CSE/CEN 598
Hardware Security & Trust

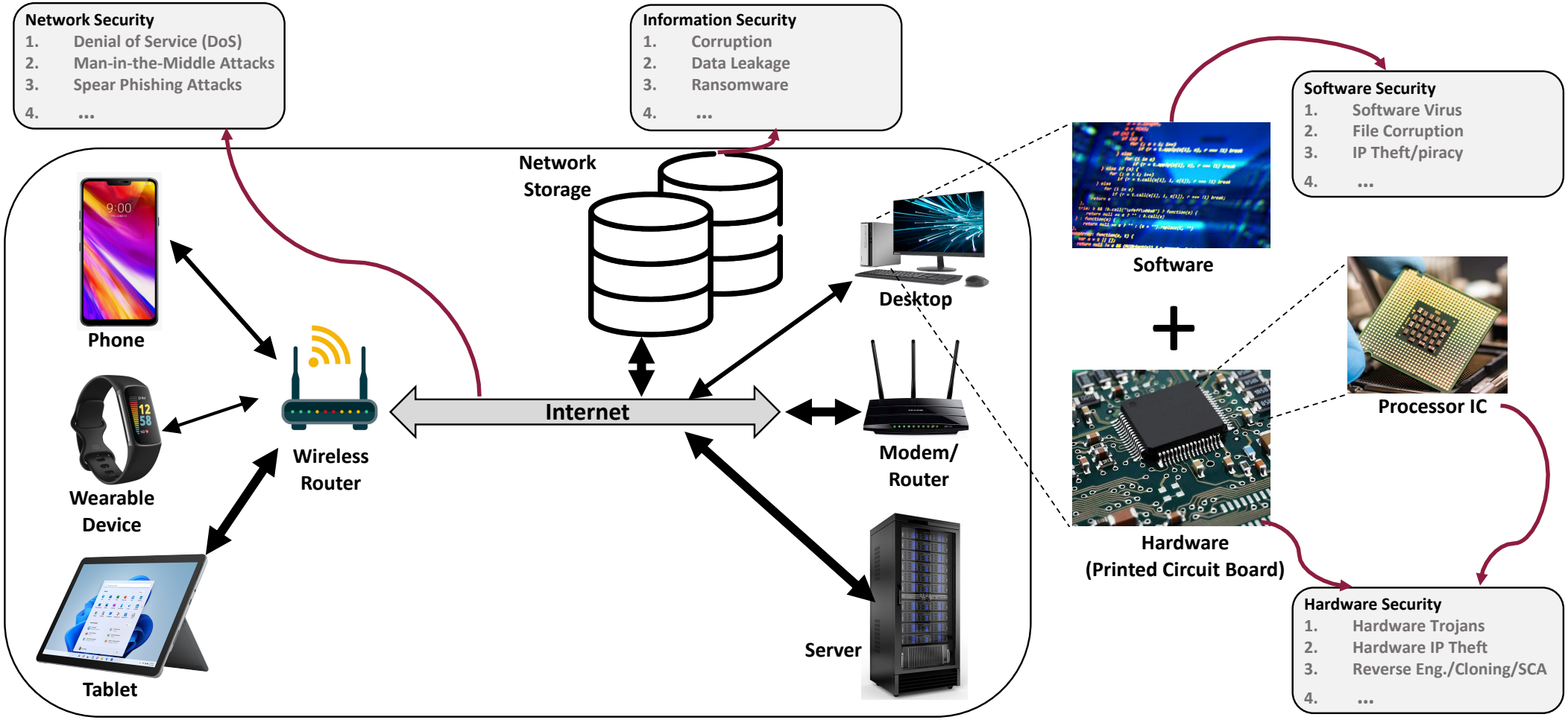
Overview of Hardware Security

Prof. Michel A. Kinsy

Computer Security

- Computer security has become an essential part of modern electronics
- Hardware security deals with security of electronic hardware
  - Including its architecture, implementation, and validation
- Like other security sub-domain, hardware security focuses
  - On attacks crafted to steal or compromise electronic assets
  - And approaches designed to protect those assets
- What are those assets
  - Integrated Circuits (IC), passive components (resistors, capacitors, inductors), and printed circuit boards (PCB)
  - And secrets stored in the electronic component – cryptographic keys, digital rights management (DRM), programmable fuses, sensitive user data, firmware, and configuration data

Security of Modern Computing Systems



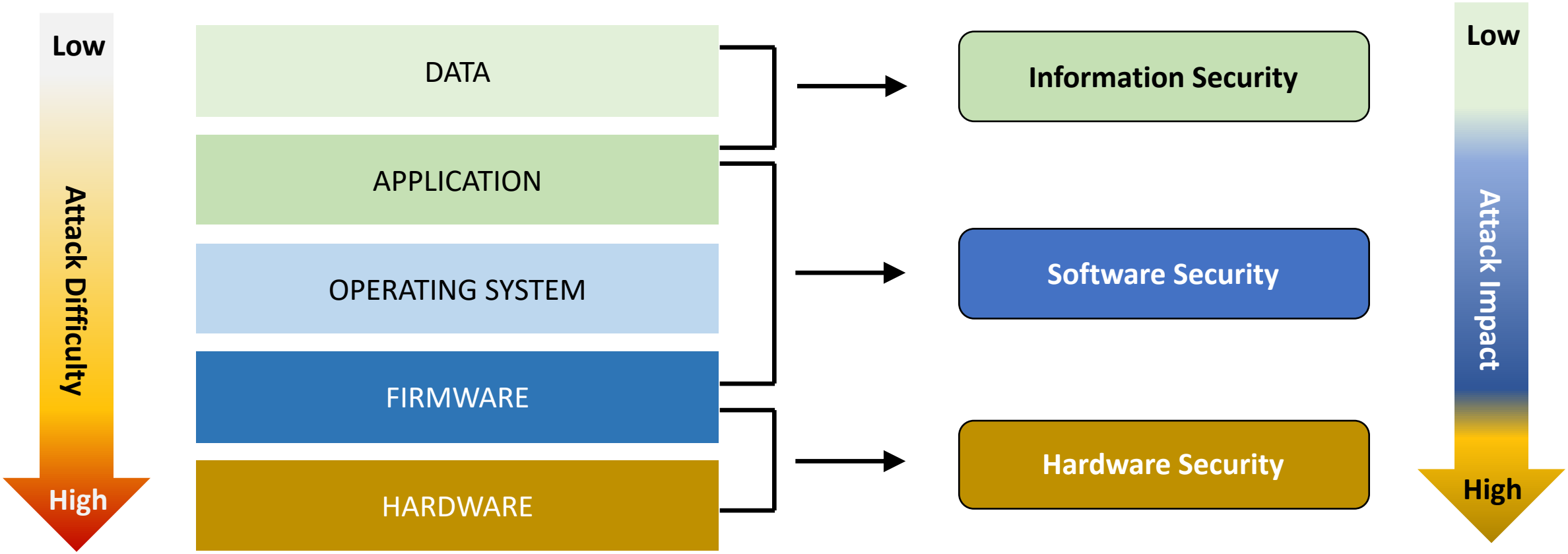
Computer Security Classes

- Software Security
  - Malicious attacks on the software, exploiting different implementation vulnerabilities (e.g., inconsistent error handling, buffer overflow)
  - Techniques to ensure reliable software operation in presence of potential security risks
- Information Security
  - General practice of providing confidentiality, integrity, and availability of information through protection unauthorized access, use, modification, or destruction
- Hardware Security
  - Attacks and protection of hardware and serves as the foundation of system security
  - Provides trust anchor for other security elements of the system

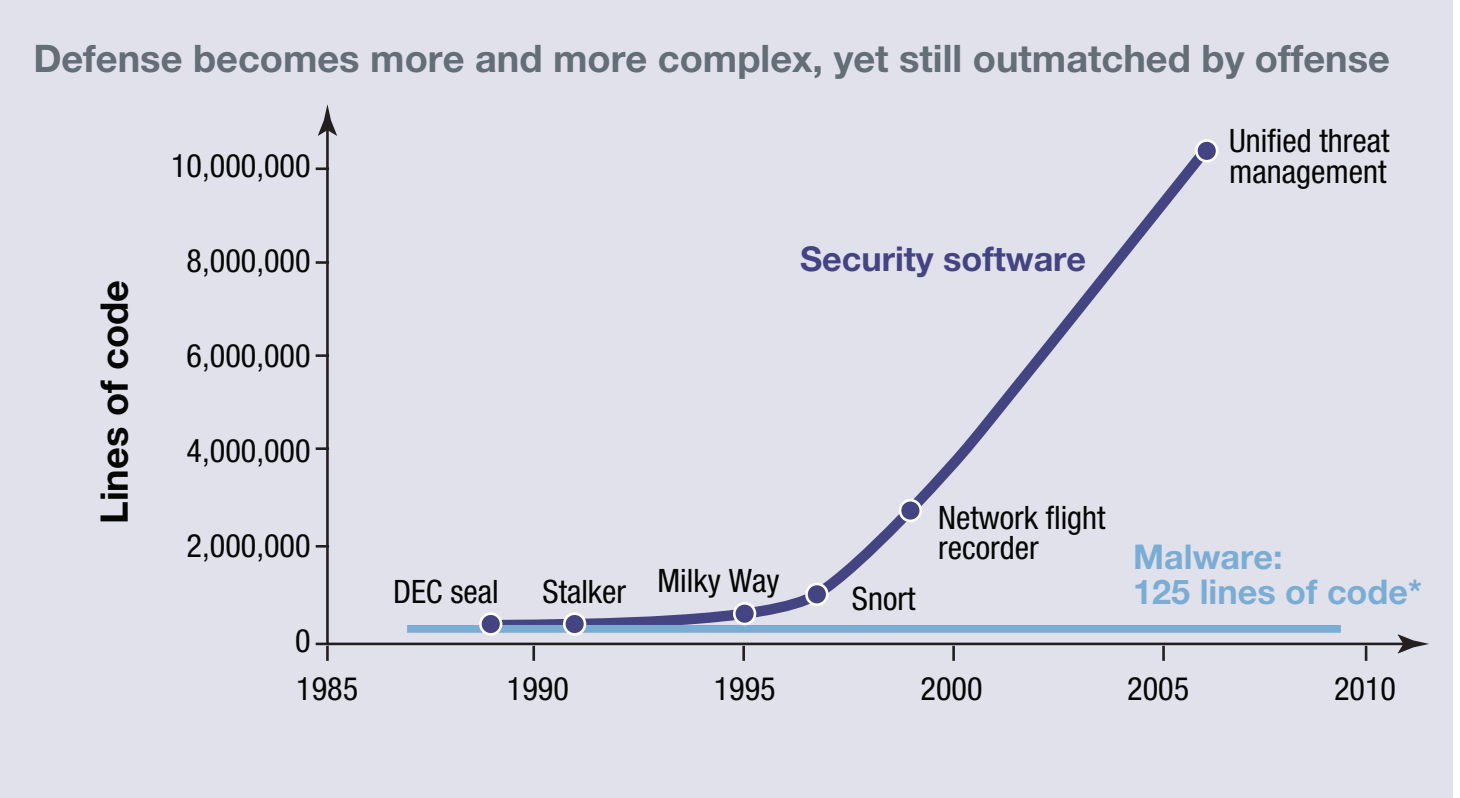
Computing Classes

- Traditionally, we have two broad computing classes
- General-purpose systems
  - Desktop, laptop, and servers
  - Complex and optimized architecture, versatile and easily programmable, and lend itself for a diverse use-case scenario
- Embedded systems
  - Digital cameras, home automation devices, wearable devices, biomedical implants
  - Highly customized design, tight hardware-software integration, and unique use-case constraints
- Emerging compute class
  - Internet-of-things (IoT) and Cyber-physical systems

Computing System Layers



Why Hardware Level Security?



Source: Defense Advanced Research Projects Agency (DARPA)

Brief to Defense Science Board (DSB) Task Force (May 2011).

Data through 2010.

Electronic Hardware Layers

- System-level layer
  - Integration of the physical components
  - PCB, peripheral devices, and encasing
- Printed circuit board layer
  - Provides mechanical support and electrical connection to the electronic components
  - Multiple layers of insulation substrate (e.g., fiberglass) to allow power and signals connectivity among the components using conductive metal (e.g., copper) traces
- Active components layer
  - ICs, transistors, relays, and passive electronic components

Electronic Hardware Types

- Digital ICs/chips
  - Work on digital signals
- Analog/mixed-signal (AMS) chips
  - Work on analog or both types of signals
- ICs classification based on usage model and availability in the market
  - Application-specific integrated circuits (ASICs)
    - Have customized functionalities – e.g., signal processing, security functions
    - Specific performance targets
    - Not readily available in the market
  - Commercial off-the-shelf (COTS) ICs
    - Flexible and programmable features to support diverse system design needs
    - Readily available in the market
    - Examples - field programmable gate arrays (FPGA), microcontrollers, processors, data converters, etc.

What is Hardware Security?

- Information Security
  - Primary focuses on information theory and cryptographic measures
  - It has been studied for years
- Software Security
  - Have also been extensively studied and analyzed
  - A large variety of solutions have been proposed
- Hardware Security
  - It is relatively new. Hardware has been traditionally considered immune to attacks
  - Hardware security relates to the hardware design, implementation, fabrication, validation, deployment to ensure secure and reliable operation of the software and system stacks.

What is Hardware Security?

- Hardware security deals sensitive assets (data, cryptographic keys, etc.) in hardware from malicious physical, software, and network, and providing an appropriate level of isolation between secure and non-secure data, code, in addition to providing separation between multiple user applications
- Two major topics in the the area of isolation
- Trusted Execution Environments (TEEs)
  - ARM's TrustZone, Intel SGX, Smasung Knox, etc.
- Protection of security-critical assets
  - Data, hardware states, access control, information flow protection

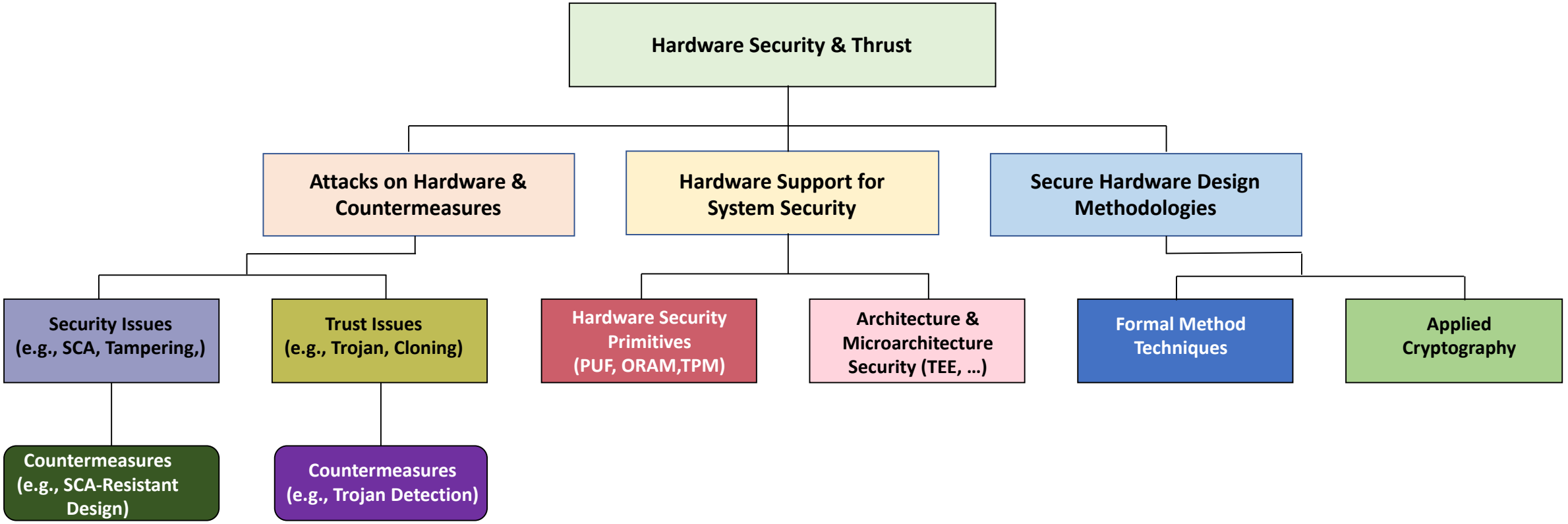
What is Hardware Security?

- Causes of hardware security and vulnerabilities
- Initially, hardware security focused on implementation-dependent vulnerabilities in cryptographic chips leading to information leakage
- Globalization of the chip manufacturing process
  - Distributed supply chain and complex electronic component provenance
  - Reduced control of the system manufacturer on the design and fabrication steps of the hardware has given rise to many growing security concerns
    - Malicious modification of ICs – Hardware Trojans
    - Untrusted design houses and foundries
- Hardware Attacks
  - Side-channel attacks where secret information of a chip can be extracted through measurement of analysis side-channels, that is, physical signals like power, signal propagation delay, and electromagnetic emission
  - IP Piracy and reverse engineering, counterfeiting, microprobing attacks on ICs, physical tampering of traces and components on the PCBs, Bus snooping in PCBs, Access to privilege resources through testing and debugging infrastructure

Hardware Security vs. Hardware Trust

- Hardware security issues arise from its own vulnerability to attacks
- Hardware trust issues arise from involvement of untrusted entities and components in the life cycle of the hardware
  - Untrusted IP and Compute-Aid Design (CAD) tool vendors, untrusted design, fabrication, test, and distribution facilities, and lack of traceable provenance
  - These parties are capable of violating the trustworthiness of the hardware component and system
- Trust issues often lead to security concerns

Hardware Security vs. Hardware Trust



Attacks, Vulnerabilities, and Countermeasures

- Attack Vectors
  - They are means or paths for bad actors (attackers) to gain access to the hardware components for malicious purposes – i.e., to compromise it or to extract secret assets store in the hardware
- Attack Surface
  - It is the sum of all possible security risk exposures.
  - It is the aggregate of know, unknow, and potential vulnerabilities
  - Chip-Level Attacks
    - Chips can be targeted for reverse engineering, cloning, malicious insertion, side-channel attacks, and piracy
  - PCB-Level Attacks
    - Design information of most modern PCBs can be extracted through relatively simple optical inspection (e.g., x-Ray Tomography)
    - PCBs are common targets for attacks, as they are much easier to reverse engineer and tamper than ICs
  - System-Level Attacks
    - Complex attacks involving the interaction of hardware-software components

Attacks, Vulnerabilities, and Countermeasures

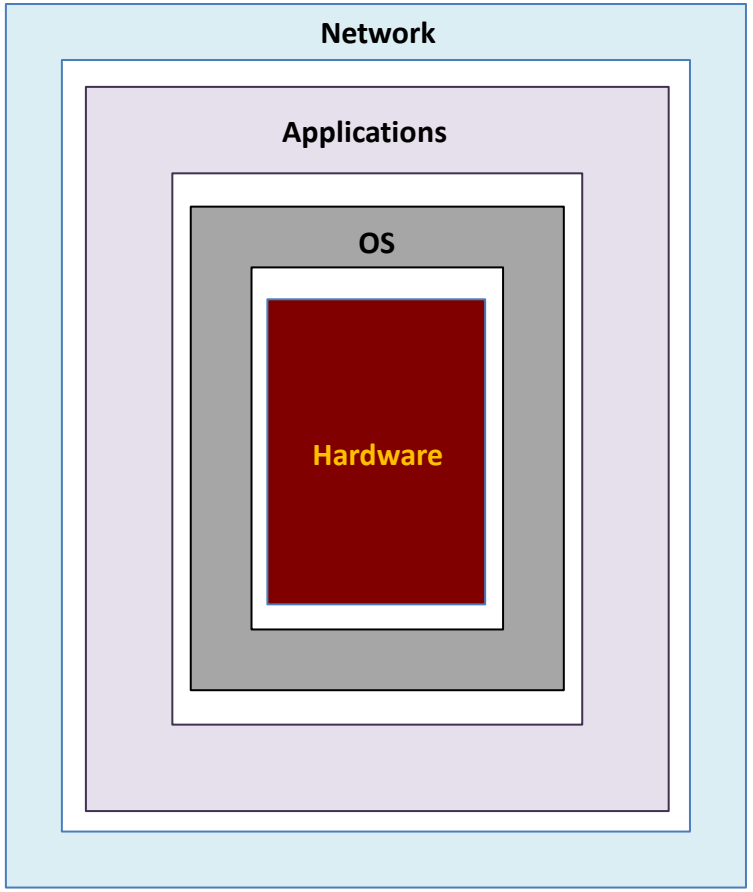
- Security Model
  - Attacks on hardware can take many forms
    - An attacker's capabilities, physical or remote access of the system, and assumptions of the system design and usage scenarios play essential roles in the techniques that can be used to launch an attack
    - In order to describe a security issue or a solution, it is important to unambiguously describe the corresponding security model
  - A security model should have two components
    - Threat Model
      - Describes the threats including, the purpose and mechanism of an attack
    - Trust Model
      - Describes the trusted parties or components

Attacks, Vulnerabilities, and Countermeasures

- Vulnerabilities
  - Functional Bugs
    - Most vulnerabilities are caused by functional bugs and poor design and testing practices
    - Weak cryptographic hardware implementation and inadequate protection of assets in a hardware
  - Side-Channel Bugs
    - They are implementation-level issues that leak critical information inside the hardware through different forms of side-channels
    - Attackers find these vulnerabilities by analyzing the side-channel signals during operation of the hardware
  - Test/Debug Infrastructure
    - IC engineers needs to have certain visibility into the internal states of the hardware to verify the correctness of operation and infrastructures for debugging the hardware
    - These infrastructure can be missed by attackers to mount attacks
  - Access Control or Information-Flow Issues
    - Composition and information flow among components when poorly designed can be leveraged by attackers

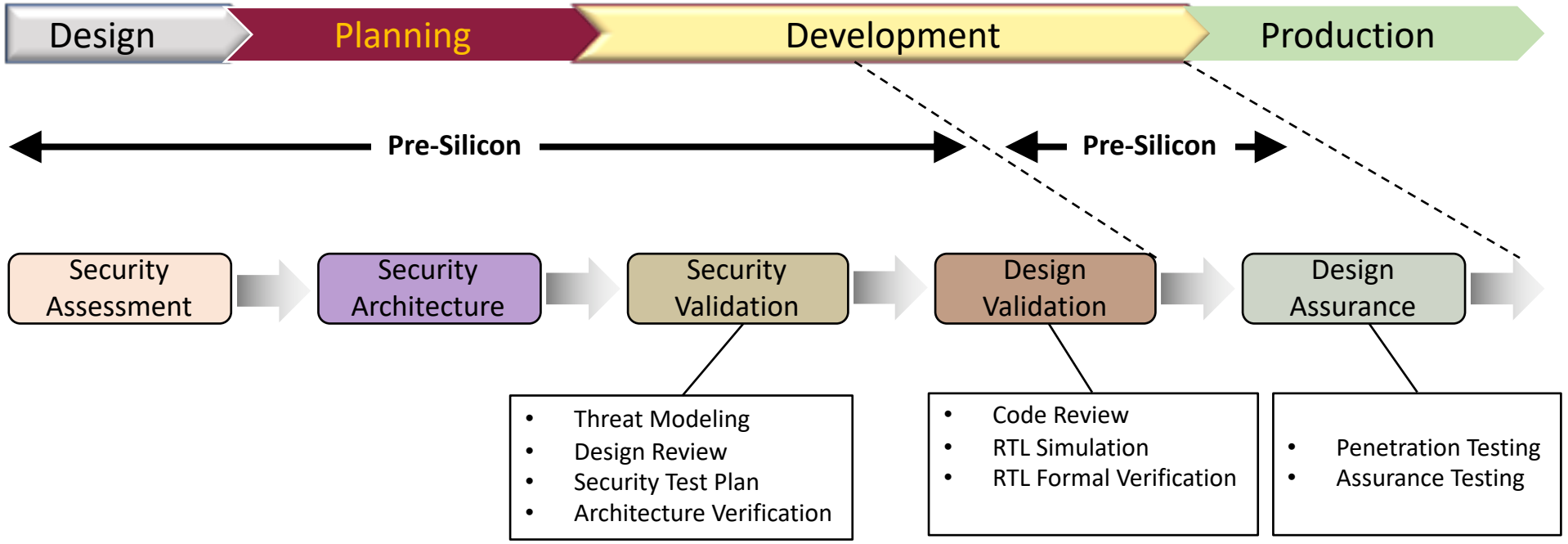
Attacks, Vulnerabilities, and Countermeasures

- Countermeasures
  - Circuit Level
    - Hardware obfuscation
  - Digital Design
    - IC watermarking
  - Datapath & Control
    - Self-repair and regeneration of datapaths
  - Component Level
    - Hardware security primitives (PUF, ORAM, RNG,…)
  - Architecture Level
    - Secure computing architectures
      - Secure heterogeneous system-on-chip (SoC) architectures

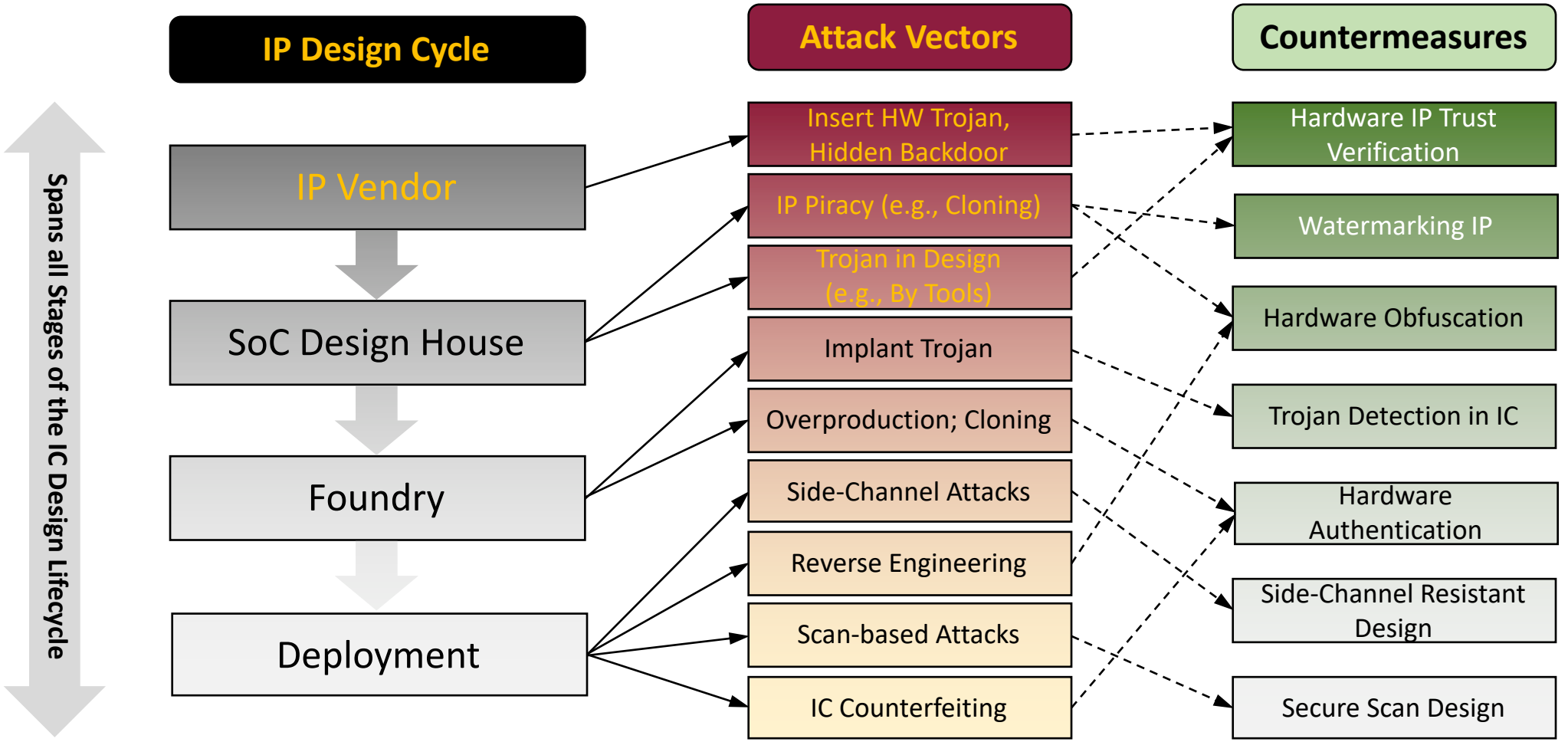


Attacks, Vulnerabilities, and Countermeasures

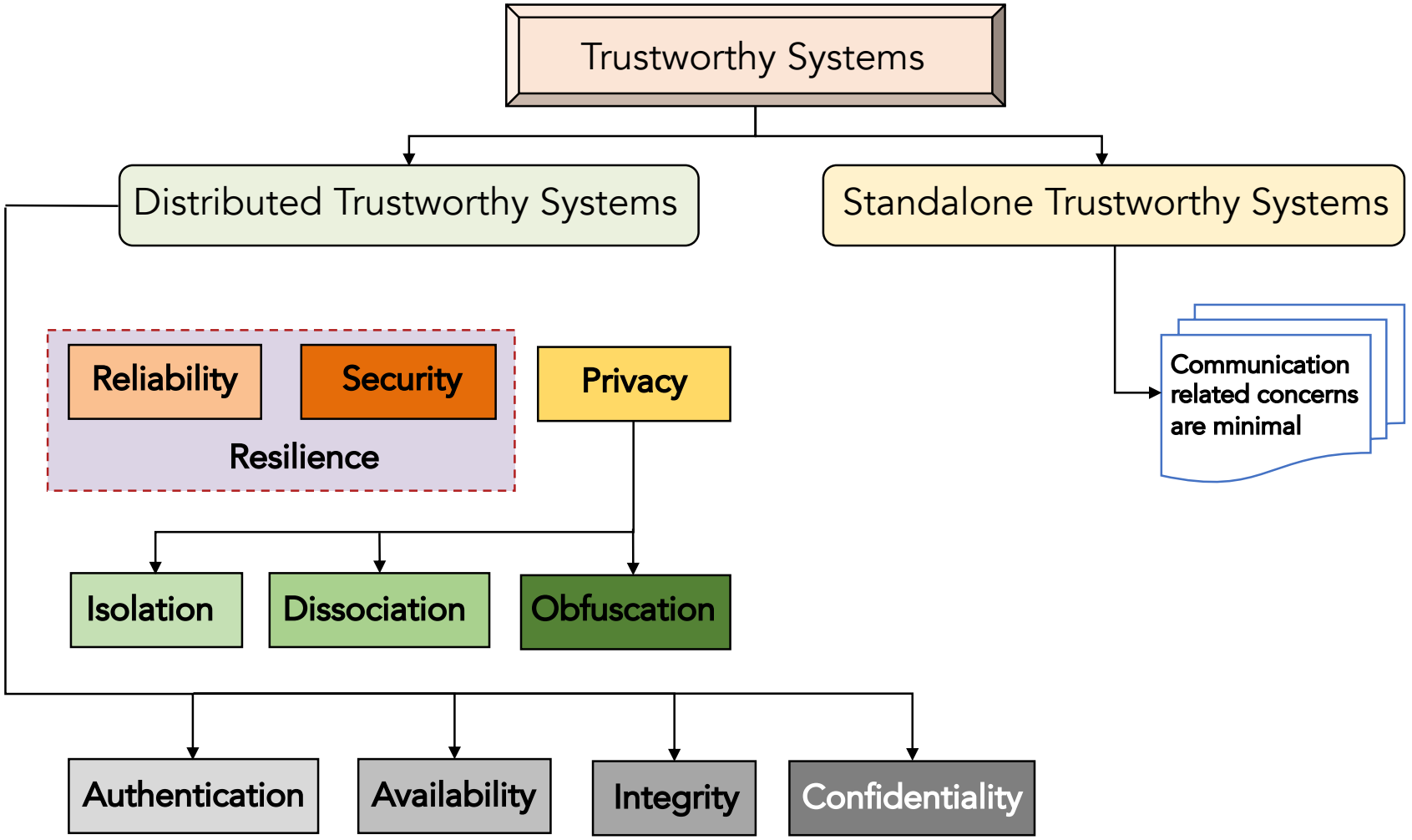
- Countermeasures
  - Design solutions
    - Design-For-Security (DFS) techniques have emerged as powerful countermeasures
      - For example, circuit obfuscation
  - Test and Verification Solutions
    - Use of formal methods in different stages of the design cycle



Attacks, Vulnerabilities, and Countermeasures



Trustworthy Hardware System Design



Functional Security Properties of Systems

Confidentiality:

- Secure channels / symmetric cryptography:
  - One-time pads
  - Stream ciphers
  - Block ciphers
    - Modes of operation
- Key exchange / key distribution:
  - Public / private cryptography
  - Forward secrecy
- Obfuscation:
  - Indistinguishability obfuscation
  - Deniable encryption
  - Program obfuscation
    - Opaque predicates
  - Hardware obfuscation
    - Anti-tamper, split manufacturing, …
- Private lookups, private metadata:
  - Mix networks, oblivious RAM, onion routing
- Isolation
  - Virtualization, containerization, sandboxing…
  - Secure architectures,
    - Trusted execution engines, secure enclaves
  - Formal verification
- Zero-knowledge proofs

Integrity:

- Message integrity:
  - Error correction codes
  - (Cryptographic) hash functions
- Privacy-preserving computation:
  - Multi-Party Computation (MPC):
    - Oblivious Transfer
    - Yao's Garbled circuits
    - Universal composability
  - Homomorphic Encryption (HE)
  - Hardware Root-of-Trust (HRoT):
    - Physical unclonable functions, e-fuses
  - Federated Learning
  - Distributed Consensus:
    - Digital currency, private voting
- Software security:
  - Virtual memory, file system permissions
  - App signing, sandboxing
  - Control flow integrity:
    - Shadow stacks
  - Buffer overflow protection:
    - ASLR, stack canaries
  - Malware detection:
    - Antiviruses, malware signatures
    - Hardware performance counters

Authentication:

- Asymmetric cryptography
  - One-way functions, trapdoor functions
  - Key exchange / key distribution algorithms
  - Digital signatures
- Public key infrastructure:
  - Web of trust
  - Certificate authorities, root certificates, self-signed certificates
- Passwords, biometrics, …
- Password-based key derivation

Authorization:

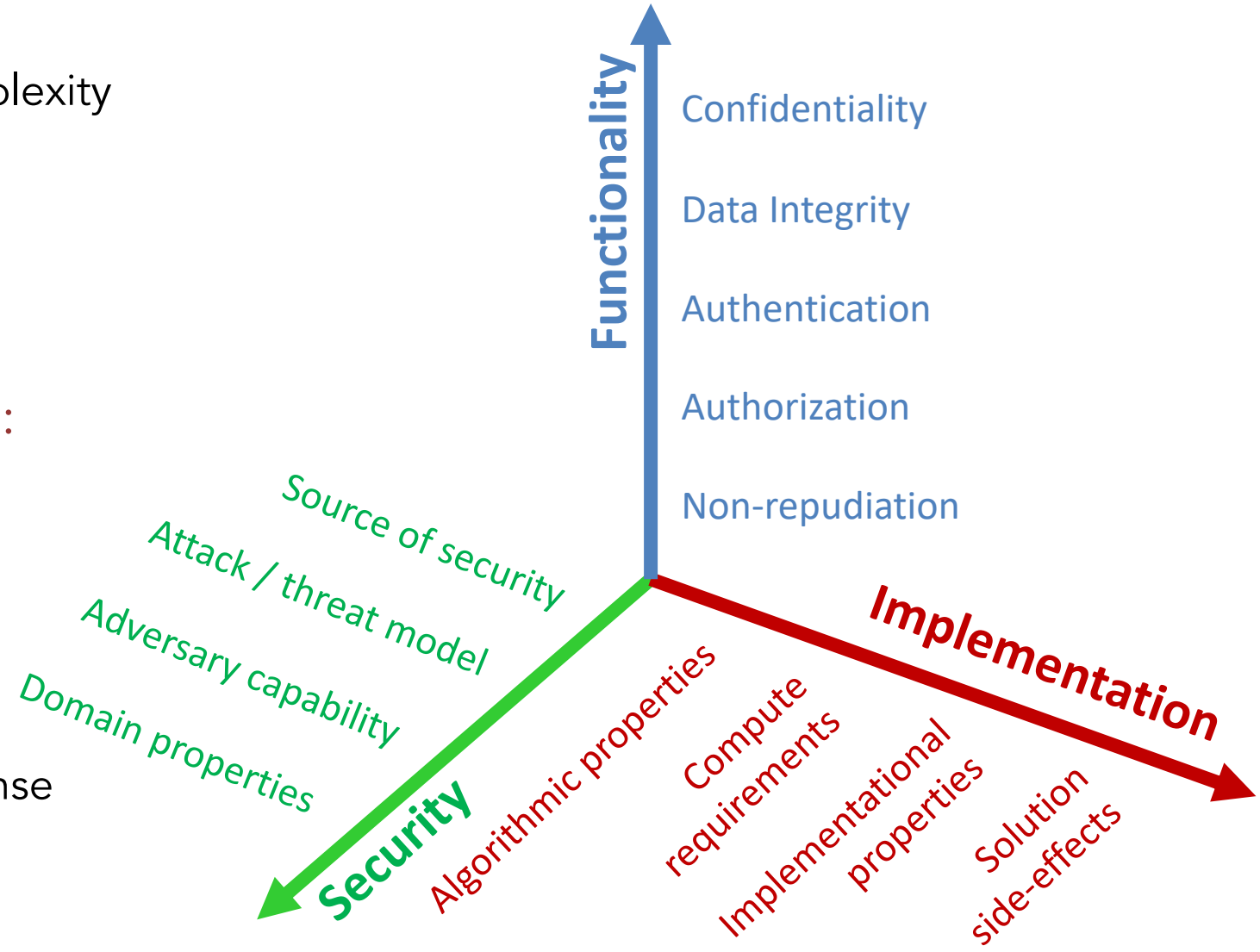
- Access Control Lists
- Role-Based Access Control
- Capability-Based Security

Non-repudiation:

- Digital signatures
- Commitment schemes
- Message authentication codes
- Deniable encryption, undeniable signatures

Security Concepts: Implementational Properties

- Implementational properties:
  - Algorithmic properties:
    - Computational / space complexity
    - Strong / weak scaling
  - Compute requirements:
    - FLOPS
    - Memory
    - Network bandwidth
  - Implementational properties:
    - Throughput
    - Latency
    - Power & area
    - Error correction, noise robustness
  - Solution side-effects:
    - Side-channel attacks & defense



Next Class

- Classic and Modern Encryption Algorithms