CWE REST API Working Group Meeting – February 8, 2024

1) Call to order – recording
2) Agenda approval
3) Minutes approval
4) Regular order
   - Note – email list reflector: cwe-capec-rest-api-working-group-list@mitre.org
   - Note – GitHub: https://github.com/CWE-CAPEC/REST-API-wg
   - Note – CWE data response content: https://cwe.mitre.org/data/xsd/cwe_schema_latest.xsd
   - Note – Specs are here: https://github.com/CWE-CAPEC/REST-API-wg/tree/main/specifications
   - Note – Latest OpenAPI spec: https://github.com/CWE-CAPEC/REST-API-wg/blob/main/specifications/openapi.yaml
   - Add Krzysztof Kepa, AMD to roster
   - Status
     - MITRE and IEEE WG can access API
     - Some have tried, but not rigorous/regular access
     - IEEE 3164 WG alerted to availability of API access
   - Feedback
     - "Improvements" doc sent around from MITRE last week
       - Content shown below
     - 1. "No descendants" (for example) issue
       - Strict 404 is requesting a URI, and nothing was there…
       - 204 getting mixed reviews
         - Some sightings of 204 would just be an error unless special handling code added
         - Some folks won't add that special code
         - 404 has the same caveats
       - Look through references in amendments doc from MITRE
     - 2. Children vs. members hierarchies/structures needs some finesse
       - Issue is related to Views vs. Categories
       - For example, "Top 25" is a list of items, not really a path to the IDs' contents
     - 3. Descendants vs. children discussion
       - Suggesting adding "proper" handling of endpoints having added children
       - Other solutions exist. We need to settle on a solution.
     - 4. Related to 3: list or graph. Again, need to decide.
5) Action item review
   - WG: please access the database via REST API and send feedback
   - Status update
6) Adjourn

Improvements doc from MITRE: (please review and forward thoughts on changes required as these items need to be addressed by MITRE):

1. The API returns an error (404 – Not Found) when requesting a relationship (e.g., **descendants**) that does not exist.

   Suggestions:

REST APIs are often designed to perform CRUD operations on a repository of some sort. Using this interpretation, it would seem like that lack of the existence of a requested URI should return a 404 – which is an error. In an API, this can also mean that the endpoint is valid but the resource itself does not exist.

However, REST API endpoints have long left that interpretation behind, so we can think of these requests as asking the "meta" question: *return a list of the descendants*, and the empty list is a reasonable response.

If we decide to stay with the 404 error response, we need to return a correct error message. Right now, the message returned can be confusing. There is also a way to return a JSON object as the error message (e.g., https://github.com/omniti-labs/jsend, https://www.rfc-editor.org/rfc/rfc7807.html).

Here are some pros-and-cons discussions from the internet:

- https://stackoverflow.com/questions/13366730/proper-rest-response-for-empty-table
- https://apihandyman.io/empty-lists-http-status-code-200-vs-204-vs-404/
- https://medium.com/nerd-for-tech/navigating-http-status-codes-for-rest-apis-39f25fcd8cc6

2. Views and Categories do not have "children", but instead have "members". Currently, the **descendants** and **parents** endpoints return a 404 for non-weaknesses, because the API does not recognize the "has_members" hierarchy.

   Suggestions:

   - Conflate the "has_member" relationship with the "child_of" relationship and use the same endpoints for both.
   - Introduce separate endpoints to traverse the "has_members" hierarchy.

3. The **descendants** endpoint returns all levels below (the full "graph") but the **parents** endpoint returns only one level up. Why the difference?

   Suggestions:

   - First, change the name of the **descendants** endpoint to **children** and return just one level down
   - How to handle returning the full list of CWEs above/below
     - Introduce endpoints **descendants** and **ancestors**, or
     - Introduce a query parameter **all=true** to **children** and **parents**
   - Traverse the hierarchy using multiple requests.

4. Instead of the endpoints returning a simple list of CWEs when multiple levels are involved, return a graph structure.