

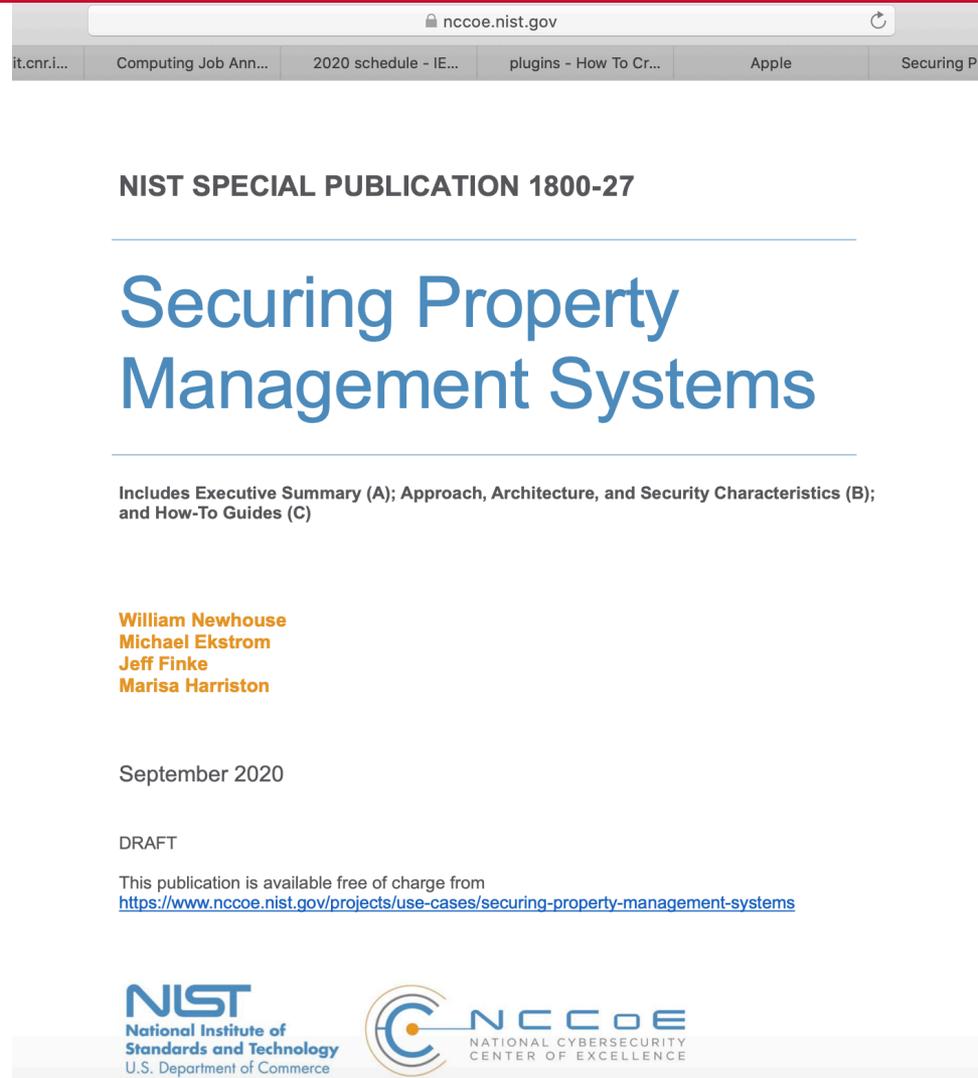
Security Architecture

Lotfi ben Othmane

News About Vulnerabilities

We have only 19 lecture sessions ant 26 students.

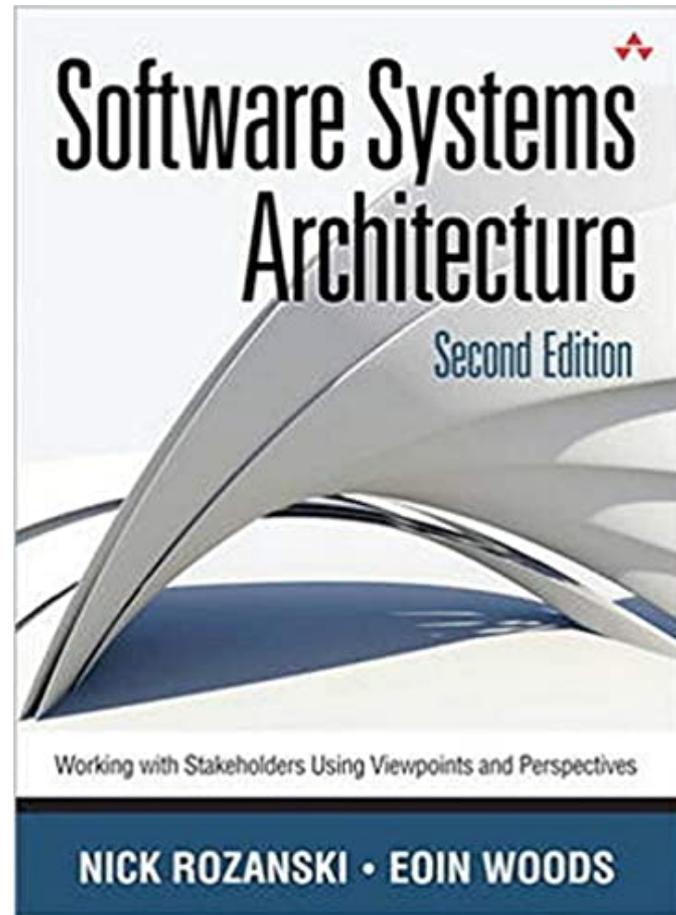
Architecture Standards



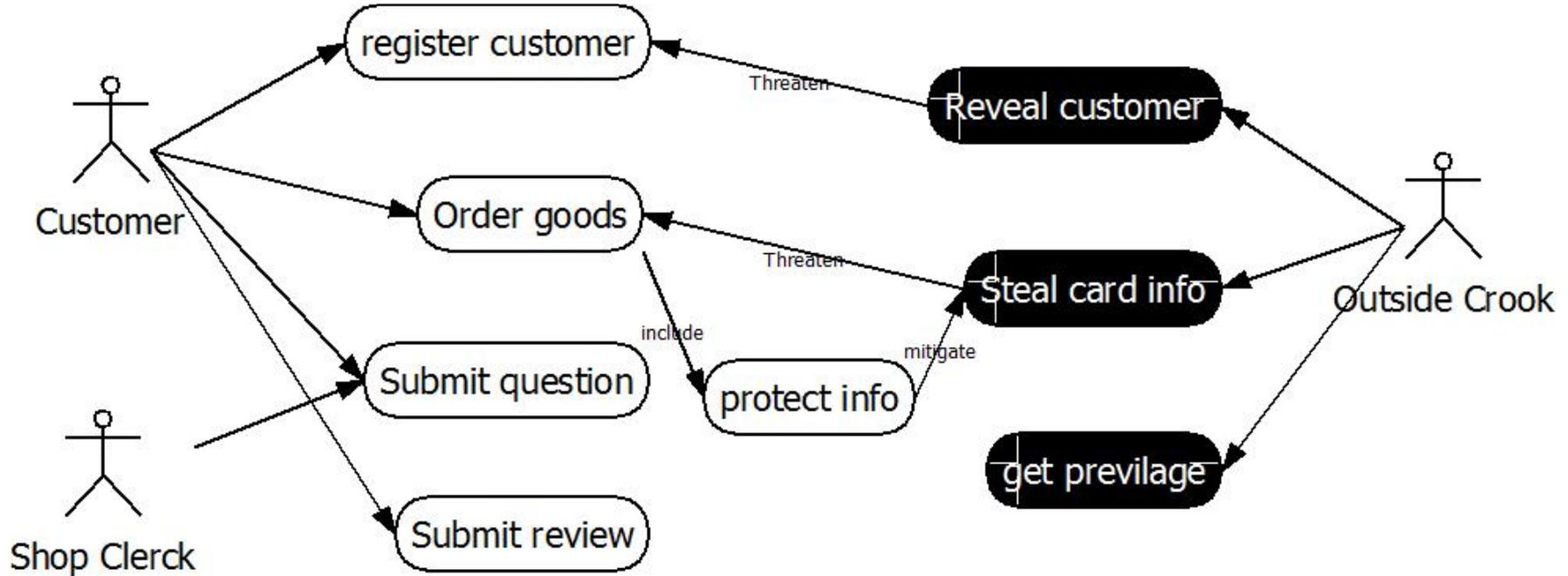
The image is a screenshot of a web browser displaying the cover page of NIST Special Publication 1800-27. The browser's address bar shows the URL 'nccoe.nist.gov'. Below the browser, the page content is centered and features the following elements:

- NIST SPECIAL PUBLICATION 1800-27**
- Securing Property Management Systems**
- Includes Executive Summary (A); Approach, Architecture, and Security Characteristics (B); and How-To Guides (C)**
- William Newhouse**
- Michael Ekstrom**
- Jeff Finke**
- Marisa Harriston**
- September 2020
- DRAFT
- This publication is available free of charge from <https://www.nccoe.nist.gov/projects/use-cases/securing-property-management-systems>
- NIST National Institute of Standards and Technology U.S. Department of Commerce**
- NCCOE NATIONAL CYBERSECURITY CENTER OF EXCELLENCE**

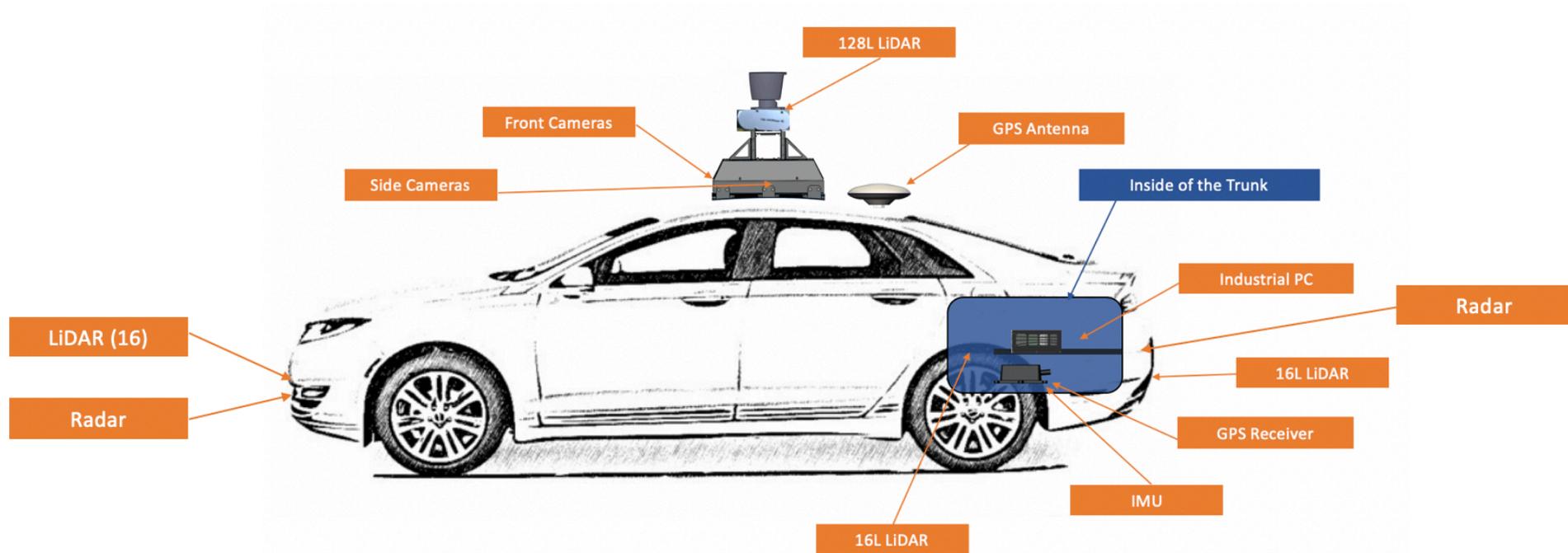
Chapter 26 "The security perspective"



Need for Security Architecture

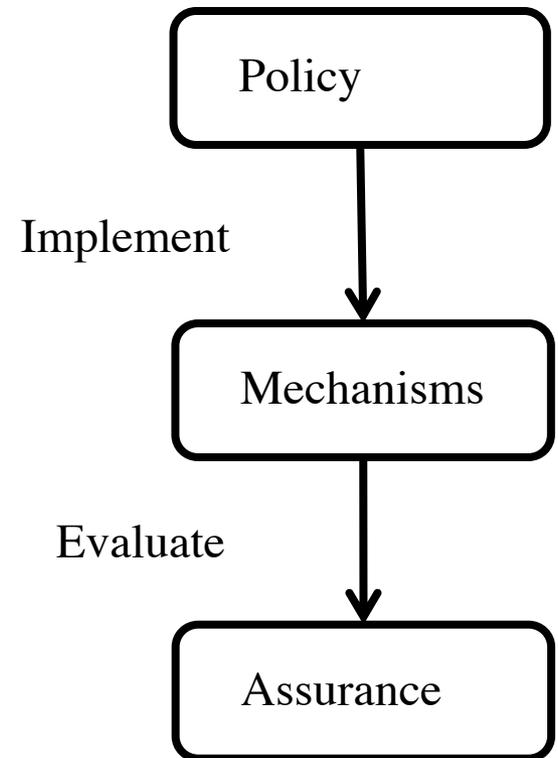


Need for Security Architecture



How do we develop secure systems?

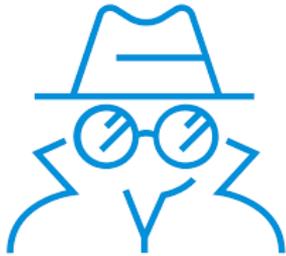
- **Security policy**—rules for managing, protecting and distributing resources
- **Security mechanisms**—functionalities that enforce security policies.
- **Security assurance**—assurance that the mechanisms do enforce the security policies.



Security Mechanisms

- Security mechanisms are: technologies, configuration setting and procedure that enforce the security requirements.
- Examples: authentication, single-sign-on, VPN, access control systems, use of SSL

Need for Security Architecture



Threats and vulnerabilities



Security requirements



Common Categories of Security Mechanisms

1. Authentication authorization, and auditing
2. Nonrepudiation
3. Availability
4. Security monitoring

Security Tactics- Authenticate the Principals

- Principal: Entity that the system needs to identify
- Resource: Items to protect
- Authenticate: Identify reliably the principals

Security Mechanisms

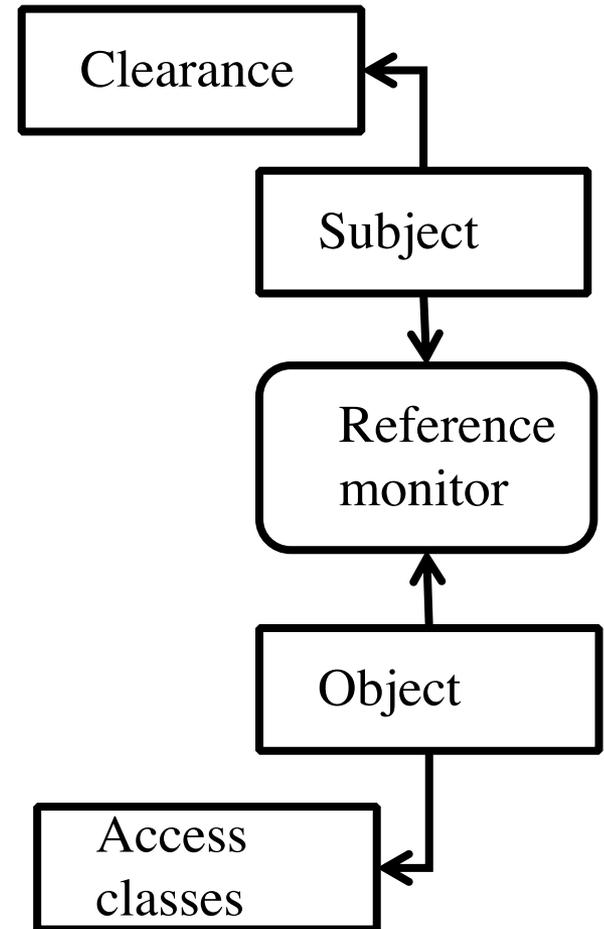
- What security policy does it enforce
- Why do you believe it is secure?
- How do you know it is secure?
- Who certifies they are secure?
- Is an e-commerce system that uses smart cards secure?



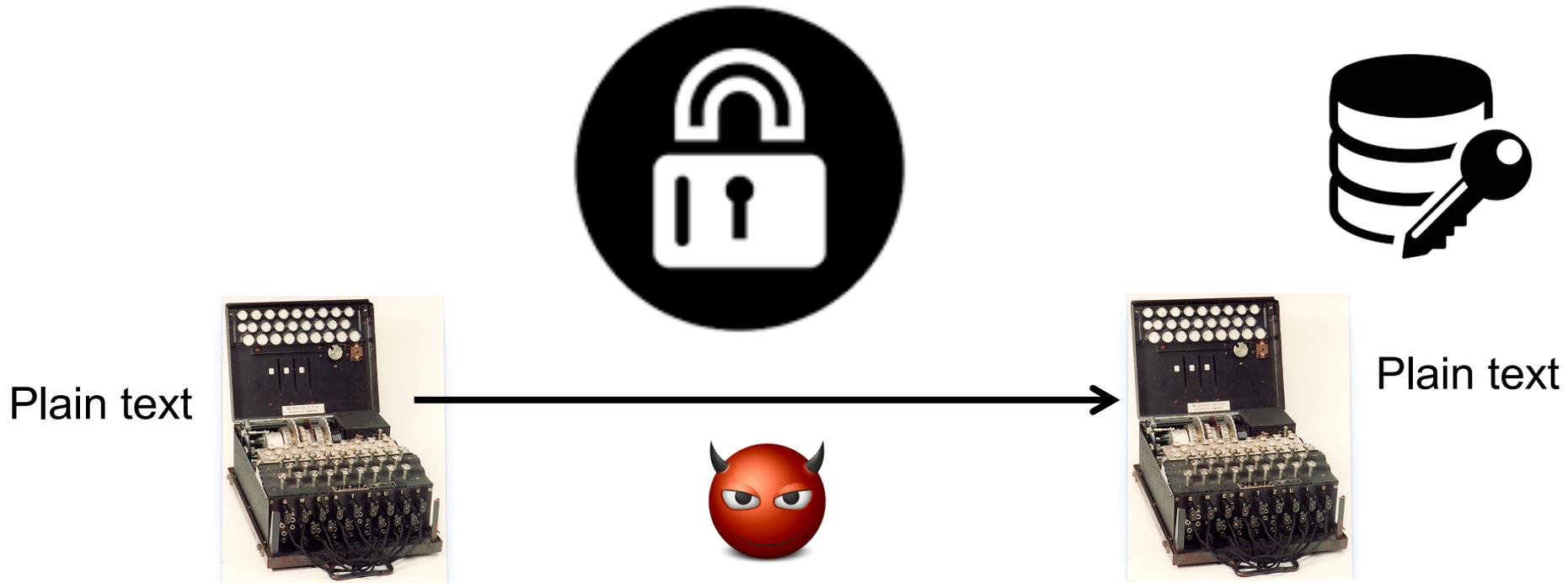
acs.com.hk

Security Tactics - Authorize Access

- Authorize: Allow principals to exercise their legitimate access rights.
- Access control policies—Rules specifying subjects accesses to objects (S,O,M)
 - Subject (S): Entities, e.g., humans -- may have clearance
 - Object (O): Information, data, software
 - Access classes or mode (M): e.g., secrecy level
- Reference monitor—Conceptual model
 - Enforces rules for the subjects to access the objects
- What are the usages of access control mechanisms?

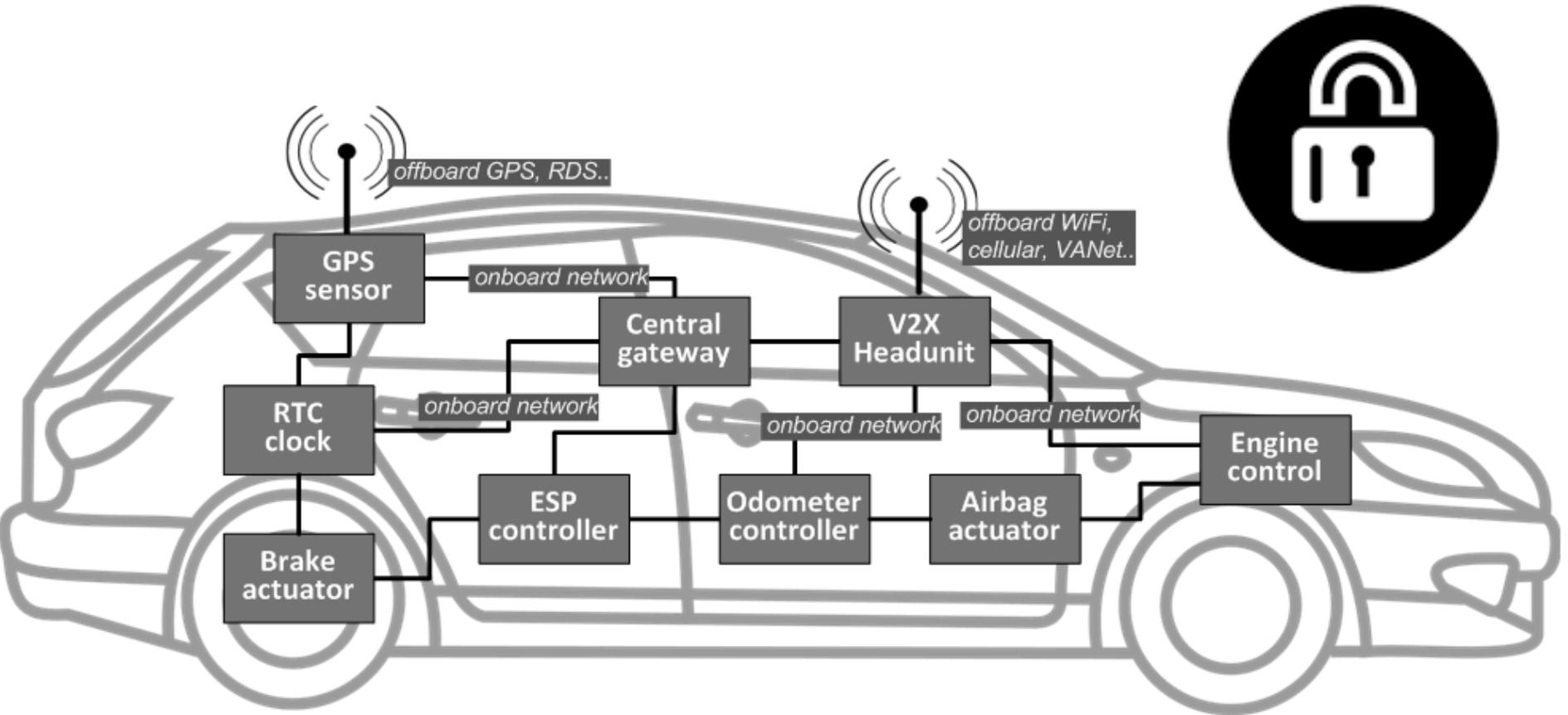


Security Tactics – Ensure Information Secrecy

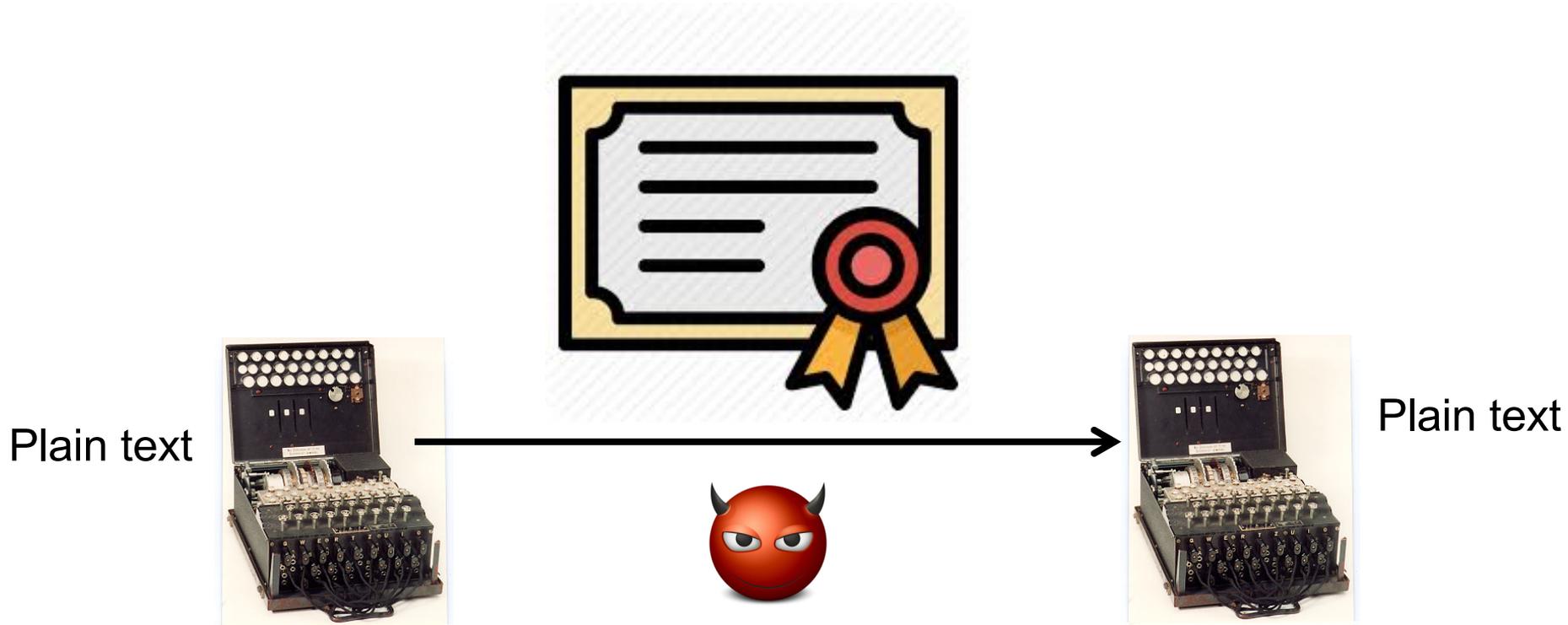


Address information disclosure

Security Tactics – Ensure Information Secrecy



Security Tactics – Ensure Information Integrity



Address unauthorized modification of the information

Security Tactics – Ensure Accountability

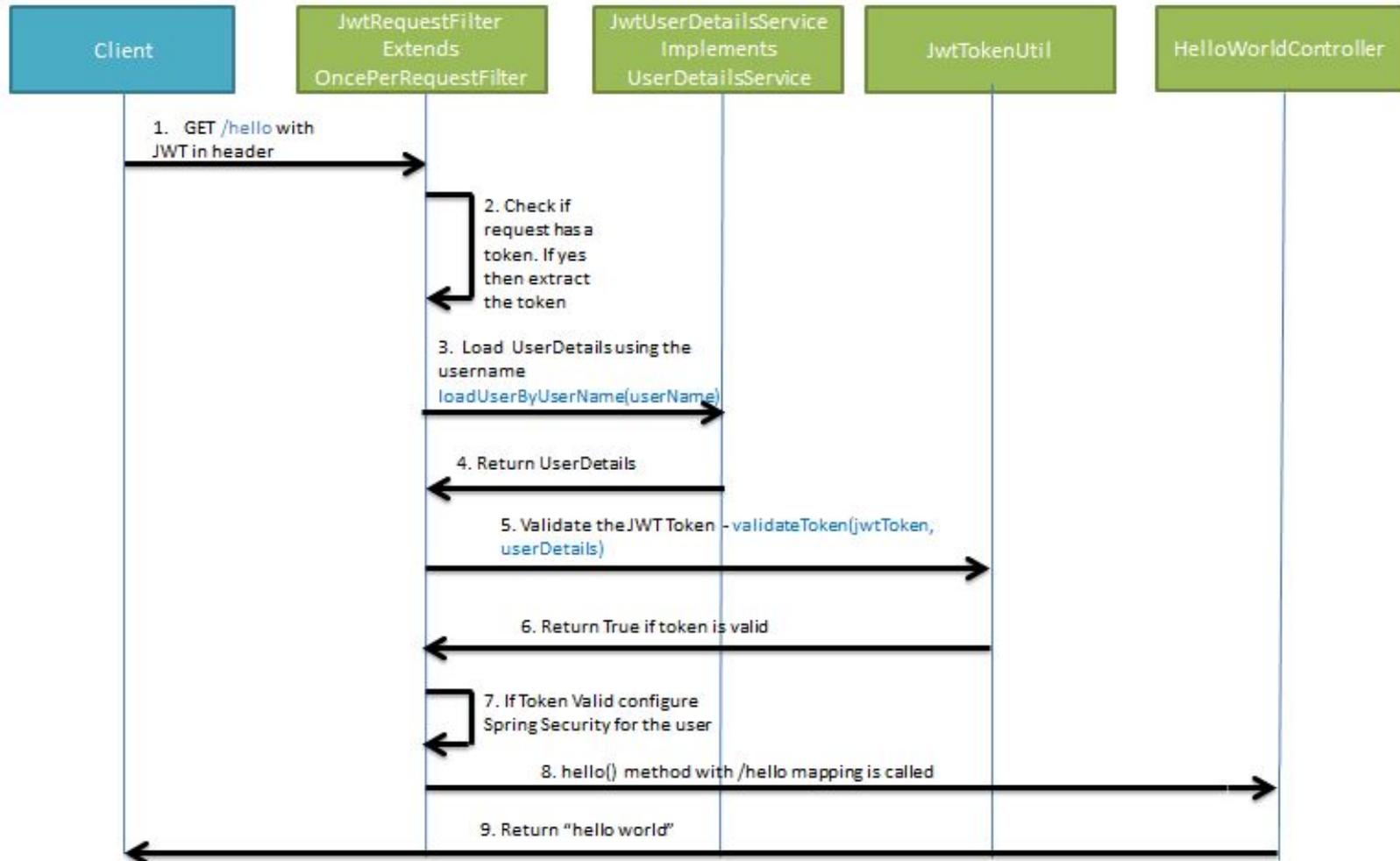


Audit and nonrepudiation for messages

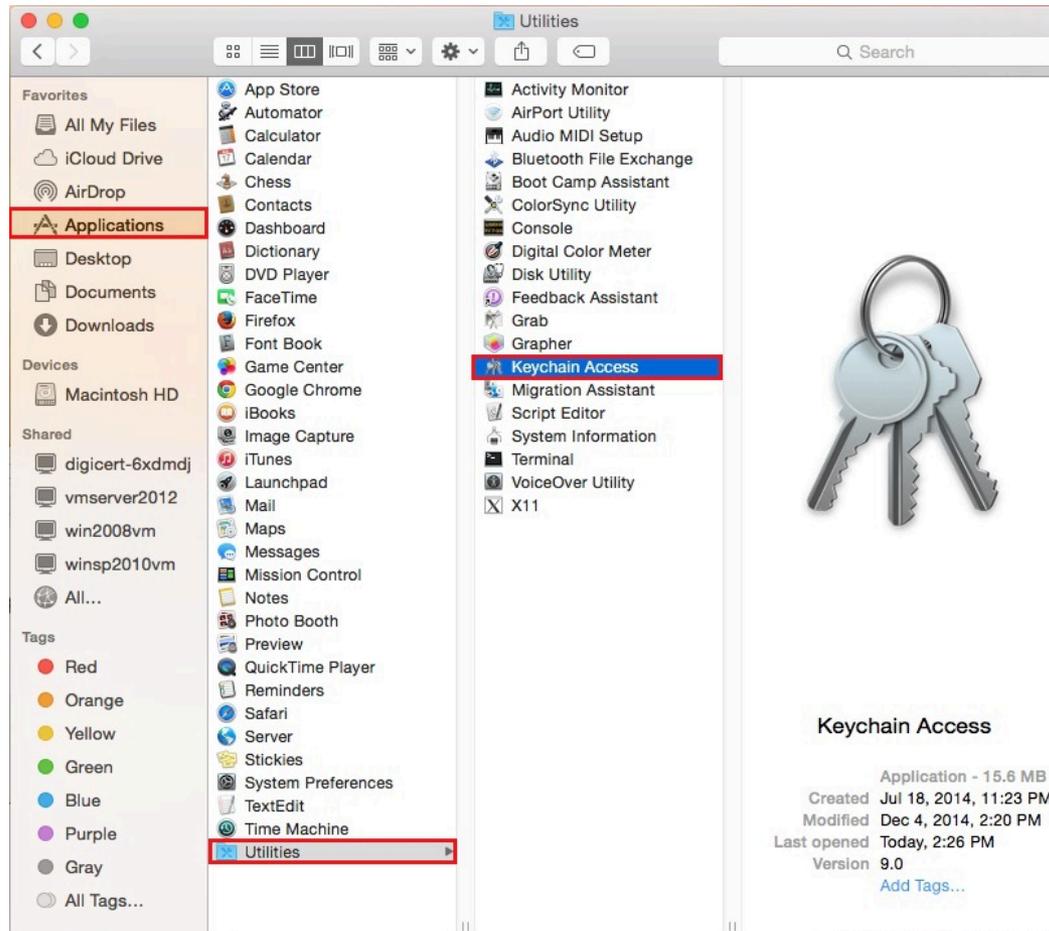
Security Tactics – Integrate Security Technologies

- Implement the security requirements implemented in existing technologies
- Use safe configurations
- Ensure correct use of the technologies

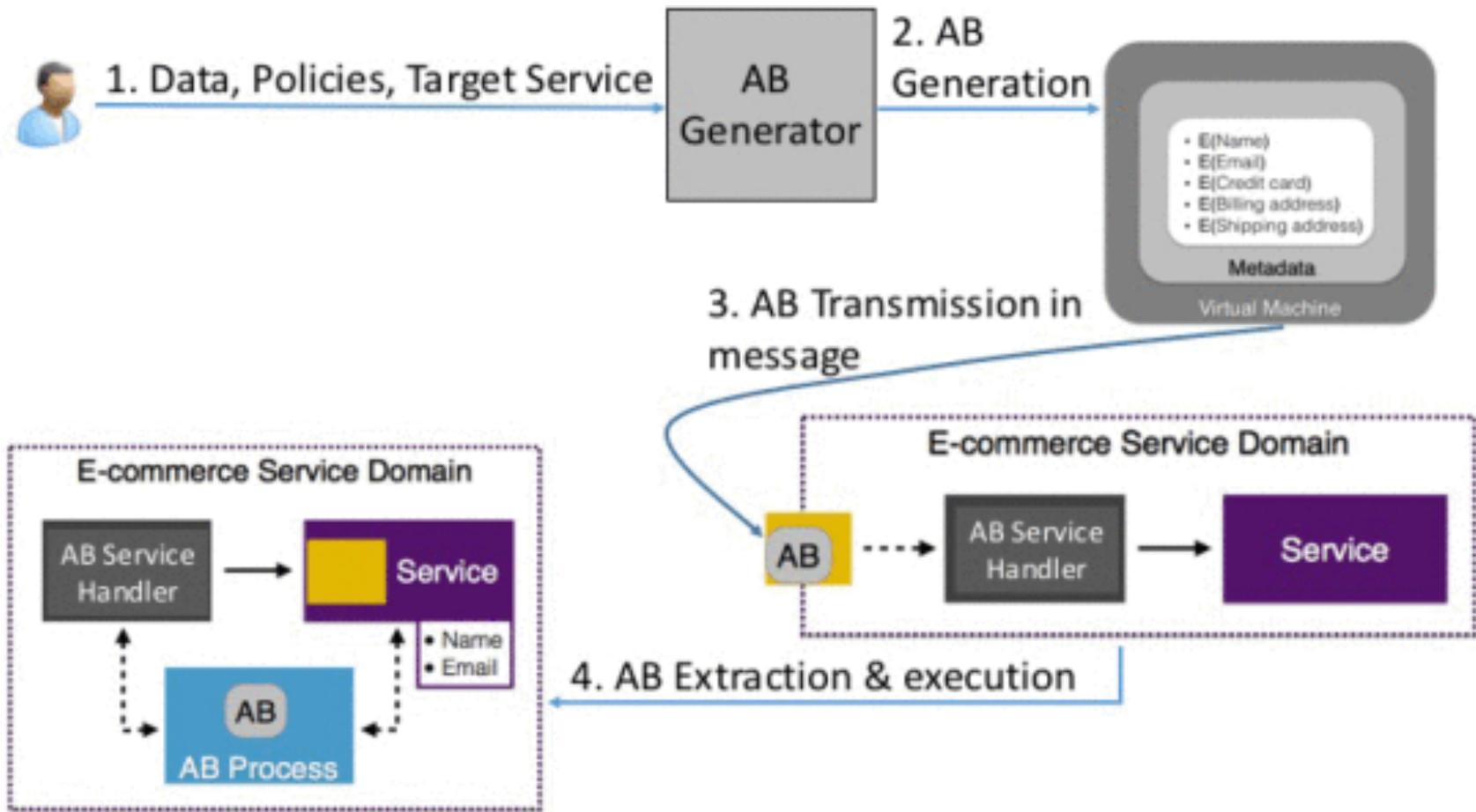
Security Tactics – Integrate Security Technologies



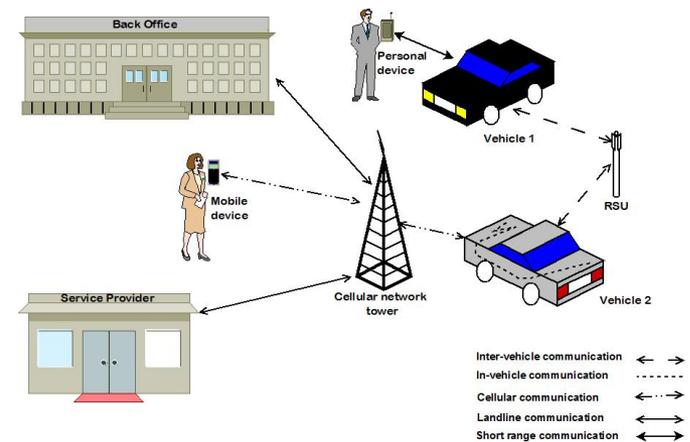
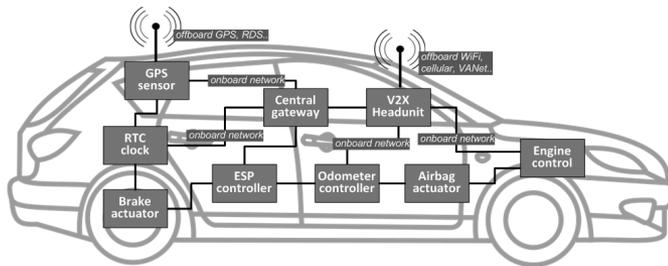
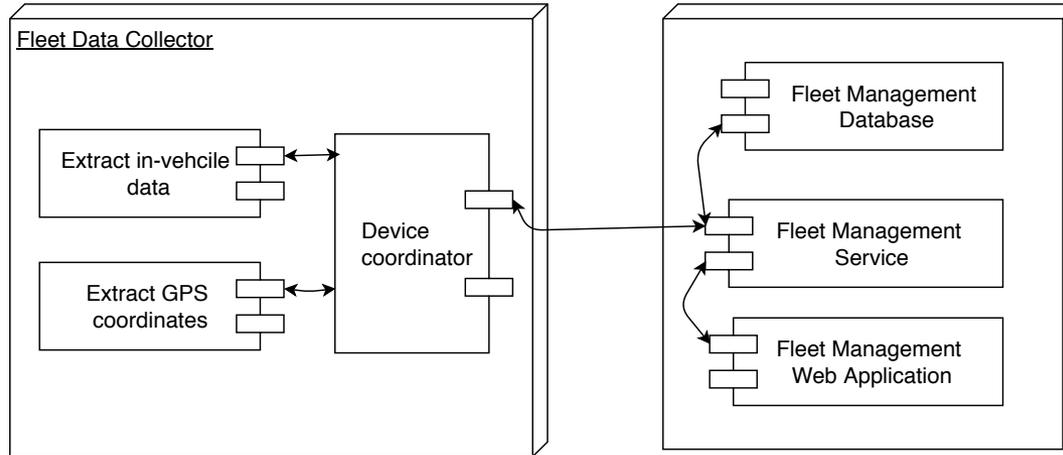
More Tactics – Correctness of the Software Behavior



More Tactics – Distribution of Confidential Data



Systems Are Complex.....



What Is Software Architecture?



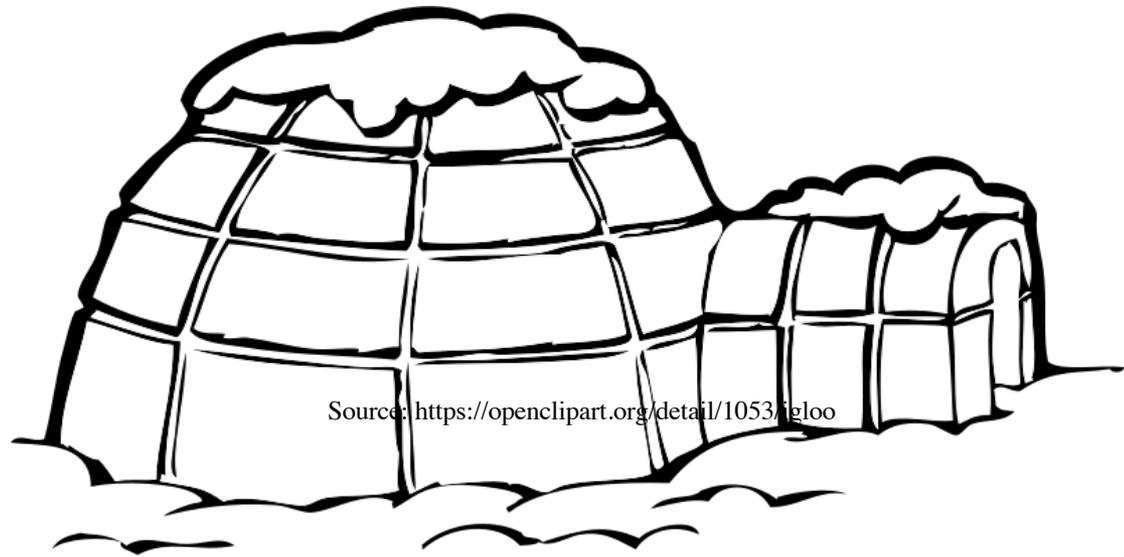
What is Software Architecture?

- “The organizational structure of a system or component.”
 - IEEE Standard Glossary of Software Engineering Terminology

- “The fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution.”
 - IEEE 1472

What is Software Architecture?

Every software implements a software architecture



What is a Security Pattern?

- A **security pattern** describes a particular recurring security problem that arises in specific contexts and presents a well-proven generic solution for it.
 - The solution consists of a set of interacting roles that can be arranged into multiple concrete design structures, as well as a process to create one particular such a structure.

Why Do We Need Security Patterns?

1. Codify basic knowledge
2. Share experience

Structure of Security Patterns

- Context
- Problem
- Solution
 - Includes scope, e.g., #od users
- Consequences
 - Other information may be added such as implementation or lessons learned

Example 1- Password Design and Use

Context - A password mechanism for authentication

Problem - create, use, and manage password while they are accessible to owners and not to imposters

Solution – Factors to consider in the design

- Composition, length, and life time, etc.

- Ownership, data entry, and authentication period, etc.

- Distribution, storage, and transmission, etc.

Consequences

- Increase protection of passwords

- Password guessing reduced

Example 2 - Single Access Point

Example - Grant/Deny external access to a system after checking client rights

Context - Provide external access to a system and ensure no misuse or damage by the client

Problem – Multiple-part systems could be misused by complicated interactions

Solution - Check access legitimacy based on given policy through a single access

Consequences - Simple implementation, no redundant authorization checks, cumbersome to use, single point of failure

Example 3 – Secure Channel

Example – Transfer sensitive data between two parties through Internet

Context – The system delivers functionalities and sensitive information to clients across the public internet

Problem – How to ensure the protection of in-transit data through a public network is secure

Solution – Create secure channels to obscure data in transit and ensure the client and server exchange information to set a secure channel

Consequences – Security is improved, scalability is potentially impacted, cost and maintenance overhead

Wrap up

- Every software implements an architecture
- Architectural documentation is needed to build, evaluate, and maintain complex software
- Views and perspectives help stakeholders to focus on their individual concerns

Open Questions

- How to extract the security architecture from the code of the given software?
- How to verify the security architecture of a software given only the code?
- How to design a usable security description language?
- How to automate the application of security patterns?

Thank you

Any Question?