Hardware CWE™ Special Interest Group (SIG)

Chair: Bob Heinemann (MITRE)

Co-Chair: “Manna” Parbati Kumar Manna (Intel)

MITRE Team: Steve Christey Coley, Alec Summers

MITRE

November 14, 2025

CWE is sponsored by U.S. Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA).

Agenda

REMINDER: This meeting is being recorded.

1	RTL Weaknesses Ad-Hoc Working Group Formation	Group Discussion Bob to lead	50 min

Housekeeping

- **Schedule:**
  - **Next Meeting December 12:**
    - **12:30 – 1:30 PM EST (16:30 – 17:30 UTC)**
    - **Microsoft Teams**
- **Contact: cwe@mitre.org**
- **Mailing List:** *hw-cwe-special-interest-group-sig-list@mitre.org*
- **Minutes from previous meetings available on our GitHub site:**
  - **https://github.com/CWE-CAPEC/hw-cwe-sig**

Announcements

- **CWE 4.19 scheduled for release on 12/4/2025**
- **CWE Content Development Repository (CDR) is now fully public.**
  - Enables the broader community to view, track, and contribute to entries submissions.
  - Content suggestions begin with the CWE Submission Form.
  - CDR can be accessed here:
    - https://github.com/CWE-CAPEC/CWE-Content-Development-Repository/

Meeting Structure

- **Topics**
  - RTL Ad-Hoc WG Motivation (10 min)
  - Ad-Hoc Working Group Model (5 min)
  - Lightning Talks (20 Min)
  - Open Discussion (10 min)
  - Call for members to join working group (5 min)

CWE is sponsored by U.S. Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA).

Motivation to form an Ad-Hoc RTL Working Group

- **Within the last year there has been demonstrated interest from our community in exploring issues with RTL.**
  - Oct/Nov 2024: Joerg Bormann did an in-depth discussion on “Security Issues Arising from Hardware Design”.
  - May 2025: Raghul Saravanan and Sudipta Paria made a submission titled ”Logic Transformation Discrepancy during RTL Synthesis” (CDR #152).
  - Oct 2025: Jagadish Nayak expressed interest in a submission about poorly written RTL code that produces stateful elements by mistake.
- **With multiple occurrences of this topic, we felt it was time to have an activity focused around on how to handle these issues within CWE.**
- **Group discussion might help, like it did for transient execution and MIHW**

Framing the RTL Difficulties from a "Weakness" Perspective

- **Current RTL-related descriptions say that weaknesses (insecure behaviors) can appear when using RTL – but there might be multiple weaknesses**
  - "missing signals in sensitivity lists"
  - "a malicious user exploiting/injecting bugs"
- **Security/vulnerability implications are not always clear**
- **Some parts of CWE are similar; but how does RTL fit?**
  - CWE-1038: Insecure Automated Optimizations
    - CWE-1037: Processor Optimization Removal/Modification of Security-critical Code (Spectre/Meltdown)
    - Child: CWE-733: Compiler Optimization Removal/Modification of Security-critical Code
  - CWE-1068: Inconsistency Between Implementation and Documented Design
  - CWE-440: Expected Behavior Violation
- **Is "RTL" a part of the "SDLC," and if so, is it more like a Mode of Introduction?**
- **Known CWE conceptual gaps**
  - Maintaining correct "state" - CWE-1250 (SW?), CWE-1431, CWE-1423, CWE-1245, others
  - Chaining relationships? (e.g., compiler removing an integer overflow check (CWE-733 creating CWE-190) enables buffer overflows (CWE-119) that the developer tried to avoid

Ad-Hoc Working Group Model

Provide a community-driven response to emerging hardware concerns that lead to concrete CWE outcomes.

Formation:

- **Identify a concern, gauge interest, call for volunteers, set kick-off meeting, establish a chair, set scope of work.**

Operations:

- **Weekly/bi-weekly calls + async work**

Workstreams:

- **Analyze issue, map to existing CWEs, update existing CWEs, propose new CWEs.**

Ad-Hoc Working Group Sunsets

Lightning Talks

Cycuity: CWE category for incorrectly written RTL?

1. **Cycuity Inc:** Focused on Hardware Security in Microelectronics

2. **Problem:** Incorrectly coded RTL causes a security weakness. Can this issue be identified early in the design cycle?

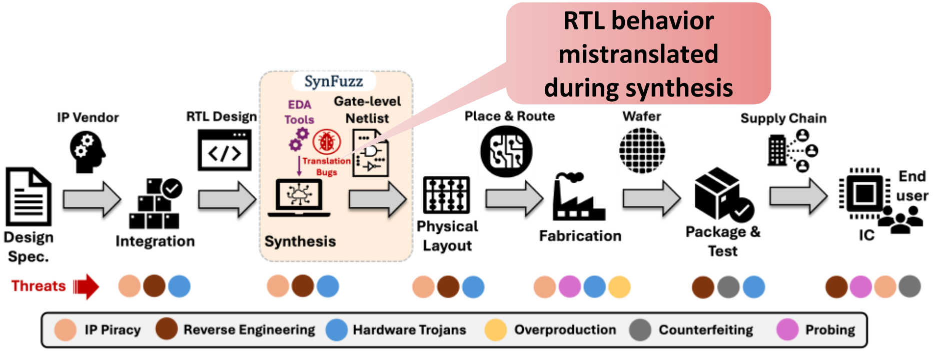
3. **Example:** The design has a secure register that can be written and read. Secure register information must be only present on the common bus when read signal is asserted.
   Missing else clause results in inferred latch and data from secure register stays on after read is de-asserted.

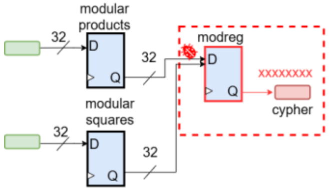
```
always @(posedge clk) begin
     if (reset) sec_reg <= 0;
     else if (write) sec_reg <= data_in;
end
always_comb  begin
      if (read) data_out = sec_reg; // No else clause
  end
```

4. **Working group output:**
   1. Identify and provide CWE IDs for security weaknesses caused by incorrectly written RTL
   2. Create CWE class that covers all such CWE IDs

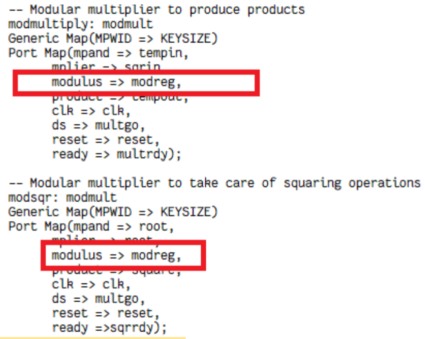
5. **Discuss:** Is it possible to create a Category CWE for variants of security weaknesses created by incorrectly written RTL?

Translation Bugs: Beyond RTL Verification





Multiple Drivers in RSA Crypto Core



```
-- Modular multiplier to produce products
modmultiply: modmult
Generic Map(MPWID => KEYSIZE)
Port Map(mpand => tempin,
      mplier => sqrin,
      modulus => modreg,
      product => tempout,
      clk => clk,
      ds => multgo,
      reset => reset,
      ready => multrdy);

-- Modular multiplier to take care of squaring operations
modsqr: modmult
Generic Map(MPWID => KEYSIZE)
Port Map(mpand => root,
      mplier => root,
      modulus => modreg,
      product => square,
      clk => clk,
      ds => multgo,
      reset => reset,
      ready =>sqrrdy);
```

Motivating Example

a,b,c;

begin

sensitivity list

Observation

Such weaknesses were found by **SynFuzz** in different open-source designs

- **CVE-2025-51677** : OR1200 Process
- **CVE-2025-51679** : OR1200 Process
- **CVE-2025-51675** : OR1200 Process
- **CVE-2025-51678** : PicoRV32 Proces

Tangible Outputs

- **Study Translation bugs arises during synthesis from RTL to gate-level**
- **Categorize the weaknesses from a broader perspective and create a CWE entry**

SynFuzz Paper : https://arxiv.org/pdf/2504.18812

Contributors

Dr. Sai Manoj P D

Raghul Saravanan

Nitin Patnala

GEORGE MASON UNIVERSITY

Dr. Swarup Bhunia

Sudipta Paria

Aritra Dasgupta

UF UNIVERSITY of FLORIDA

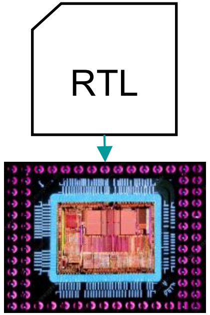
Joerg Bormann

Joerg Bormann

- Responsible for formal security verification tools @ Siemens EDA

Problem:

- RTL makes HW design look like SW design.
- However, it still is HW design, and HW specifics need to be observed.
- Otherwise: Security risks.



Examples:

- Chip behavior not verified: **RTL constructs with Sim / Syn Mismatches, Combinatorial loops, Combinatorial logic with too high propagation delay, Inappropriate Clock Domain Crossings**
- SW-typical weaknesses with different security impact**: Unused logic**
- Different cost structure mandates different mitigations: **States without Reset**

Does this list of examples spark more ideas?

Open Discussion – Please use raise hand feature

CWE is sponsored by U.S. Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA).

Call for Working Members

- **You can either express you interest verbally here or at a later time email**

CWE@MITRE.ORG

Next Meeting (December 12)

CWE@MITRE.ORG

- **Mailing List:** *hw-cwe-special-interest-group-sig-list@mitre.org*
  - ***NOTE: All mailing list items are archived publicly at:***
    - *https://www.mail-archive.com/hw-cwe-special-interest-group-sig-list@mitre.org/*
- **What would members of this body like to see for the next HW SIG agenda?**
- **Questions, Requests to present? Please let us know.**