Foundations of Cybersecurity
C and C++ Secure Coding

Gianluca Dini
Dept. of Information Engineering
University of Pisa
Email: gianluca.dini@unipi.it

Version: 2022-03-06

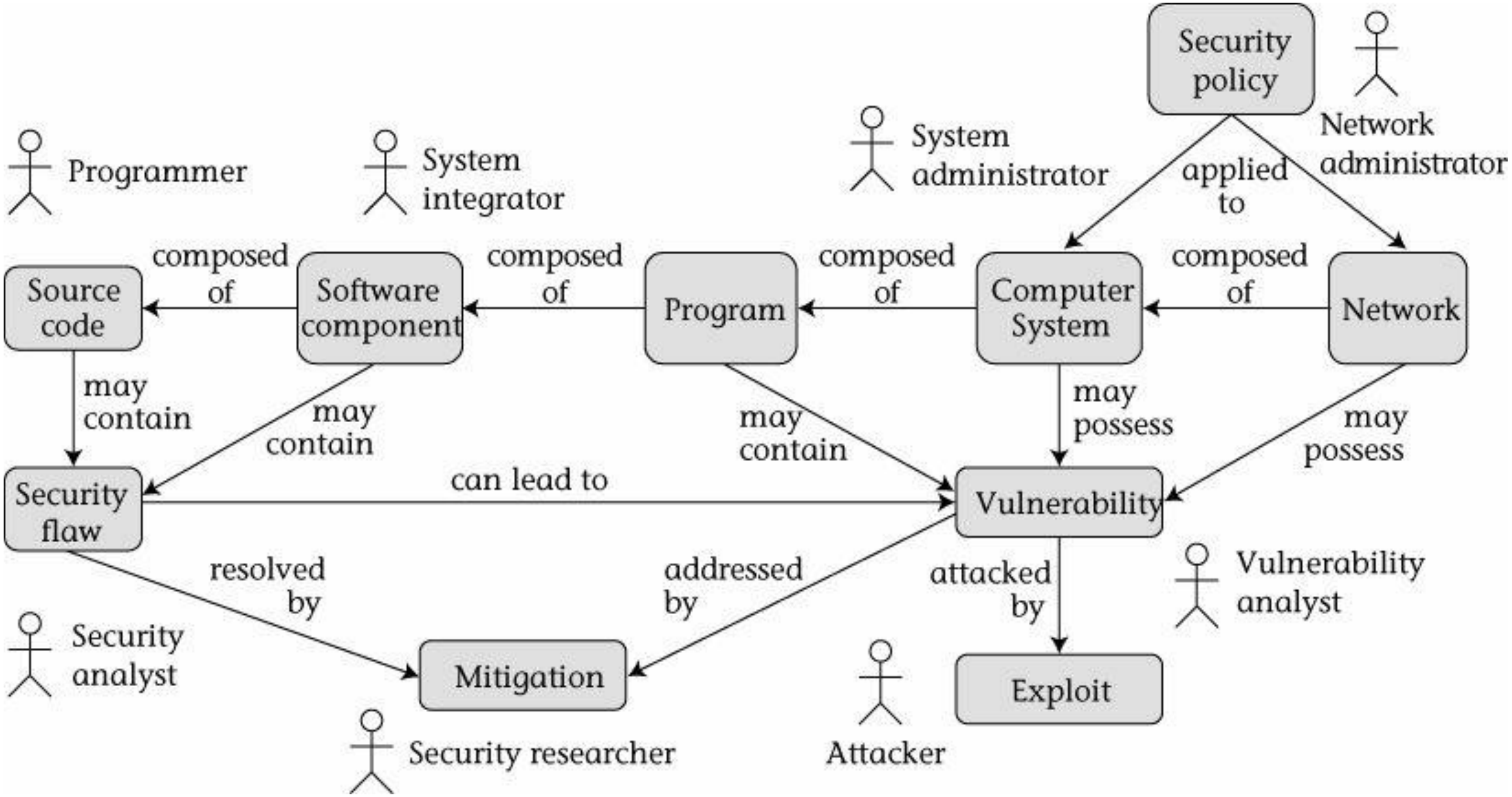
Credits

- These slides come from a version originally produced by Dr. Pericle Perazzo

Secure coding - definitions

SECURITY CONCEPTS

Concepts, actors, and relationships

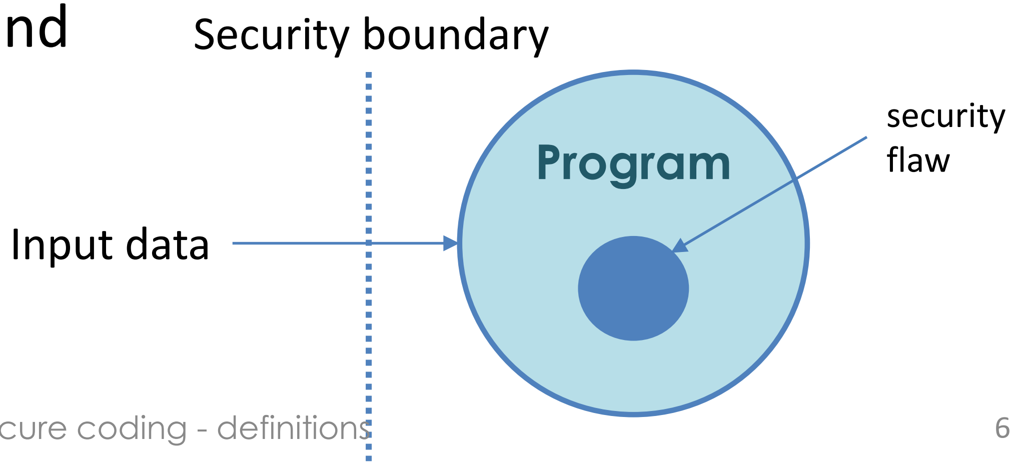


Security Policy

- A set of rules and practices that specify or regulate how a system or organization provides security services to protect sensitive and critical system resources
- Explicit or implicit

Security flaws and vulnerabilities

- A *security flaw* is a *software defect* that poses a potential security risk
- A *vulnerability* is a set of conditions that allows an attacker to violate an explicit of implicit security policy
  - Not all security flaws lead to vulnerabilities
  - Programs, systems and networks



Security boundary

Input data

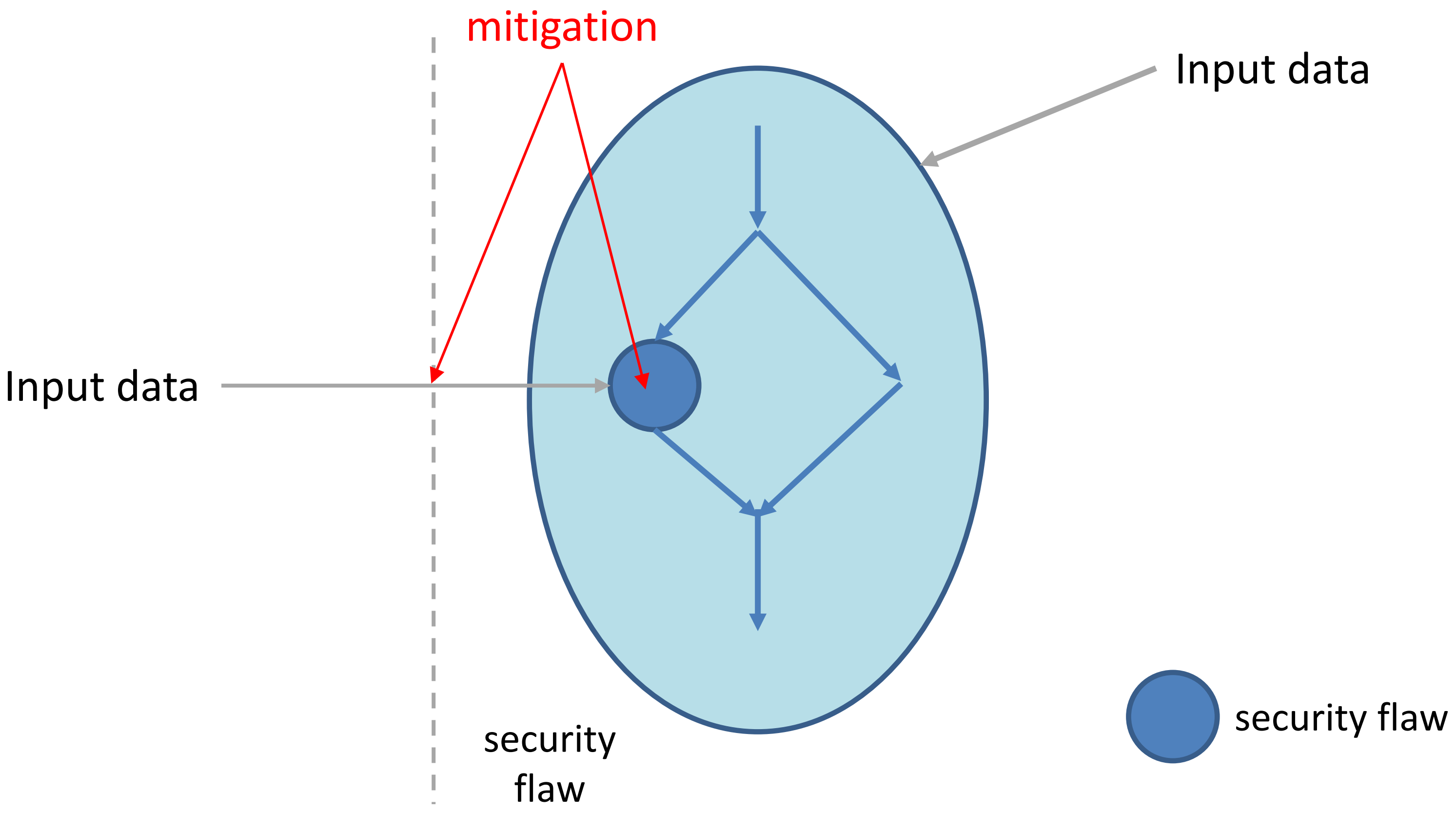
Program

security flaw

Exploits and mitigations

- Exploit is a technique that takes advantage of a security vulnerability to violate an explicit or implicit security policy
  - Proof-of-concept exploit vs malware
- Mitigations are methods, techniques, processes, tools, or runtime libraries that can prevent or limit exploits against vulnerabilities
  - Aka countermeasures or avoidance strategies
  - Solution for a software flaw vs a workaround to prevent exploitation of a vulnerability

Exploits and mitigations



Secure coding

A BUNCH OF DEFINITIONS

Secure Coding

- Programming errors which caused the most common/dangerous vulnerabilities
- Remediation best practices
- Risk assessment
  - Exploitation probability
  - Impact
  - Remediation cost
- Objectives:
  - Protect customers
  - Limit patches

Secure coding - definitions

TAINT ANALYSIS

Secure Coding

- Strongly language-dependent
- C/C++ are particularly error-prone
  - Intended to be lightweight
  - Power-to-the-programmer philosophy
- C/C++ still broadly used
  - Embedded devices
  - High-load servers
  - Legacy code

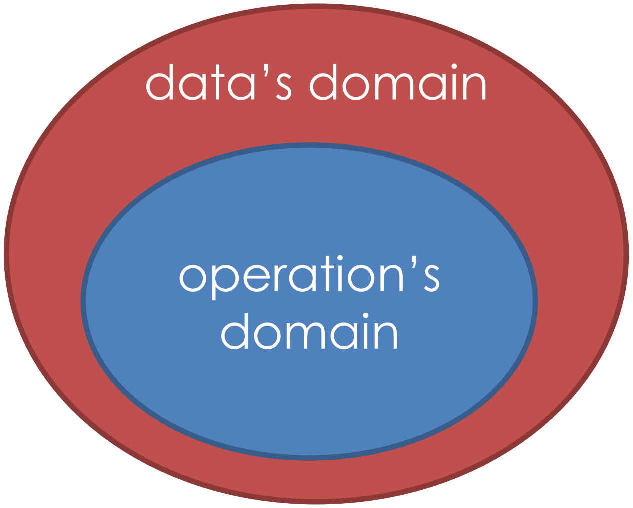
Undefined Behavior

- Undefined behavior: C/C++ gives no requirement
  - Out-of-bound buffer access
  - Null pointer dereferencing
  - Signed integer overflow
- Unspecified behavior: C/C++ gives multiple possibilities
  - Argument evaluation order in function calls
- Unexpected behavior: well-defined behavior unanticipated by the programmer

Taint Analysis Terminology

- Tainted data
  - Not sanitized data from an external source
  - Operations on tainted data gives tainted data
- Restricted sink
  - Operand/argument with domain smaller than its type domain
- U3B happen when tainted is given as input to a restricted sink

restricted sink:



Sanitization

- Sanitization removes taint from data
  - By replacement: replace out-of-domain values with in-domain values
  - By termination: terminate execution path