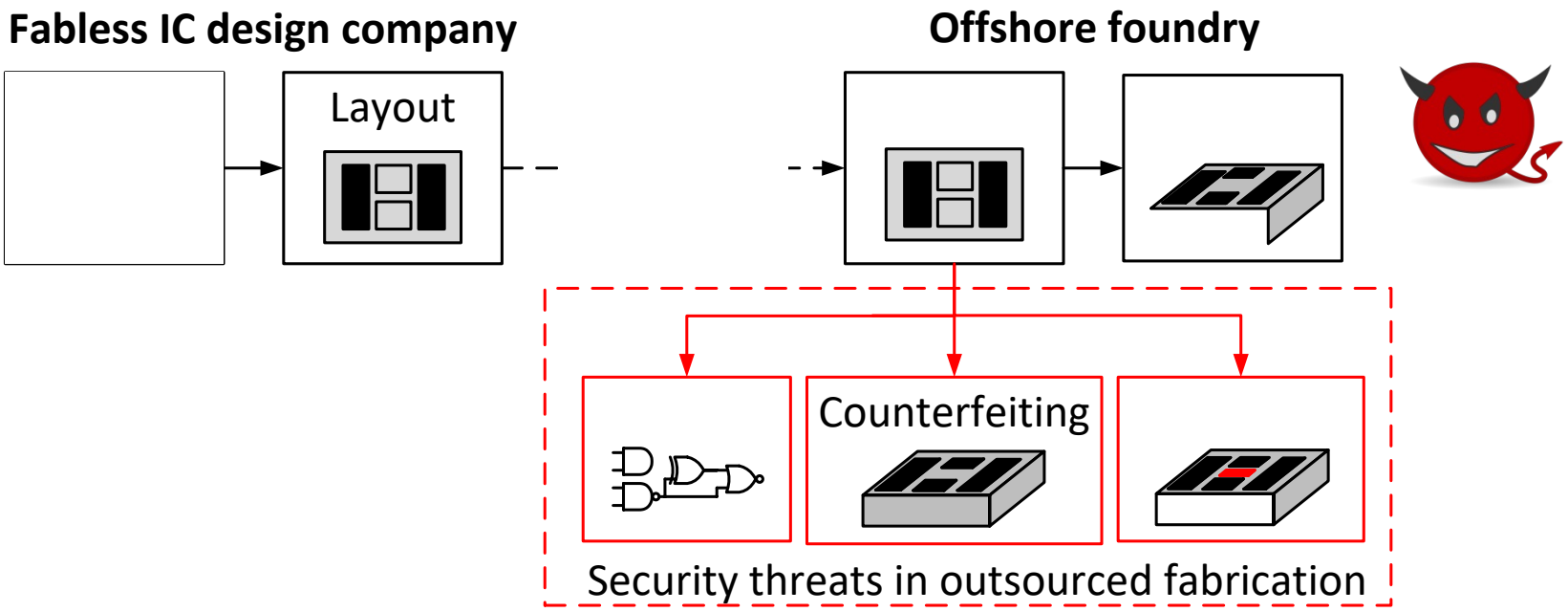
Verification and Validation for Hardware Security Constructs

A. Srivastava
Director Institute for Systems Research
Professor Dept. of ECE
University of Maryland

Hardware Security

- Design Obfuscation
- Trojan Detection and Mitigation
- Side Channel Attacks



Motivation

- Hardware security has become a significant technical problem with significant impact on DoD and Commercial systems at varying levels.

- There is a huge dearth of sound hardware security metrics and provable methods to "certify" the security guaranteed by current approaches.

Example: Design Obfuscation

- **Outsourced IC Fabrication**
  Access advanced semiconductor technology at low cost
- **Security Threats**
  Attacker: untrusted offshore foundry
  Knowledge: layout files of the outsourced design
  Goal: IP piracy, counterfeiting

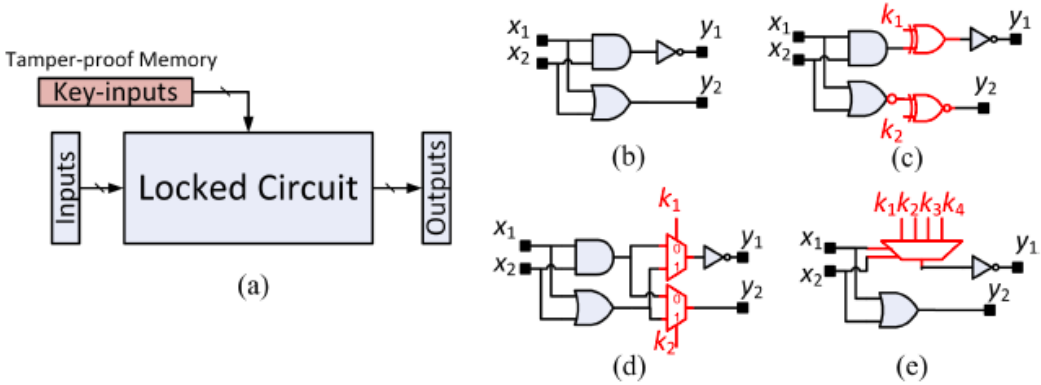


Fig. 1. Logic locking techniques: (a) Overiew; (b) An original netlist; (c) XOR/XNOR based logic locking; (d) MUX based logic locking; (e) LUT based logic locking.

Traditional Metric: Number of Unique Function Incorporated by Keys, Error Rate

A Scientific Approach Towards Metrics for Design Obfuscation

- Formal categorization of attack surfaces and attackers capabilities.
  - Does the attacker just have GDSII or also a working system procured from the market
  - How knowledgeable is the attacker, How capable is she w.r.t. access to functional/circuit analysis tools and equipment.

- Triage for each security solution.
  - Just because a security solution is broken does not make it irrelevant.
  - May still be applicable in low cost, low attacker capability scenarios.

Metrics for SOTA Obfuscation Technologies: SHIP and SAHARA Experience

There are various attack scenarios of interest.
In each scenario, the attacker has the obfuscated netlist.

- **Scenario 1**: The attacker does not have any information about the original design.
- **Scenario 2**: The attacker has a knowledge library
  - redacted design may or may not be from this library.
- **Scenario 3**: The attacker has a working chip ("oracle") from which the correct input-output pairs can be queried.
  - Internal flip flops accessible through test structures
  - More sophisticated imaging based attacks may be feasible
    - Applicable in nation state attackers.

- Scenario 3 is most researched in literature (e.g. SAT attack), but Scenarios 1 and 2 are not.

Scenario 2: The Adversary is Knowledgeable..But No Working System

- The untrusted fab is a knowledgeable adversary
- Knowledge representation: The adversary has a library of designs (e.g. open-source or previously seen)

  Can be used to "learn" the types of circuits structures typically used in designs

  Knows which Boolean functions from scenario 1 are more likely than others, further reducing uncertainty

  The exposed/unredacted portion can be used to "match" with each of the library module

  - Logic Equivalence Checking (LEC) tools can compare redacted netlists with those in the library
  - Functional analysis and structural mapping can also be used
  - Launch SAT attack with knowledge library elements with high degree of fit to decipher potential bitstreams

Scenario 2: The Adversary is Knowledgeable

Scenario 1: Zero-Knowledge Attacker

Unique Functional Distribution

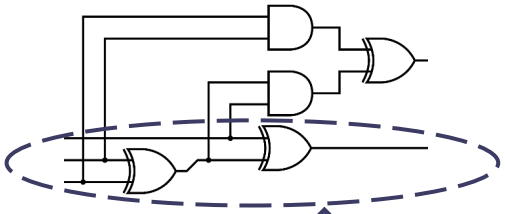
Redacted Design

Knowledge Library

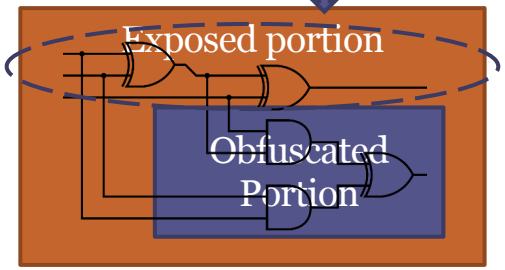
Scenario 2: Knowledgeable Attacker

- Reduced Unique Functional Distributions
- Matching Exposed Portions to Identify Potential Fits with Prior Knowledge
  - LEC Tool and Functional Analysis
  - Launch SAT attack with knowledge library elements with high degree of fit

Library module example



Matched parts

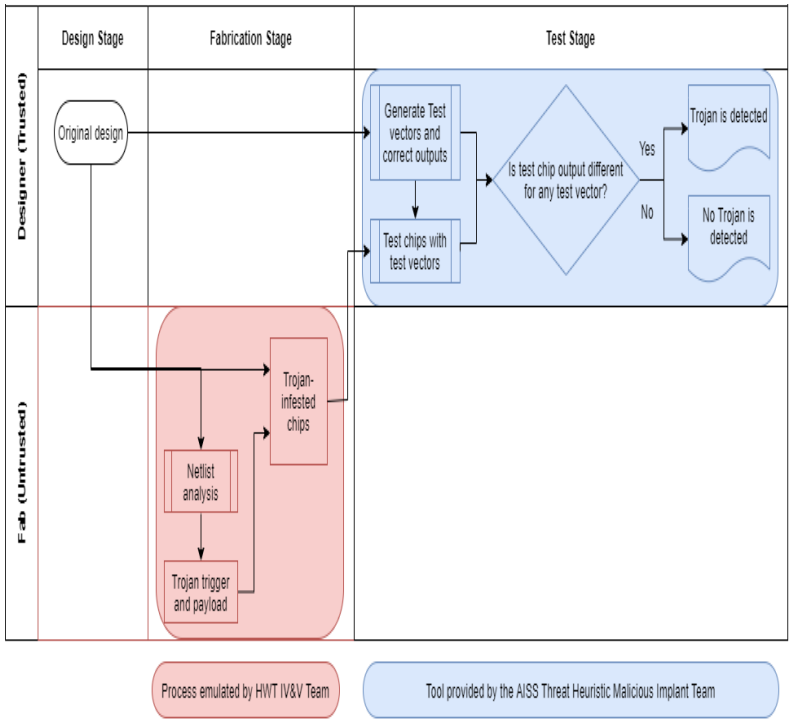


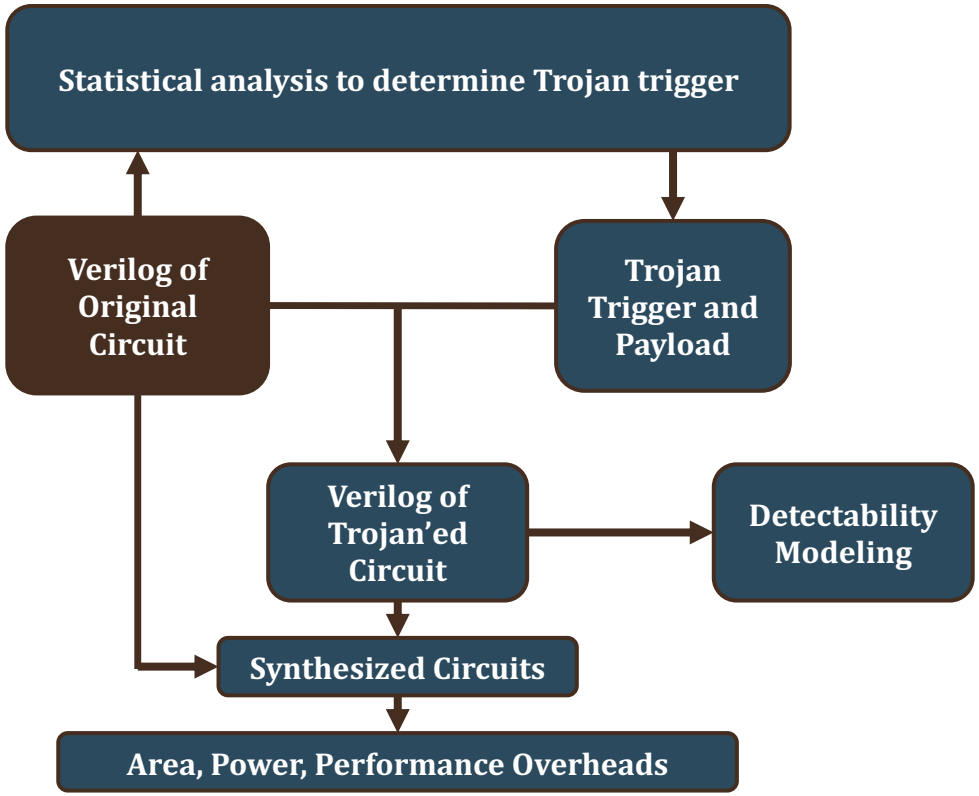
Redacted module example

RTL Design Library	Fully redacted design		
	SHA256	SHA512	GPS
SHA256	**Mapped**	Unmapped	Unmapped
SHA512	Unmapped	**Mapped**	Unmapped
GPS	Unmapped	Unmapped	**Mapped**

- Knowledgeable Attacker: Library +LEC
- Such mapping is **NOT** affected by:
  - Changing the names of DFFs and I/O pins
  - Redacting more gates in the periphery of the redacted module

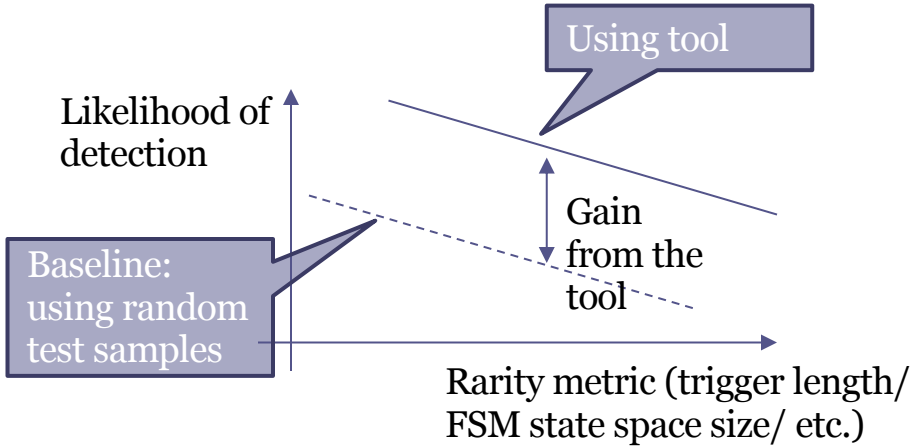
Vulnerability and Detectability Analysis for Trojan Mitigation Methods: DARPA AISS





Vulnerability and Detectability Analysis for Trojan Mitigation Methods: DARPA AISS

1. Efficacy
   1. Capability of detecting each type of HWT
      - Stress test with HWTs implemented by IV&V Team.
      - For each type: test a spectrum of implementations (e.g. trigger rarity) and observe how the detectability changes.
   2. False positive rate
      - Test-based detection has 0% false positive in theory.
2. Usability
   1. Ease of use
   2. Documentation of tool usage



	HWT free design	HWT infested design
HWT detected	**False Positive**	True Positive
HWT not detected	True Negative	False Negative

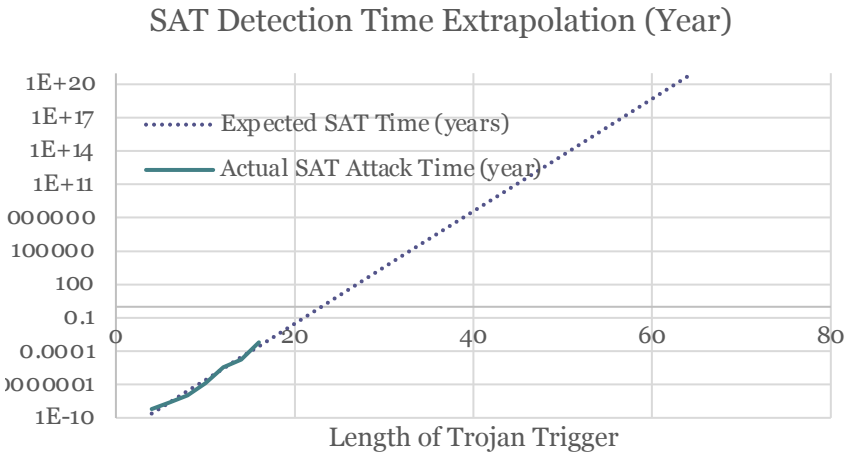
$$\text{False positive rate} = \frac{False\ positive}{Total\ HWT\ free\ designs}$$

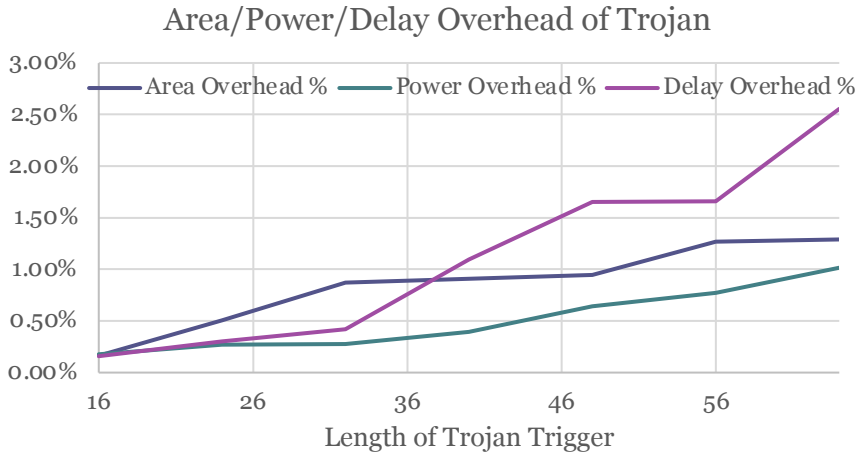
Vulnerability and Detectability Analysis for Trojan Mitigation Methods: DARPA AISS

Benchmark: 32-bit multiplier (~10000 gates)

Testing-based Trojan detection:

Trojan Trigger Length	How many times is the Trojan detected?	
	Test with 10K rare-value-based samples	Test with 1M random samples
16	0	10
24	0	0
32	0	0
40	0	0
48	0	0
56	0	0
64	0	0





Error impact of Trojan Payload:
53.1% Hamming Distance when triggered.

Summary

- A strategic layered approach to vulnerability analysis is needed.
- Different levels of access and control from the attacker need to be modeled.
- Sound mathematical constructs and formulations.
- AI based models for attackers knowledge.