Your Hardware has Bugs -

Managing Hardware Vulnerabilities

Peter Mell, Computer Security Division

Irena Bojanova, Software and Systems Division

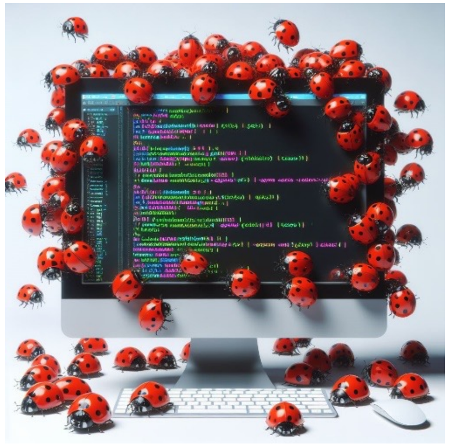
Computer Scientists

National Institute of Standards and Technology

Historically hardware was viewed as an ‘immutable root-of-trust'

- Is this because greater care is taken in designing hardware?
- Software has never achieved this
  - Large IT company had the slogan “Unbreakable”, “Can't break it, can't break in”, (2002)
- But…
- Hardware is made with code and contains code
  - Hardware design (Verilog, VHDL), code etched on a chip
  - Hardware contains microcode and firmware
  - Worse than software, hardware often can’t be updated/patched/fixed
- Hardware is more than software; it can have physically related vulnerabilities
- Hardware is complex
  - Systems on a chip (SOCs) contain many integrated modules, designed independently
  - Continually increasing functionality in products ensures instability/insecurity

Software Vulnerability Landscape

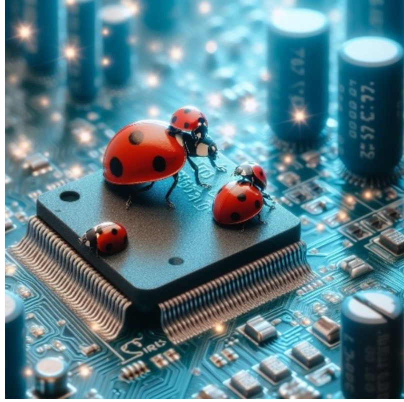


- There are estimates for between 15 and 50 bugs per 1000 lines of **delivered** code
  - Some of those bugs will affect security
- 28,000 security vulnerabilities published in 2023

- Software has a mature ecosystem to handle the vulnerabilities
  - Common Vulnerabilities and Exposures (CVE)
  - National Vulnerability Database (NVD)
  - Common Vulnerability Scoring System (CVSS)
  - Exploit Prediction Scoring System (EPSS)
  - Known Exploited Vulnerabilities (KEV)
  - Common Weakness Enumeration (CWE)
  - NIST Bugs Framework

- 130 weakness types (i.e., CWEs) cover 94% of vulnerabilities (i.e., CVEs)
  - These are ways in which software has security failures

Hardware Vulnerability Landscape



- 104 hardware weakness types identified (i.e., CWEs)
- Only half of these have observed examples (i.e., CVEs)
  - Hardware producers don't always disclose, in part because they can't patch
  - Many may have surfaced during development and were fixed pre-production
- We know of only 131 published hardware vulnerabilities (i.e., CVEs)
- Only 3 of the hardware weaknesses (CWEs) overlap with software weaknesses
  - Is hardware security really that different from software?
  - 101 of the hardware weaknesses are hardware specific
  - Do the rest of the 127 software weaknesses apply (130 software vulns – 3 overlap)?
    - After all, hardware contains code

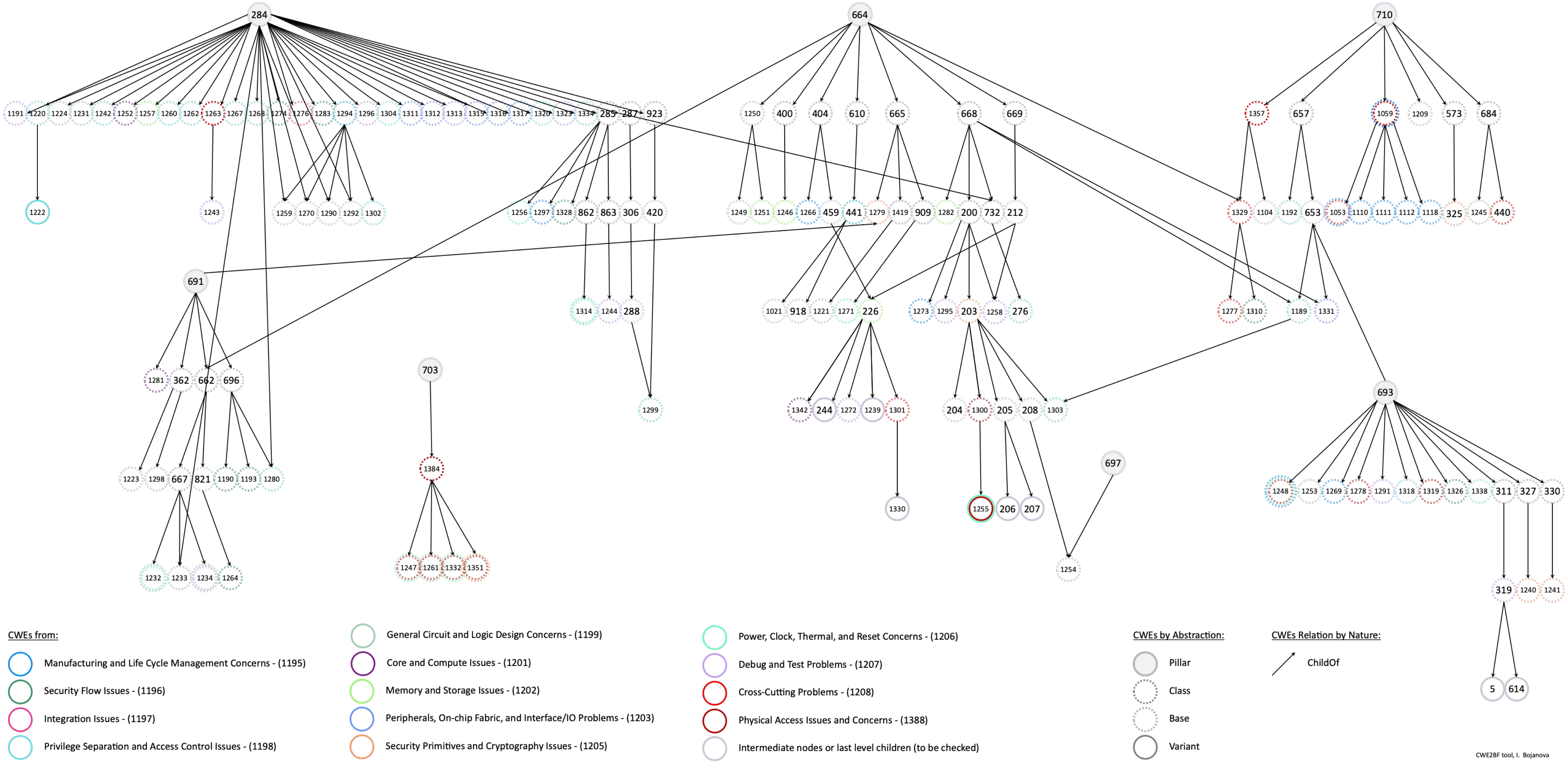
- Trust-Hub Vulnerability Database (University of Florida)
  - 38 physical attacks
  - 23 vulnerabilities (look more like weaknesses)

Hardware Weakness Categories

This is mostly 'where' they occur

1. Core and Compute Issues (CWE-1201)
2. Cross-Cutting Problems (CWE-1208)
3. Debug and Test Problems (CWE-1207)
4. General Circuit and Logic Design Concerns (CWE-1199)
5. Integration Issues (CWE-1197)
6. Manufacturing and Life Cycle Management Concerns (CWE-1195)
7. Memory and Storage Issues (CWE-1202)
8. Peripherals, On-chip Fabric, and Interface/IO Problems (CWE-1203)
9. Physical Access Issues and Concerns (CWE-1388)
10. Power, Clock, Thermal, and Reset Concerns (CWE-1206)
11. Privilege Separation and Access Control Issues (CWE-1198)
12. Security Flow Issues (CWE-1196)
13. Security Primitives and Cryptography Issues (CWE-1205)

NIST Research: Hardware Weakness Hierarchies Looking at 'how' they are exploited



CWE2BF tool, I. Bojanova

Looking Towards the Future

- Much of the software vulnerability management infrastructure can support hardware vulnerabilities
- Progress is being made in this direction
- Hardware contains weaknesses not found in software
    - Importance of the physical dimension
- Hardware should share more than three weaknesses with software
    - but this has yet to be determined

Presentation / Speaker Information

Title: Your Hardware has Bugs - Managing Hardware Vulnerabilities

Abstract: Hardware has historically been viewed as trustworthy while software has never achieved this. However, hardware is vulnerable; it has additional complications and weaknesses not present in software. Software, while massively vulnerable, is supported by an extensive management ecosystem. Hardware can leverage this ecosystem and work is being done to promote such support.

Peter Mell Biography: Peter Mell is a computer scientist with the National Institute of Standards and Technology. He has conducted computer security research for over 25 years and has over 65 academic publications. He created and managed the National Vulnerability Database (NVD). He assisted in the development of two versions of the Common Vulnerability Scoring System (CVSS). He is currently investigating hardware weaknesses, how they can be exploited, where they occur, and what damage can be done.

Irena Bojanova Biography: Irena Bojanova is a computer scientist at the National Institute of Standards and Technology. She has conducted formal methods and computer security research for over 35 years and has over 80 academic publications. She invented and is creating the NIST Bugs Framework (BF), which goal is formalization of software security weaknesses and vulnerabilities. She is the Editor of the Cybersecurity Department of IEEE *IT Professional* magazine and a member at large of the IEEE Computer Society Publications Board Executive Committee.