# Hardware CWE™ Special Interest Group (SIG)

Gananand Kini, Bob Heinemann, Luke Malinowski, Gage Hackford, Chris Lathrop, Steve Christey Coley, Alec Summers

MITRE

March 10, 2023

# Agenda

REMINDER: This meeting is being recorded.

- **Housekeeping and Announcements**
- **Working Items for this meeting:**

| | | | |
|---|---|---|---|
| 1 | Crypto HW Weaknesses Community Discussion | Luke M | 30 min |
| 2 | Scope Exclusions and Updates on Submissions Process | Steve C | 20 min |
| 3 | Feedback on our current use of GitHub (if time permits) | Bob H | 05 min |

# Housekeeping

- **Schedule:**
  - **Next Meeting:**
    - **Scheduled for April 14**
    - **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 4.11 targeted for release sometime in May 2023**
  - Might get moved to late April
- **CAPEC announcement (Rich P.)**

# CAPEC Next Steps

- **Given the lack of demonstrable, widespread adoption and value delivery, CAPEC will be a low priority going forward**
- **CWE/CAPEC Board is considering various options (not mutually exclusive)**
  - Continue to host CAPEC site on a server (definite)
    - potential banner of halted maintenance and development
  - Open-source ongoing maintenance and development
  - Transition program to a willing organization for ongoing maintenance and development

- **CAPEC stakeholder survey open through 3/10 (see any CAPEC page)**

# Hardware Cryptographic Weaknesses

## Luke M.

# CWE Applications to Cryptography and Hardware

- **The CWE team is investigating gaps with respect to cryptography, and we believe some potential gaps relate to hardware.**
- **Observations**
  - Secure key management concerns – are critical keys stored in an HSM?
  - Currently little mention of cryptographic hardware (HSM, Crypto IP, TPM?)
  - No discussion of hardware in relation to PKI (Web of trust) systems
  - "key management" in hardware is lacking (OpenTitan KEYMGR)
- **Want to ensure the advantages provided by hardware for cryptographic applications like TLS are properly captured in CWE**
- **Fundamentally think there are gaps in hardware and cryptography and we are looking to have a discussion as to whether that is the case and/or if it is adequately covered**

# How would a motivating example be categorized?

- **Not storing critical keys in an HSM (Hardware security module)**
  - HSM’s only provide Oracle access to the key (sign, verify, encrypt, decrypt)
    - Usage of an HSM is not just mitigating the weaknesses: CWE-316: “Cleartext Storage of Sensitive Information in Memory” or CWE-319: transmitting in the clear
    - Don’t want to say all crypto weaknesses are access control mistakes
  - CWE-668: Exposure of Resource to Wrong Sphere?
    - CWE-522 -Insufficiently Protected Credentials – ?
  - CWE-200 - Exposure of Sensitive Information to an Unauthorized Actor
    - CWE-202->Exposure of Sensitive Information Through Data Queries
- **Separate from the category/tagging: “CWEs relevant to an HSM”**

# PKI & HW CWEs (mind the gap)

- “A public key infrastructure (PKI) is a set of roles, policies, **hardware**, software and procedures needed to create, manage, distribute, use, store and revoke digital certificates and manage public-key encryption.” - Wikipedia
- OpenTitan KEYMGR Discusses attestation, subordinate key creation, and secure management of intermediate state
- CWE-1205: Security Primitives and Cryptography Issues
  - CWE-203 Observable Discrepancy->CWE-1300 Physical Side Channels
  - CWE-325 Missing Cryptographic Step
  - CWE-1240 Use of a Cryptographic Primitive with a Risky Implementation
  - CWE-1241 Use of Predictable Algorithm in Random Number Generator
  - CWE-1279 Cryptographic Operations are run Before Supporting Units Ready
  - CWE-1351 Improper Handling of Hardware Behavior in Cold Environments

# Potentially Gap Covering Cryptographic CWEs

- CWE-295:BASE: Improper Certificate Validation
- CWE-297:VARIANT: Improper Validation of Certificate with Host Mismatch
- CWE-299:BASE: Improper Check for Certificate Revocation
- CWE-319:BASE: Cleartext Transmission of Sensitive Information
- CWE-320:CATEGORY: Key Management Errors
- CWE-321:VARIANT: Use of Hard-coded Cryptographic Key
- CWE-322:BASE: Key Exchange without Entity Authentication
- CWE-323:VARIANT: Reusing a Nonce, Key Pair in Encryption
- CWE-324:BASE: Use of a Key Past its Expiration Date
- CWE-347:BASE: Improper Verification of Cryptographic Signature
- CWE-1279:BASE: Cryptographic Operations are run Before Supporting Units are Ready
- CWE-1291:BASE: Public Key Re-Use for Signing both Debug and Production Code

# CWE Scope Exclusions

## Steve C.

# Why Have “Scope Exclusions” Instead of Just Defining CWE’s “Scope”? (as of March 2023)

- **CWE’s definition of “Weakness” (updated in 2022): *"A condition in a software, firmware, hardware, or service component that, under certain circumstances, could contribute to the introduction of vulnerabilities.”***

- **Frequently, people suggest adding entries to CWE that do not satisfy this “weakness” definition, but they want CWE to treat them as “weaknesses”**
  - CWE’s focus is on “behavior” of products
- **Other times, people effectively suggest the expansion of CWE’s scope beyond “traditional” software/hardware**
- **“Scope exclusions” attempt to formalize decisions about what can or cannot be included in CWE as an official “weakness” entry**

# Scope Exclusions – Overview – as of March 9, 2023

- **Some discussion in various groups, especially ICS/OT SIG and HW SIG, beginning in Spring 2022**
- **Integrated into external submission review process in Summer 2022**
  - Related work: other “quality” problems in external submissions
- **Continued development by CWE Team in late 2022**
- **Plans to publish on CWE web site ASAP (days/weeks)**
- **Phases:**
  - Development
  - Community Proposal ← ***we are (almost) here***
  - Community-Wide Review
  - Decision: Accept / Reject

# Scope Exclusions – Summary (as of March 9, 2023)

| ID | Name |
|---|---|
| SCOPE.NOTREAL | Not a real-world issue |
| SCOPE.HUMANPROC | Human/organizational process |
| SCOPE.MOTIVE | Motivation instead of the mistake |
| SCOPE.SITUATIONS | Focus on situations in which weaknesses may appear |
| SCOPE.GROUPING | Grouping of issues without a common behavior |
| SCOPE.NOMITS | No actionable mitigations |
| SCOPE.CUSTREL | Not customer-relevant |
| SCOPE.CONFLICT | Conflict/contradiction with other weaknesses |
| SCOPE.NOSEC | Not security-related |
| SCOPE.ADMINERR | Admin/user error |

Focus for HW SIG Meeting is limited. See "Other Scope Exclusions" section of this presentation for the other scope exclusions.

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

# What’s in a “CWE Scope Exclusion”

- ID
- Name
- Description
- Rationale
- Examples
- Resolution
- Debate
- Status

- … (future) various tracking metadata

# SCOPE.NOMITS - No actionable mitigations

| Info | Details |
|---|---|
| **Description** | There are no actionable mitigations available to the developer/designer/manufacturer to prevent or reduce the weakness. |
| **Rationale** | If a product has a weakness type with no known fixes or mitigations, then a CWE entry would not be helpful to the developer / designer / manufacturer, i.e., it is not actionable. |
| **Examples** | Some techniques for analyzing hardware do not have any practical real-world mitigations for highly-resourced adversaries, yet they remain a concern to defenders. |
| **Resolution** | Submissions will be reviewed on a case-by-case basis, delayed, and possibly cited as examples until this exclusion is finalized after extensive community feedback. |
| **Debate** | As of January 2023, there is some community disagreement about this scope exclusion: (1) Developers could avoid the affected functionality altogether. (2) Weaknesses without mitigations could serve as topics for academic study. |
| **Status** | Development |

# SCOPE.CUSTREL - Not customer-relevant

| Info | Details |
|---|---|
| **Description** | The issue is not relevant to the threat model or security concerns of the product's owner/operator (i.e., the customer). |
| **Rationale** | Traditionally, the focus on publicly-disclosed vulnerabilities has been on products that affect customers. Widening the scope to include non-customer interests would risk creating weaknesses with no relevance to customers who want to acquire secure products. |
| **Examples** | CWE-1278 is about avoiding reverse engineering using certain techniques, which does not affect the customer directly. A separate submission says: “"Protections and measures intended to prevent or hinder reverse engineering of Intellectual Property (IP) are not present in the product design.“ CWE-1297 is about potential for leaking of NDA information between non-customer organizations during the manufacturing process. |
| **Resolution** | Submissions will be reviewed on a case-by-case basis, delayed, and possibly cited as examples until this exclusion is finalized after extensive community feedback. |
| **Debate** | As of January 2023, there is some community disagreement about this scope exclusion. **Customers are not the only participants within the ecosystem**, and concerns such as Intellectual Property violations have clear financial implications for vendors if compromised. If not handled carefully, allowing CWEs in this area could cause conflicts; e.g., if a CWE is created that a product doesn't obscure its code enough, that CWE would apply to all open source products.<br>Note that there may be a logical inconsistency between SCOPE.CUSTREL and SCOPE.CONFLICT, since there may be differences in threat models across different industries, and each participant might have different priorities. For example, in healthcare, availability is a priority that may conflict with password requirements, since locking out a medical device user may cause security issues. [Debate edited for “brevity”] |
| **Status** | Development |

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

# SCOPE.CONFLICT - Conflict/contradiction with other weaknesses

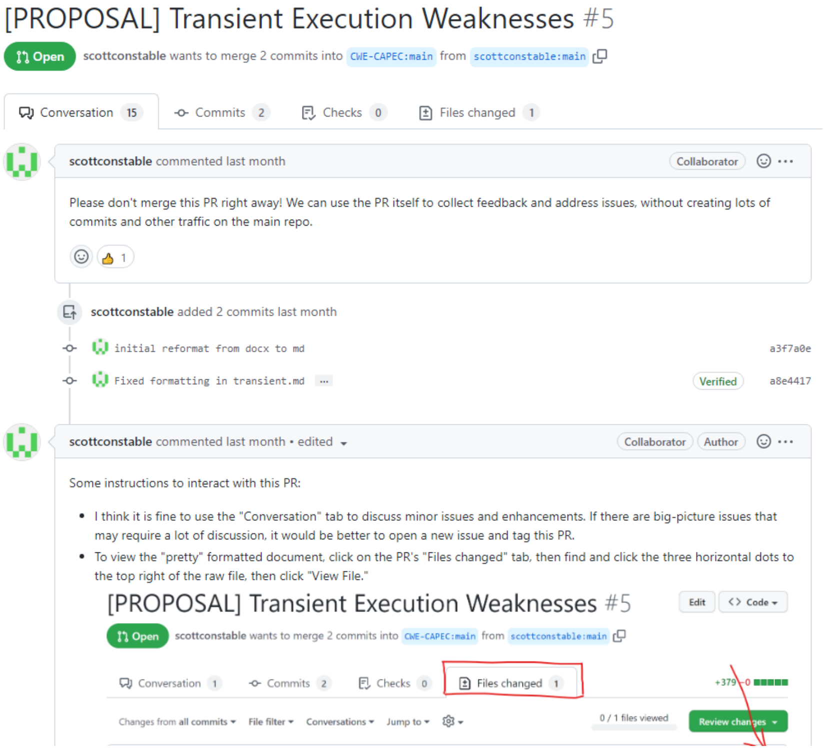
| Info | Details |
|---|---|
| **Description** | The issue directly conflicts with or contradicts other CWE entries, e.g., "X is bad" in one CWE, and "Y is bad so do X instead" in another CWE. |
| **Rationale** | Directly-conflicting CWE entries can cause user confusion and/or suggest some lack of agreement as to what constitutes "secure“ products. |
| **Examples** | "The product has code that can be easily reverse-engineered, allowing adversaries to steal Intellectual Property." This statement implies that all open source code has a weakness because it is not obfuscated. |
| **Resolution** | The submission might be rejected unless it reveals some other problems with existing CWE content, which might prevent it from progressing to later stages until the existing CWE issues can be addressed satisfactorily. |
| **Debate** | As of January 2023, one area of active debate involves the desire of some community members for CWE to **include weaknesses for code/logic that can be easily reverse-engineered** (note that this is also affected by SCOPE.CUSTREL). If CWE covers a concern that effectively promotes code obfuscation (i.e., says that it's too easy to extract the real code/logic), this **could be seen to directly conflict with CWE-656: Reliance on Security Through Obscurity**, which effectively follows Kerckhoff's principle, summarized as: "a cryptosystem should be secure, even if everything about the system, except the key, is public knowledge." As another example, in 2021, members of the CWE-Research list discussed older CWE entries related to password aging that directly implied that password aging was an important capability, but the modern belief is that password aging should be unnecessary. Yet, **creation of a new entry such as "Reliance on Password Aging" would conflict with the original CWE entries and any products that still rely on passwords**. Another potential conflict arises when design decisions require tradeoffs between security and other desired features such as safety and reliability…. [edited for “brevity”] |
| **Status** | Development |

# Active GitHub Items

## Bob H.

# GitHub for Document Collaboration and Discussion

- Scott has proposed a way:
  - Submit a Pull Request.
  - Make comments in conversation tab.
  - Can link to specific lines in Files Changed tab.



[PROPOSAL] Transient Execution Weaknesses #5

Open scottconstable wants to merge 2 commits into `CWE-CAPEC:main` from `scottconstable:main`

**scottconstable** commented last month

Please don't merge this PR right away! We can use the PR itself to collect feedback and address issues, without creating lots of commits and other traffic on the main repo.

**scottconstable** added 2 commits last month

```
initial reformat from docx to md          a3f7a0e
Fixed formatting in transient.md   Verified   a8e4417
```

**scottconstable** commented last month • edited

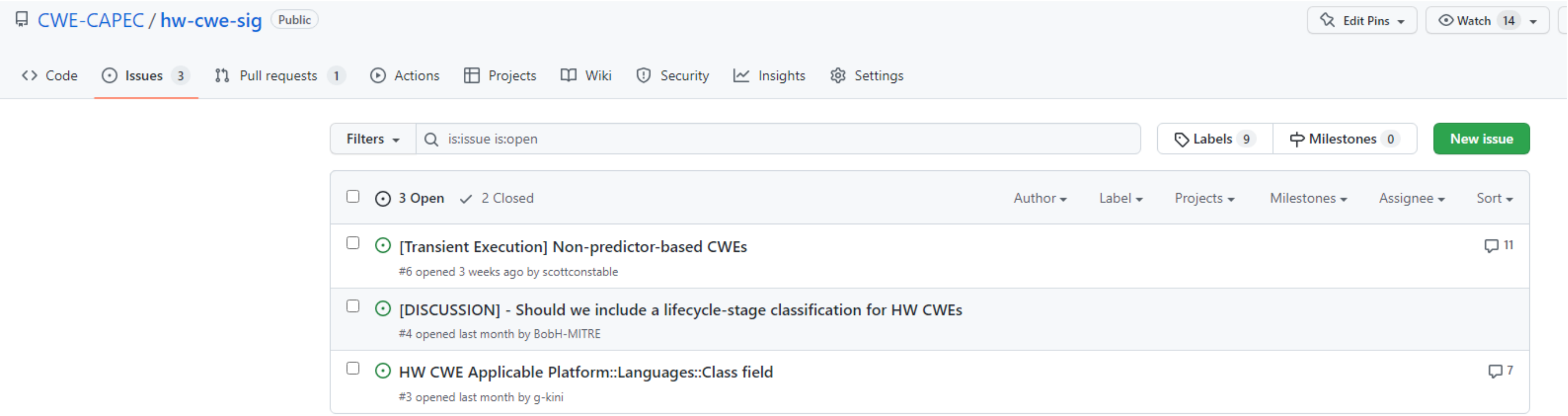
Some instructions to interact with this PR:

- I think it is fine to use the "Conversation" tab to discuss minor issues and enhancements. If there are big-picture issues that may require a lot of discussion, it would be better to open a new issue and tag this PR.
- To view the "pretty" formatted document, click on the PR's "Files changed" tab, then find and click the three horizontal dots to the top right of the raw file, then click "View File."

[PROPOSAL] Transient Execution Weaknesses #5

Open scottconstable wants to merge 2 commits into `CWE-CAPEC:main` from `scottconstable:main`

# GitHub for Community Items – Feedback


CWE-CAPEC / hw-cwe-sig Public

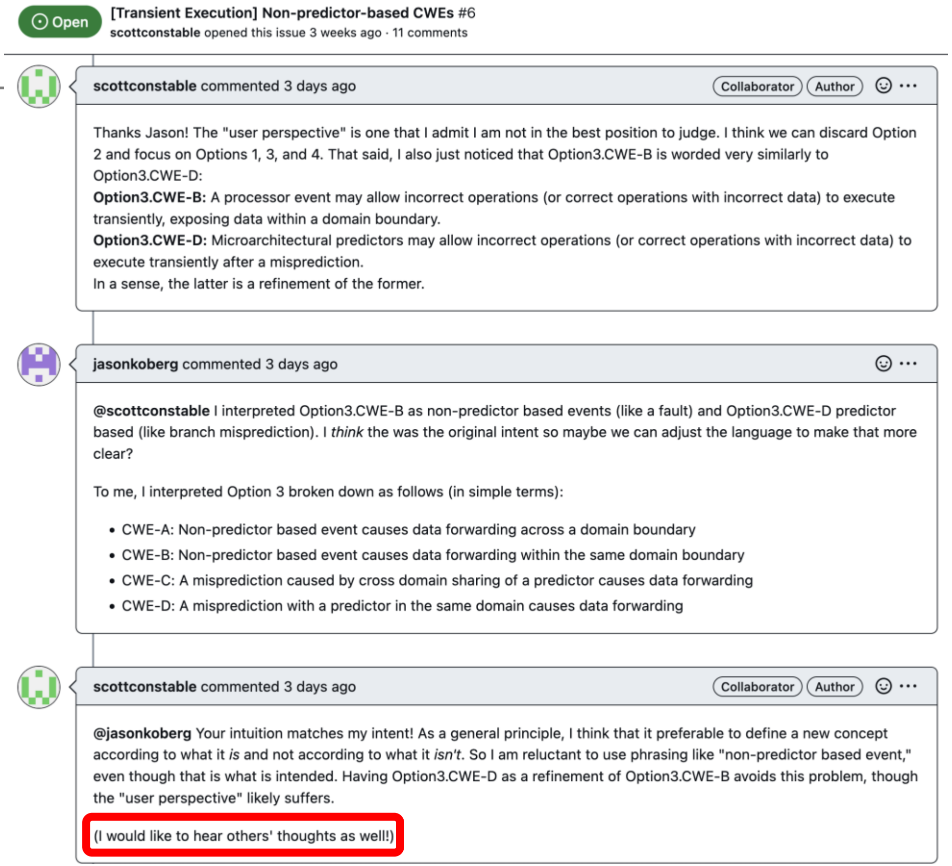
○ 3 Open ✓ 2 Closed

- [Transient Execution] Non-predictor-based CWEs — #6 opened 3 weeks ago by scottconstable — 11
- [DISCUSSION] - Should we include a lifecycle-stage classification for HW CWEs — #4 opened last month by BobH-MITRE
- HW CWE Applicable Platform::Languages::Class field — #3 opened last month by g-kini — 7

ProTip! Find everything you created by searching author:BobH-MITRE.

- **Are you able to access and work with this easily?**

# Some awaiting feedback



**[Transient Execution] Non-predictor-based CWEs #6**

**scottconstable** opened this issue 3 weeks ago · 11 comments

**scottconstable** commented 3 days ago

Thanks Jason! The "user perspective" is one that I admit I am not in the best position to judge. I think we can discard Option 2 and focus on Options 1, 3, and 4. That said, I also just noticed that Option3.CWE-B is worded very similarly to Option3.CWE-D:

**Option3.CWE-B:** A processor event may allow incorrect operations (or correct operations with incorrect data) to execute transiently, exposing data within a domain boundary.

**Option3.CWE-D:** Microarchitectural predictors may allow incorrect operations (or correct operations with incorrect data) to execute transiently after a misprediction.

In a sense, the latter is a refinement of the former.

**jasonkoberg** commented 3 days ago

**@scottconstable** I interpreted Option3.CWE-B as non-predictor based events (like a fault) and Option3.CWE-D predictor based (like branch misprediction). I *think* the was the original intent so maybe we can adjust the language to make that more clear?

To me, I interpreted Option 3 broken down as follows (in simple terms):

- CWE-A: Non-predictor based event causes data forwarding across a domain boundary
- CWE-B: Non-predictor based event causes data forwarding within the same domain boundary
- CWE-C: A misprediction caused by cross domain sharing of a predictor causes data forwarding
- CWE-D: A misprediction with a predictor in the same domain causes data forwarding

**scottconstable** commented 3 days ago

**@jasonkoberg** Your intuition matches my intent! As a general principle, I think that it preferable to define a new concept according to what it *is* and not according to what it *isn't*. So I am reluctant to use phrasing like "non-predictor based event," even though that is what is intended. Having Option3.CWE-D as a refinement of Option3.CWE-B avoids this problem, though the "user perspective" likely suffers.

(I would like to hear others' thoughts as well!)

# Next Meeting (April 14)

**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.**

# Backup

# Mailing List Items

**Mailing List:** *[hw-cwe-special-interest-group-sig-list@mitre.org](mailto: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/](https://www.mail-archive.com/hw-cwe-special-interest-group-sig-list@mitre.org/)*

# Active Discussions – Category Name Update

- A sig member has suggested that the category CWE-1206 name should be changed from "Power, Clock, Thermal, Reset Concerns" to
- "Power, Clock, Thermal, Reset Concerns, or Semiconductor Defects."
- The rationale is this: The current title does not discuss hardware deterioration.
- Manufacturing and Life Cycle Management Concerns (1195) covers "defects that arise in the semiconductor-manufacturing…”
- **Recommendation is not to make this change.**

https://github.com/CWE-CAPEC/hw-cwe-sig/issues/1

# Active Discussion: CWE-1248 - Issue with phrase "Security Guarantees"

- **The SIG Community is having issue with the phrase "security guarantees" in the extended description.**

- **Some have issue with semantics of products are incapable of making claims and others have issue with the language as their may be specific legal consequences of the term.**

- **The sentence under scrutiny is**
  - “If such faults occur in security-sensitive hardware modules, ***security guarantees*** offered by the device will be compromised.”

- **There are two proposals at the moment:**

- **Hareesh proposes, "If such faults occur in security-sensitive hardware modules, security guarantees offered by the device vendors/ manufacturers will be compromised."**

- **Jim Wesselkamper proposed: "If such faults occur in security-sensitive hardware modules, the security objectives of the hardware module may be compromised.”**
  - Jason O and Joe J endorse this proposal.

https://github.com/CWE-CAPEC/hw-cwe-sig/issues/2

# SCOPE.NOTREAL - Not a real-world issue

| Info | Details |
|---|---|
| **Description** | The issue has not happened and cannot occur in real-world software or other electronic logic, and/or has no actual security implications. |
| **Rationale** | - |
| **Examples** | Due to a misinterpretation of source material, CWE-365 originally described an insecure behavior that did not exist in any real-world compilers, so it was deprecated. |
| **Resolution** | As of January 2023, the submission will be rejected or delayed pending close review, possibly including community engagement. |
| **Debate** | No active debate. |
| **Status** | Development |

# SCOPE.HUMANPROC - Human/organizational process

| Info | Details |
|---|---|
| **Description** | The submission focuses on a problem in a human or organizational process or policy (e.g., insufficient developer training) that is not measurable and does not produce concrete artifacts that identify weaknesses. Note: this scope exclusion does NOT apply to cases in which humans directly influence insecure behavior of a product, such as insecure configuration (which can also be automated or performed by code, not humans). |
| **Rationale** | Weaknesses can emerge as a result of these activities. Other efforts cover this area, e.g., BSIMM, OWASP SAMM, and NIST Secure Software Development Framework. Note that this exclusion is similar to SCOPE.ADMINERR in that it reflects human actions. |
| **Examples** | Lack of developer training, insufficient testing, not following secure coding standards, corporate policy gaps, or human resource / people-management problems such as not hiring enough people. CWE-1297 is about the potential for leakage of NDA information between non-customer organizations during the manufacturing process. |
| **Resolution** | As of January 2023, such submissions will be rejected (if framed as "weaknesses") but could help modify existing CWEs for elements such as modes of introduction. |
| **Debate** | No active debate, but manufacturing processes (of concern to hardware, ICS/OT, supply chain, etc.) may fall under this exclusion, as would ICS/OT. |
| **Status** | Development |

# SCOPE.MOTIVE - Motivation instead of the mistake

| Info | Details |
|---|---|
| **Description** | Any characterization of motivation (e.g., "malicious") that does not focus on the actual weakness, whether intentionally or accidentally introduced. |
| **Rationale** | A weakness is based on the behavior of the product, and malicious actors can introduce the same weaknesses as non-malicious developers. |
| **Examples** | - |
| **Resolution** | As of January 2023, submissions are likely to be rejected unless they can be converted to actual weaknesses. Any provided references or examples could be used to modify demonstrative or observed examples of existing CWEs. |
| **Debate** | No active disagreement as of January 2023. |
| **Status** | Development |

# SCOPE.SITUATIONS - Focus on situations in which weaknesses may appear

| Info | Details |
|---|---|
| **Description** | The submission focuses on conditions or situations in which weaknesses are more likely to appear but are outside of the direct control of the product. |
| **Rationale** | CWE is focused on the flaws/defects that can occur within a product. While various situations can contribute to the introduction of weaknesses, these are better captured as modes of introduction, which are recorded in a separate field in CWE entries. |
| **Examples** | "The growing connectivity between IT and OT systems can introduce new vulnerabilities.“ (SEI ETF) |
| **Resolution** | Submissions are likely to be rejected unless they can be recast as categories or can influence modes of introduction of existing CWEs, i.e., modify entries. |
| **Debate** | No active disagreement as of January 2023. |
| **Status** | Development |

# SCOPE.GROUPING - Grouping of issues without a common behavior

| Info | Details |
|---|---|
| **Description** | The submission is a grouping of issues or concerns related to the same feature such as technology, language, development lifecycle, etc., but there is not a common behavior between them all. |
| **Rationale** | Weaknesses are based on specific behavior and other shared criteria that can be hierarchically classified under a common ancestor in the research view 1000. If no such ancestor is possible, then the submission is not likely a weakness. |
| **Examples** | "If the developer doesn't handle files well, attackers can steal or delete data.“<br>With respect to hardware, weaknesses that have physical characteristics do not share the same hierarchy in research view 1000. However, the hardware community asked for them to be grouped into a category (CWE-1388). |
| **Resolution** | The submission might be acceptable as a category, or some of its content could modify existing entries. In some cases, the submission might influence subtree organization. |
| **Debate** | No active disagreement as of January 2023. |
| **Status** | Development |

# SCOPE.NOSEC - Not security-related

| Info | Details |
|---|---|
| **Description** | The issue is solely concerned with safety, reliability, or another property not related to security. Clarification on privacy: if an issue is related to desires for access control or preservation of confidentiality, it is within scope of "security". |
| **Rationale** | Many aspects of industrial safety, such as correct electric shielding and insulation, do not affect security. There are many safety-focused standards throughout different industries, and expanding CWE's scope to include safety-only issues would limits utility to most current CWE users. |
| **Examples** | - |
| **Resolution** | Submissions will be reviewed on a case-by-case basis, delayed, and possibly cited as examples until this exclusion is finalized after extensive community feedback. |
| **Debate** | As of January 2023, it is suspected that there will be some community disagreement about this scope exclusion. |
| **Status** | Development |

# SCOPE.ADMINERR - Admin/user error

| Info | Details |
|---|---|
| **Description** | The issue focuses on security errors that are made by an admin or user of the product/service, not the developer or maintainer of the product/service. |
| **Rationale** | Traditionally, CWE has covered defects or flaws within the product itself. This is separate from how administrators or users may misuse or abuse the products in ways that were not intended, typically through insecure configuration, so there is little that product developers can do. It seems likely that such issues will not be relevant to most CWE users.<br>Note that this exclusion is similar to SCOPE.HUMANPROC in that it reflects human actions. |
| **Examples** | - |
| **Resolution** | Submissions will be reviewed on a case-by-case basis, delayed, and possibly cited as examples until this exclusion is finalized after extensive community feedback. |
| **Debate** | As of January 2023, this is a new exclusion. It is suspected that there will be some community disagreement about this scope exclusion, since **many services deploy and manage their code for customers in ways that obscure the distinction between system administrators and software developers** ("DevOps"). CWE is also intended to cover misconfigurations made by admins, such as running a process at an excessively high privilege level. |
| **Status** | Development |