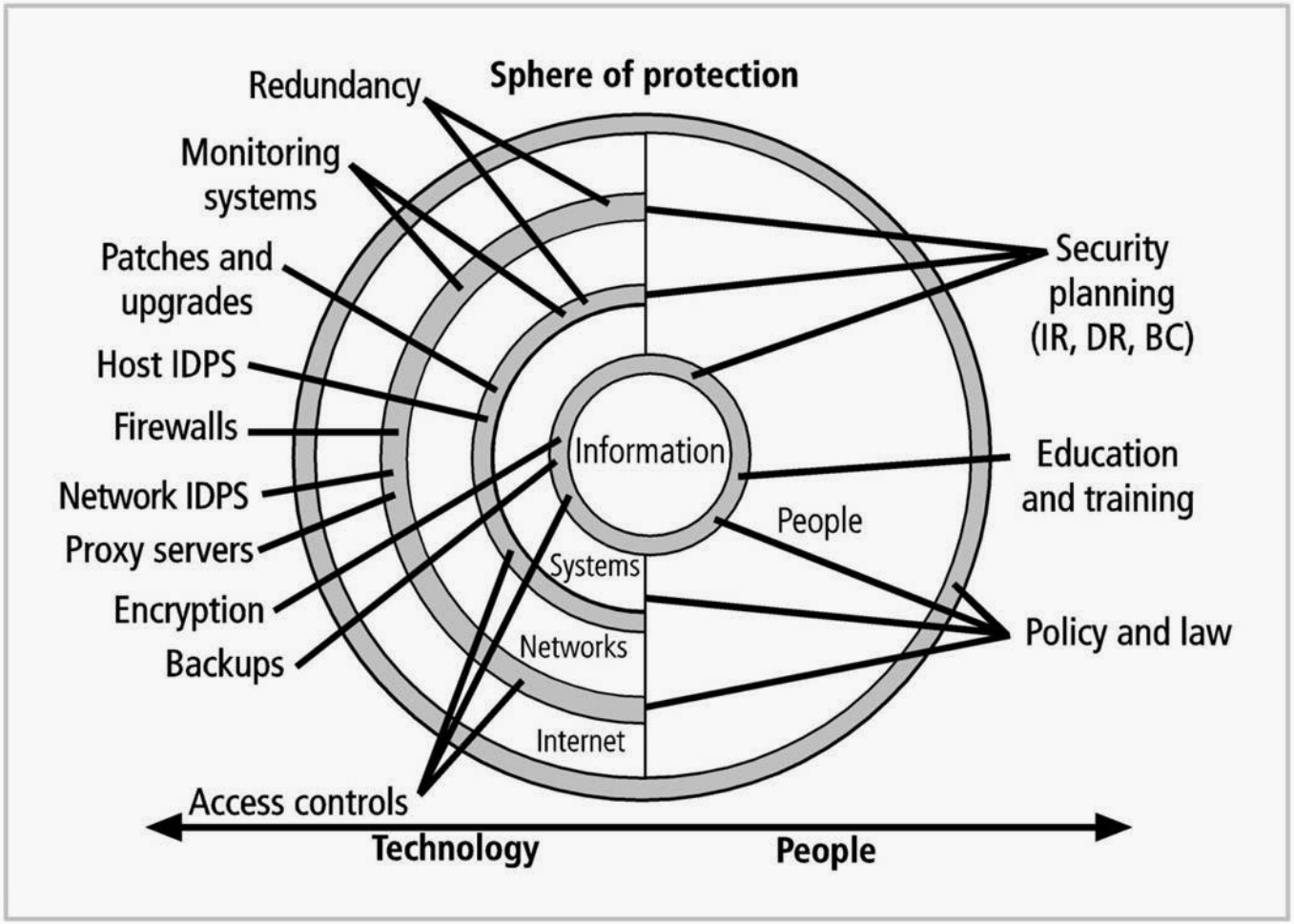
Where should security mechanism(s) be placed?



Controls and Countermeasures

How?

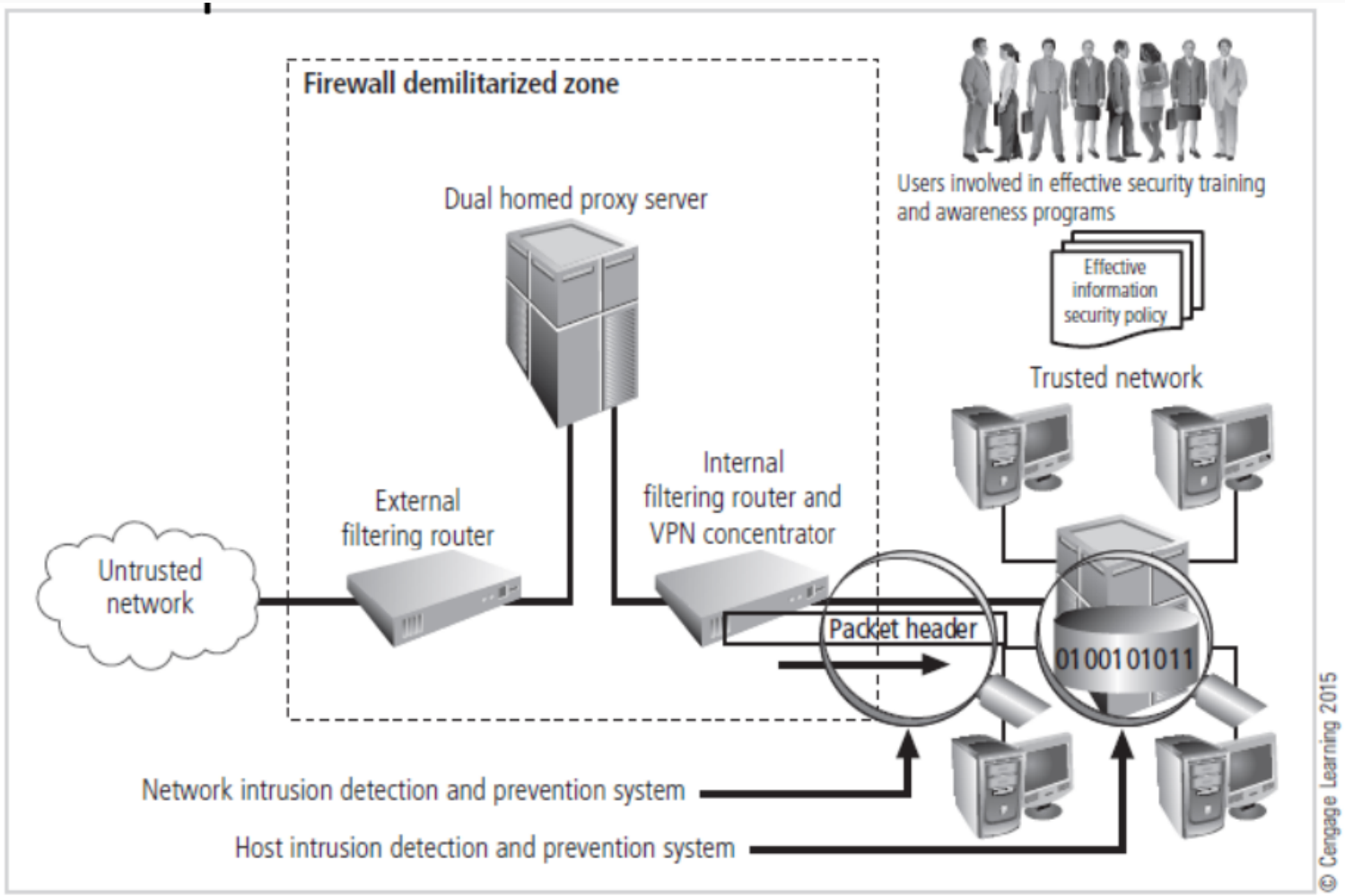
Sphere of use and sphere of protection → Defence in depth



Controls and Countermeasures

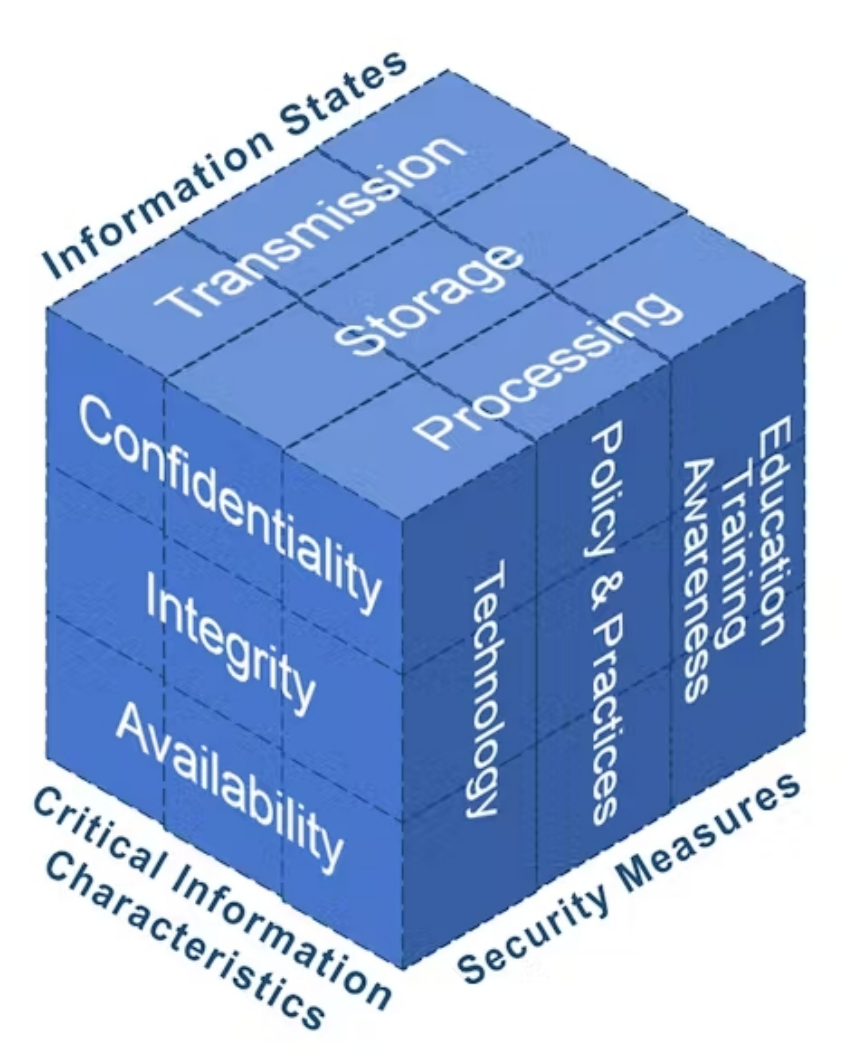
How?

Defence in depth



Controls and Countermeasures

John McCumber designed a framework for establishing and evaluating information security (Assurance)



Approaches to Security Implementation

Bottom-Up Approach is a Grassroots effort in which systems administrators attempt to improve security of their systems.

Key advantage is:

- Technical expertise of individual administrator

Bottom-Up Approach lacks a few critical features:

- Participant support
- Organisational staying power

Top-Down Approach is initiated by upper management

- Issue policy, procedures, and processes
- Dictate goals and expected outcomes of project
- Determine accountability for each required action

Next: Basics of Access Control