HW CWE SIG Meeting

Friday, September 13, 2024

Members in Attendance

Gananand G Kini	Monroe, Bruce	Hallman, John
Bob Heinemann (Chair)	Mell, Peter M.	Charles Timko
Alec J Summers	Constable, Scott D	Evan Bryers
Robert Van Spyke	Bojanova, Irena V.	Jason Oberg
Gage Hackford	Frost, Sandy	Mohan Lal
Nicole Fern	Ahmed, Faheem	Swami, Shivam
James Pangburn	Ford, Thomas	Das, Amitabh
Mike Borza	Rich Piazza	Iyer, Priya B
Forbes, Justin	Kepa, Krzysztof	Salehi, Soheil
Sathyamurthi Sadhasivan	Masike, Takunda	Krell, Allen
Manna, Parbati K, (Co-Chair)	Milind Kulkarni	Wortman, Paul
Keerthi Devraj	Joe Reisinger	Frye, Gordon
Mohan Lal	Harner Alexander	Raghudeep
Daniel DiMase		

Agenda

- **SystemVerilog vs Verilog Discussion**
- **HACK@DAC Story**
- **Covert Channel recommendations**

Housekeeping

- Next meeting: October 11th, 12:30 – 1:30 PM EST (16:30 – 17:30 UTC), MS Teams.
- Contact: cwe@mitre.org
- Mailing list: hw-cwe-special-interest-group-sig-list@mitre.org
- Minutes from previous meetings: https://github.com/CWE-CAPEC/hw-cwe-sig

Announcements

- CWE 4.16 release is planned for October
- CWE 5.0 is planned for early 2025

Call for topics
No topics submitted

SystemVerilog
A recap on the discussion regarding the use of SystemVerilog for demonstrative examples was given. The SIG Chair had confirmed with the group that it was decided to add SystemVerilog to language enumeration. No objections were raised and changes will be implemented in an upcoming release.

HACK@DAC – Arun

Overview:

Black Hat Asia and Black Hat USA 2024, we presented about HACK@DAC and the different learnings and what were the initial motivations for getting started with the competition.

We have researched lots of products spanning CPUs, servers, clients, and networking technologies and have found over 500 vulnerabilities, in addition to authenticity research, our specialty has been in root causing and categorization. Our academic partners are well known professors, and they do everything related to hardware and software security. They are very well-received in the academic world with 45,000 citations.

We will talk about the value of organizing CVEs , how we organize HACK@DAC, how it is unique and discuss key takeaways

There have been many attacks published against all of these. We have seen more attacks leveraging vulnerabilities and weaknesses and going lower in the stack. We discussed “hard fails” in one of our early papers. These bugs can be exploited.

Major challenges presented:

1. Limited awareness of hardware security weaknesses.
2. Limited tooling availability
3. Cost of fixing bugs

These three challenges are our primary motivators. All of these items together facilitated the creation of HACK@DAC.

Value:

HACK@DAC is what is called an “open box” approach. Participants are given the SOC and the source code. That offers a much finer grain scope to look into how the security features are implemented. In a “closed box” scenario, we have a designer-centric approach.

How HACK@DAC is Organized

There are two phases in the competition:

Phase 1: Over a two-month period, an open-source design is chosen, and security features are then added to it. We then look up CVEs and user experience to understand how they look, and we develop a list of bugs. The bug is inserted into the open-source chip and the spec is updated. When the buggy design is ready, we announce the competition and participants register. They then begin evaluating the design and submitting bugs. We evaluate the bugs and update the scoreboard periodically.

Phase 2: Top 10 teams are invited to join the competition live over the course of two days.

Winners are announced and some are given the opportunity to speak. There have been over 300 teams from all over the world who have participated. A lot of the participants are now working in top companies.

We wanted to raise awareness, get more tools, and provide a shift in mindset. HACK@DAC is adding a lot of vulnerability and mitigation examples to the CWE. Many teams have also set out to develop their own tools and published some of their work.

Covert Channel Recommendations

These recommendations are based upon comments from prior meetings. We are looking for buy in and help from the community. Any proposed changes are contingent upon review from the CWE tech lead.

Incidental Channels

Incidental channels are unintended communication channels formed by valid properties such as execution time, power consumption, and use of shared resources. When data flows through these resources, data values and metadata may now be inferable by malicious actors. Whenever there are shared resources, incidental channels will occur. Incidental channels are a subset of emerging resources. These concepts merge well.

Recommendation: Consider creating a new CWE based on incidental channels and have it organized under CWE-1229.

Covert Channels

Threat model of covert channels requires attackers to be able to access relevant secret information before exposing it to a covert channel. It has two requirements:

1. Communication channel that an unauthorized actor can manipulate.
2. Actor needs to have access to information that is unauthorized for them to have access to.

An incidental channel does not lead to an introduction of a vulnerability unless there is another weakness present that allows unauthorized access to data. This is an important relationship.

Recommendation: Recommend we explore how to represent the current Covert Channel Weakness (CWE-514) as a composite of Incidental Channel and Improper Access Control.

There currently is no hardware view in covert channels. Paul Wortman suggested in his paper that Covert Channels should be covered in the Security Flow Issue, General Circuit and Logic Design Concerns, and Debug and Test Problems categories. After looking through the descriptions of the General Circuit and Logic Design Concerns and Debug and Test Problems categories, it did not appear those are a good fit for covert channels. After reviewing the remaining HW categories it appears that the Privilege Separation and Access Control Issues category is also a good fit.

Recommendation: we put new incidental channel CWE and Covert Channel CWE in the Security Flow Issue and Privilege Separation and Access Control Issues Categories in HW View.

There was discussion about the designer's intent and if that topic should be considered when describing covert channels. Bob H had commented that the definition of a weakness does not take into account the designer's intent; it is focused on the product's behavior. However, if the community feels that this is an important point to capture we should look for some other element other than the description or the extended description to capture it.

With regards to the comment about CWE-514 being too software centric. The description and extended description of that entry appear to be generic enough to cover both hard and software. It was recommended that the entry could be updated to include a hardware focused demonstrative example.

The above recommendations were presented to the group and asked to think about if these are the changes they would like to make for covert channels. It was communicated to the group that for MITRE to take on these changes that we would need buy-in from the group and need volunteers to help develop out some of the material.