User Experience Working Group

May 28, 2025

Participants

- Steve Christey
- Chris Coffin
- Matthew Coles
- Matt DiVisconte
- Scott Drinkall
- Jim Duncan
- Farbod Foomany
- John Keane
- Connor Mullaly
- Przemyslaw (Rogue) Roguski
- Paul Wortman

Agenda

- CWE and Proactive Vulnerability Management
- CWE Content Development Repository (CDR) Overview
- Open Discussion

Meeting Summary

- **CWE and Proactive Vulnerability Management**
  Przemek presented a high-level summary of a Proof of Concept (POC) implemented at Red Hat, focusing on improving SAST scanning results and prioritization models based on CVE records.
- **CWE Content Development Repository (CDR) Overview**
  Connor presented the role of the CDR in increasing transparency and community contributions to CWE content, and how community members can provide feedback and comment on submissions.

Action Items

1. **Follow-up on Next Steps for POC:** Share more real results for the POC and next steps in the next meeting. (Przemek)
2. **Next Meeting Preparation:** Inform Chris if more time is needed in the next call to present additional results and ideas. (Przemek)

Meeting Notes

CWE and Proactive Vulnerability Management (Przemyslaw Roguski)

- **Background:** Przemyslaw presented slides that are a high-level summary from POC at Red Hat from the beginning of the year. Many SAST scanning vendors include links to the CWE weaknesses directly from the CWE program. Currently, SAST scanning results contain a lot of false positive findings, and processing all automatically triggered SAST results is time-consuming and inefficient. As a POC, they sought to improve SAST scanning findings by a prioritization model based on CVE records and associated CWE weaknesses. The intention was to identify pattens between SAST scanning results for a product/component and similar weaknesses reported.
- **Finding:** There are some patterns. Based on those patterns, there can be a more precise message delivered to the engineering teams to triage weaknesses.

- **Benefits:** Reduces the engineering time spent on patch production and release process; reduce the number of reported CVEs against the portfolio; improve the customer experience.
- **Process:** Przemyslaw shared a process diagram (in the slides). There are two types of data: weaknesses from CVE records and weaknesses from SAST scanning results based on software development review categories. AI is used to find patterns in the data, resulting in more precise weakness categories.
- **Question:** Matt asked if CVE ratings were based on relative severity in terms of prioritization and not CVSS scores. Przemyslaw responded that analysis is already done by the vulnerability management team. Analysis is based on associated weaknesses, CVSS score, how the Red Hat product is impacted by the vulnerability. The outcome is the Red Hat severity rating, used as a reference for customers.
- **Question:** John relayed a best practice to acknowledge false positives but allow the developer to not report up the chain-of-command overcomes issues. John also mentioned the Fortify tool that identifies critical findings (critical, high, medium, low) with high confidence of prioritization and high correlation with OWASP Top 10.
- **Question:** Steve, noting that the Developer View 699 is incomplete, asked how to standardize CWEs that were not part of 699. Przemyslaw responded that using AI to map from a random CWE record was not successful. Rather, they mapped specific categories from the 1400s View to the Software Development View. If a weakness was reported by the SAST that was not by default in the Software Development View, they looked for the weakness in 1400s, the category, and the map category in the Software Development View.
- **Question:** Steve, noting that CWE-284 for Improper Access Control covers multiple concepts, asked how to address this lack of precision. Przemyslaw responded that any imprecise SAST scan results are reviewed manually, as there is no easy way to associate it automatically.

CWE Content Development Repository (CDR) Overview (Connor Mullaly)

- **Background:** The CDR is a public repository that holds actively developed community submissions for CWE content, such as a modification to an existing CWE entry or a submission for a new CWE entry. Public accessibility allows for more engagement and visibility.
- **Steps to submit content:** Connor gave a walk-through for contributing CWE content. At a high level: 1) web submission and internal processing, 2) collaborative content development, 3) publication on CWE website at the release periods.
- **Required elements:** 1) submitter contact information, 2) submission details, 3) related weaknesses, 4) references.
- Once processed, the submitter receives an email to direct them to their entry on the CDR.
- **GitHub page:** The CDR is hosted on GitHub: https://github.com/cwe-capec/cwe-content-development-repository
    - There are 3 distinct engagements with the CDR: 1) submitter views their submission and engages the CWE team, 2) someone provides their own perspectives directly to the CDR, 3) someone views but does not directly engage.

    - The page outlines guidelines for content submission and external content submission phases.
    - Each submission is treated as its own issue in the CDR. Labels identify phases.
    - Each submission links to a text file that is stored in the repository and has an ID and submission date.
- **Participant response:** Chris asked about the location for participants to comment and make suggestions. Connor responded that comments can be posted for each issue. For comments/suggestions, add comments directly to the issue. These are recorded into the submission file and recorded. For process improvements, there is a feedback template. Issues associated with content submission should not be submitted as a new issue. Steve noted that the review identifies different problem types, which are put into the discussion framework. So far there have been minimal unrelated comments. Matt stated that as people become more familiar with content submissions and aware of their availability, there could be more noise.
- **Conference engagement:** Matt asked if this content has been presented at a conference, such as at VulnCon. Education is needed about CWE, how to submit good content, and how to overcome challenges to submission. Connor responded that doing so would create more visibility. Jim suggested a half-day workshop at VulnCon and presenting later this year at BSidesRDU (or other BSides) or maybe Triangle InfoSeCon. Matt suggested a Defcon AppSec Village presentation.
- **Leveraging CWE WGs:** Matt asked if there was an intent to use CWE working group memberships as candidates for commenting, given that they are already in the CWE community. Connor responded that the Hardware WG and AI WG have relied heavily on the CDR. There have been several submissions from the WGs. The WGs can send them to external parties for review. Steve noted that CDR has been under development for 2 or 3 years but was only launched publicly in April. More time is spent on development than considering outreach methods to build a community-oriented approach. CDR has been used as a tracker for hardware submission and an interactive environment for AI. Chris offered the idea of voting to determine if a CWE was ready.
- **Presentation suggestions:** John suggested giving a presentation at the next Software and Supply Chain Assurance (SSCA) forum. The most recent one had discussion on the use of AI tools to generate software. But no one discussed how to check for vulnerabilities.
    - Contact: Bob Martin
- John emphasized the importance of reporting correct processes to avoid errors. Generative AI provides some useful information to assist developers.
- Connor suggested that any modifications to existing content from the user experience side could be sent to the CDR. There is not an image upload but could be shared in the description if hosted elsewhere.

NEXT MEETING 6/25