AWG Meeting Notes

101320 Meeting Notes

* No changes to the Agenda proposed by attendees
* An email was sent to SPWG proposing that the AWG make a recommendation to the SPWG and Board regarding the deployment schedule for IDR
* Chandan’s proposed changes to JSON 5.x were voted on last time. The question is whether the vote counts as acceptance. The attendees interpreted last week’s vote as being an approval vote. No negative responses to the topic occurred in the email list, which was one condition for accepting these changes.
* IDR Deployment criteria
  + No changes to the criteria were initially proposed
  + User stories should be tied to unit tests
  + Suggestion that we include unit test code coverage for future development
    - There may be a feature in Github that permits a test script to be run when code is submitted
  + Results of community testing should be provided when available
  + Going forward, having a user acceptance test suite with tests that refer to user stories and exercise those use cases seems to be the best way to have tests that prove user story fulfillment.
  + Concerns about concurrency were raised, there is an effort to work with an internal testing team at MITRE to execute proper concurrent tests.
  + Also addressed was a concern to be able to audit actions of IDR, for a scenario if a bug enters IDR that allows improper reservations like double reserving an ID. It was discussed to arrange for IDR to do structured JSON logging with unique request IDs of reservation actions which would allow the secretariat to review logs to audit the actions of a production IDR.

100620

Notes from the meeting

* IDR update, we’re in the midst of a two-week testing period. We ask that testing wrap up by Oct 9th.
* At next week’s meeting we’d like to discuss findings from the test period.
* We’ve proposed a three-phase approach to deploying IDR. AWG consensus vote on whether or not we should deploy, the board would also get an opportunity to vote next week. The third component is a MITRE go/no-go decision based on the internal testing.
  + Dave W. suggested that the SPWG should have a role in this process.
  + The AWG will send a note to the AWG and SPWG email lists, then open a consensus vote for the IDR for review at the next AWG meeting, at which point we can send the AWG consensus position to the SPWG and the board.
* Chandan discussion of proposed changes to JSON. Those changes can be found here: <https://github.com/CVEProject/automation-working-group/pull/92/files>
  + Those in attendance supported Chandan’s suggested changes, there was no dissent. The topic will be revisited at next week’s meeting and a decision will be made
* Slides for the October conference will be reviewed at next week’s AWG meeting to ensure that the contents reflect the community’s perspective.
  + Those slides will be sent out late this week or early next week
* Proposal to add the meeting notes to the Github site along with an action item list.
  + 12 voted in support, no one dissented
* Testing for IDR
  + It would be nice if the org can see its users and the details of the org. Details can be found here: <https://github.com/CVEProject/cve-services/issues/130>. We plan to add such features in the future.
  + MITRE has been doing internal testing with the interactions between CPS and IDR, which tests both the CNA functionality and the Secretariat functionality
  + Dave suggested that we provide assurance that IDR has been thoroughly tested

092920

Relevant links

* [Sprint Reviews](https://github.com/CVEProject/cve-services-doc/tree/master/sprint-reports)
* [AWG Services](https://github.com/CVEProject/cve-services-doc)
* [Entry Submission and Upload Service (ESUS) API](https://github.com/CVEProject/cve-services-doc/blob/master/api-endpoints.md)
* [Demo of the ESUS service](https://www.youtube.com/watch?v=N_P3AwbCLH4&feature=youtu.be) (still in progress)
* [JSON Schema](https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss)
* [IDR OpenAPI YML file](https://github.com/CVEProject/cve-services/pull/48)
* [IDR Docker container](https://github.com/CVEProject/cve-services/commit/a9d91fdecf4012f32253e029b70520ad0397ec2d)
* [Security Policy](https://github.com/CVEProject/cve-services/security)
* [Backlog board](https://github.com/CVEProject/cve-services/milestone/13)
* Phase 1 user stories
* [Acceptance of IDR phase 1 User Stories](https://forms.office.com/Pages/ResponsePage.aspx?id=SNwgxlAdUkmLOd9NVNdNggBrats_1HNIrqs-c2Q7Ob1UQlc5RVdNTUtONFdUOVo3MFRJWVpVRlhRSC4u)

Meeting Notes

**Votes/Decisions Made at this meeting:**

|  |  |
| --- | --- |
| **Topic/Question** | **Decision** |
| To adopt IDR User Stories Phase 1 as recorded in GITHUB. Adoption means this will be the agreed upon functionality of IDR Phase 1 and the basis upon which to begin addressing backlog items and new functionality. | Vote: 7-0 (IDR User Stories adopted) |
| To adopt JSON 5.x Changes noted on 9/22  when SPWG approves Tagging proposal | Vote: 5-0 (JSON changed (in principle) adopted. AWG will review implementation after Joe W. makes changes. |

* Comments on the agenda: none
* Action items
  + Work items for the SPWG, in progress
  + Chandan submitted a pull request for …
  + Votes for acceptance of User Stories
    - 2 for yes
    - 1 for no
* Thu: IDR phase 1 user stories
* IDR testing
  + On Thur a note was sent to the AWG mailing list announcing the start of testing on the 26th
  + We have 14 community testers so far
    - Testing the interface to ensure it meets Phase 1 MVP
    - Testing should start today
    - We had a few corrections to the secretariat username and we found a few errata in the files sent previously
    - Code can be downloaded and tested, or the AWS version can be tested with the Postman scripts
    - Has anyone started testing?
      * One person started testing the down-loaded code
      * Chandan requested testing credentials, we need to provide those
  + Results of user story survey
    - 7 yes’
    - 0 no’s
  + Proposed JSON 5.x changes
    - No changes were suggested on the list
    - Suggest we accept those changes as they are
    - When the steering committee approves their changes, we can fold those in
    - Chandan proposed some additional changes in a pull request [here](https://github.com/CVEProject/automation-working-group/pull/92/files). Those will be treated separately from the set of changes under consideration now
  + AWG has been asked to provide a 60 minutes presentation at the Oct. summit
    - No concerns were expressed about the AWG doing so
    - A demo has been requested during the presentation
    - We could discuss the rollout strategy
      * What is IDR?
      * Here’s where we are with it
      * Here’s how it can be used
      * Here’s what the future holds
        + An expansion of the user registry
        + Cognito
        + User Registry
        + Phase 1 Entry Submission and Upload Service, Phase 1
        + Phase 1 website
      * We should provide information how to report bugs for IDR
    - Perhaps we could discuss the business purpose of IDR
  + Suggestion that we add an overview of the AWG services to the Github board
  + Suggestion that we add a link to the website for each of the IDR clients
  + Suggestion that we add a workflow and examples of how to use the IDR service to the wiki on Github
  + We will set up a half-hour call to discuss IDR processes
  + Currently, the web forms are the only way to get IDs
    - When IDR phase 1 is released, the webform will continue to be available
    - IDR can be used in place of the webform
  + Suggestion that we define a community-based scrum process

092220

Relevant links

* [Sprint Reviews](https://github.com/CVEProject/cve-services-doc/tree/master/sprint-reports)
* [AWG Services](https://github.com/CVEProject/cve-services-doc)
* [Entry Submission and Upload Service (ESUS) API](https://github.com/CVEProject/cve-services-doc/blob/master/api-endpoints.md)
* [Demo of the ESUS service](https://www.youtube.com/watch?v=N_P3AwbCLH4&feature=youtu.be) (still in progress)
* [JSON Schema](https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss)
* [IDR OpenAPI YML file](https://github.com/CVEProject/cve-services/pull/48)
* [IDR Docker container](https://github.com/CVEProject/cve-services/commit/a9d91fdecf4012f32253e029b70520ad0397ec2d)
* [Security Policy](https://github.com/CVEProject/cve-services/security)

Actions since last meeting

* Updated user stories since last meeting
* Posted stories on Github

Agenda

* Thu: Finalize IDR user requirement stories
  + Comments on Github workboard for IDR Phase 1
    - Issue 91: Phase 1 IDR allows CNAs to reserve IDs for the previous, current and next year, so we closed issue 91 on Github
    - Issue 86: adequately answered on the workboard
  + For IDR Phase 1, the Epic will be amended to include the statement that acceptance criteria apply to all user stories
    - Future Phases might use acceptance criteria in the individual user story rather than at the epic level
* Overview of a polling feature which was found at pollev.com/thutran955 [this might be unavailable now that the meeting has ended]
  + General agreement that anything that drives active participation is a good idea
* Joe: JSON 5.x topics
  + We need a date to review Dave W’s proposed changes to support experimentation
    - No timeline available at this time
    - The use of “x\_” as a prefix is a suggested approach
    - Chandan will submit a pull request
  + Proposed changes
* Set a max length on string properties:
  + Product Version string max length: Request max length = 50,
    - * Longest windows version string found quickly was 38 characters.
      * Github commit strings = 40 characters
  + Description: [{value}]: V4 was 3999 characters. Request new max = 10,000
  + *Was left undefined after debate over how long is long enough. Changing max length value after it has been set requires schema versioning.*
* Use ‘normal’ property names:
  + Request replacing properties with %data\_meta% with %MetaData%
  + Request changing CVE\_data\_meta to MetaData
* Standardized property name syntax:
  + Request all properties follow the same styling, recommend CamelCase
  + *Many property names carried forward through previous versions.*
  + *Comment from devs: Now is a good opportunity to make this change, since V5 changes already require users to update tooling and processes.*
* Remove Serial from CVE\_data\_meta:
  + Can find no usage in the in the published views. If not used in the published it should not be included in the schema.
  + It could be useful to use this for versioning. We also have a modification date, which could also be used for versioning.
  + Removing this field does not preclude versioning support.
* Informational: CVE\_data\_meta.replace\_by change from CSV string to an array of strings
* Send the changes as a separate email to the list and ask people to comments. We will close the voting on Friday.
  + - Are these 5 items appropriate for us to incorporate into the 5.x schema?
* Kris: How to prioritize moving forward
  + IDR rollout: no change since last week

091520

* Thu: push user stories to Github for continued analysis and discussion
* Kris: Let community know what dates we have currently and where we are
* Milestones
  + IDR code complete [Sept 18]
  + CPS changes code complete [Sept 25]
  + IDR deployed for production [Oct 11]

On Tuesday 22 Sept the following will be available:

1. On Github:  source code of complete IDR phase 1code/unit test cases for review, postman scripts to exercise the operational interface,  YAML files to define interface for programmatic testing
2. On the AWS dev web site, a full IDR Phase 1 implementation for community to exercise with mock data
3. Suggestions for community testing:
   1. Download the source, compile it and run it in a docker image, play with the interface
   2. Create a client using the SAML files as an interface specification and interface with the AWS version (need to the URL)
   3. (need some help here on what else to say (and what not to say)
4. What to do if issue are found:
   1. Emergency  (IDR falls over who do I call)
   2. Bug is found (where to report it)
   3. Where to I ask interface questions (I am coding things up and it seems to be broken)

090820

cancelled

090120

Relevant links

[Sprint Reviews](https://github.com/CVEProject/cve-services-doc/tree/master/sprint-reports)

[AWG Services](https://github.com/CVEProject/cve-services-doc)

[Entry Submission and Upload Service (ESUS) API](https://github.com/CVEProject/cve-services-doc/blob/master/api-endpoints.md)

[Demo of the ESUS service](https://www.youtube.com/watch?v=N_P3AwbCLH4&feature=youtu.be) (still in progress)

[JSON Schema](https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss)

[OpenAPI YML file](https://github.com/CVEProject/cve-services/pull/48) for IDR

[IDR Docker container](https://github.com/CVEProject/cve-services/commit/a9d91fdecf4012f32253e029b70520ad0397ec2d)

[Security Policy](https://github.com/CVEProject/cve-services/security)

Overview: We will be making changes to the way the AWG meetings are conducted and wanted to present those to the community.

* Kris: discuss changes to AWG meeting
  + Thu will facilitate AWG meetings, in part by story boarding the requirements
  + We can use the first couple story boarding sessions as a review of IDR and as a dry-run for the other services
  + Ask community if next week should be cancelled [day after vacation, some devs are on vacation]
* Thu: here’s what we propose to do:
  + Discuss existing system roles for users of IDR MVP:
    - Secretariat
    - CNA
  + Create user stories with acceptance criteria to document IDR MVP requirements
  + Asking for community to participate in the joint effort:
    - MITRE development team will create a starting point for the user stories to share with the community for discussion & review
    - Review the IDR epic (grouping of related user stories)
      * As a user I want to use an on-demand ID Reservation Service so that I can reserve CVE ID(s) without having to make a request to the CVE Secretariat.
  + Putting together user stories during this meeting was a bit of sudden turn and should be better postponed until the next meeting so participants are prepared for the discussion
  + A smaller team was identified to help work on user stories
    - Dave, Kent, Martin, Fabio
  + Revisit collaboration tools
    - In the short term, we should stick to the use of the existing tools.
    - We should continue to use the email list for the meeting notes but should augment that by adding them to the Github site as well.

082420

* Lew: IDR currently on track to be deployed in the test environment by end of week
  + The AWS instance will be available and we will populate it with the latest version of the code. As the code is modified, we will push that code to that AWS instance and publish the source code to Github
* Matt: overview of the IDR MVP requirements [link to IDR requirements on Github]
  + We should tag code so it is labelled and the community knows which branches have been deployed
  + Sept. 9th we should be done with the final code changes. One example is ensuring atomicity in a JSON DB. Once complete, that code will be uploaded to Github and deployed to the AWS instance.
  + We might need to make additional changes to the JSON 5.0 schema to accommodate required functionality in IDR
  + We are planning to be able to deploy a production instance on or after Sept. 25th
* Joe: continued discussion of the JSON 5.x changes to the additional properties
  + How much experimentation and where should that experimentation go?
  + No experimentation is not an option, so the choices are either “some experimentation” [container level] or “all experimentation”
  + We need to pick a direction. Do we allow experimentation at the container level or throughout the schema?
  + Remove additional properties on the description. We can limit the size of the CNA limit. Currently, the description is limited only by the default JSON size.
  + Two conversation: how do we provide extensibility in description and what does extensibility look like in JSON? Keep a limit on size so it doesn’t get too large. If there’s a need to extend something, the CNAs should work together to agree, then add it to the JSON while ensuring backwards compatibility.
  + We could remove “additional properties false” and add a naming scheme for additional properties like “x\_”
    - The challenge with x\_ is cross-CNA naming
  + 3 options to choose from currently:
    - Leave as is with container level experimentation
    - Modify container.description property (Dave and Chandan to mock up)
    - Remove all additional property restrictions within containers
  + A choice needs to be made, so progress can continue.

081820

Relevant links

[Sprint Reviews](https://github.com/CVEProject/cve-services-doc/tree/master/sprint-reports)

[AWG Services](https://github.com/CVEProject/cve-services-doc)

[Entry Submission and Upload Service (ESUS) API](https://github.com/CVEProject/cve-services-doc/blob/master/api-endpoints.md)

[Demo of the ESUS service](https://www.youtube.com/watch?v=N_P3AwbCLH4&feature=youtu.be) (still in progress)

[JSON Schema](https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss)

[OpenAPI YML file](https://github.com/CVEProject/cve-services/pull/48) for IDR

[IDR Docker container](https://github.com/CVEProject/cve-services/commit/a9d91fdecf4012f32253e029b70520ad0397ec2d)

[Security Policy](https://github.com/CVEProject/cve-services/security)

* Lew: ESR approved
  + We are waiting for the AWS instance to be created. Once available, we will create a test environment so people can test their client-side code. We hope to have that by next Friday.
  + An email to the CNAs is being drafted and should go out in advance of the deployment
* Lew: Matt is unavailable today, but his agenda item was discussing the email to the CNAs soliciting information for the User Registry (the last sub-bullet above)
* Joe: update on the IDR conversion
  + Mocker object setup, so tests run properly.
  + CPS should be ready to test against IDR when it is available for testing. This is only relevant insofar as CPS interaction with IDR is a pre-requisite for IDR to go into production.
* Joe: continued discussion of the JSON 5.x changes to the additional properties
  + One example of the problem is that the description field could become a wall of text.
    - This results from CNA guidance directing CNAs to describe the vulnerability and all products and the how each product is affected in the description field, instead of recording the product and affects within the corresponding schema properties.
    - CVE rendering/display should be separated from property data.
    - Rich text useful for explaining how a vulnerability affects a set of products
  + “Description” is intended to uniquely describe the vulnerability. Other parts of the CVE data structure can expand on more detail and provide additional info. The description is intended to be a unique identification of the vulnerability. The CVE structure has an affects field with vendor, product, version which provides additional information and context to the description. In JSON 5.x we introduced an impact and metrics block and those additional properties in the schema might provide an alternative to the “wall of text” content in the description.
    - All of these allow us to have effective descriptions without bloat. This should be a sufficient solution, but without bloating the description field. There’s a concern about a wall of text in the description field as it makes things less useful, not more useful. It’s a question of how we organize the information so that’s it is easy to consume.
  + Part of the goal is to allow some freedom to allow tailoring, so there’s value in having flexibility.
  + How should we organize the information in a sensible way?
  + We can normalize information by providing defined properties, while additional properties allows non-normalized input (must still be valid JSON).
  + If we captured the information well, we might be able to auto-generate descriptions from the information provided, similar to how Mad Libs work.
    - We don’t currently capture all the information that would be needed for this, so we’re in an intermediate state where we still need the description field.
    - 5.x might allow us to start extracting this info from the fields
    - If we agree on driving towards the goal of autogenerating descriptions, then this issue should be viewed as an opportunity to move towards that goal. We should look for ways to get there without causing bloat.
  + We should update the documentation so that the guidance on populating the JSON schema aligns with the intended use of the 5.x schema. This would move us towards an auto generation capability. CNA guidance should direct CNAs to place data within CVE properties as defined by the CVE schema definition, instead of overloading the description property with duplicative content.
  + What is our position on extending properties within the containers? When we talked about removing additional properties that was also described as a general approach for all content.
    - There are reasons developers and downstream users want to avoid that. It would require the consumption of the entire object, then you’d need to parse out the pieces you want. You couldn’t just get the default CNA container.
  + We could use ESUS to enforce limits on the JSON (such as file size).
  + No change option, the defined container limits additional content limited locations within the schema, which limits bloat to outside of the defined fields.
  + ESUS will preserve data as it is sent. If there’s a wall of text, it will continue to be in the JSON, it just won’t be mingled with the other objects.
  + As a future enhancement, ESUS would allow consumers to request only the ‘core’ defined properties of the schema they’re interested in be returned. Thus ignoring additional and experimental content.
  + Allow at the container property level, which allows us to experiment within a container.
  + Question: Is there a list of what’s considered “core”? Answer: it’s what is defined in the 5.13 version on the Github repo.
    - Question: If we’re fighting bloat, but they can still bloat within the container, then what value is there in restricting it to the core if the core allows bloat?
      * Answer: If it’s not defined as required, it wouldn’t be part of the core.
  + The ease of filtering appears to be primary benefit of leaving the JSON as-is.
    - Would be good to have input from the broader community and the downstream users. [add this to executive summary in the notes email]
  + Trying to figure out how we support extensibility in the 5.x format. There’s a recognition that our needs will evolve, and the current format will be insufficient. We want to experiment with different ways to express data so we can identify new and agreed-upon ways of expressing data.
    - What are the rules for extensibility in the 5.x format? Is it full? Only in containers?
  + We need a way to support extensibility. It’s not clear what the community’s tolerance for extensibility is. Is it limited to parts of the schema, or everywhere?
    - Some support for limited extensibility because it makes it easier to scan the schema and find it.
    - Also support for allowing greater extensibility.
  + From implementation perspective, limited extensibility. Are we saying in addition to the other fields, we will have an extensibility object within each container? Yes, this would be additional properties that are not explicitly defined with the schema.
    - Is there an intention to constrain how things are named or will it be the wild west of naming? A: we could require a prefix on properties. Property names are part of a name space and we might want to constrain that name space for official use. Or we could define a single property under which all extensions go (which is the wild west approach).
    - Using prefixes makes a lot of sense. Using the CNA-name as the prefix would enable to community to rally around an experimental property and avoid duplication of tags between CNAs/ADPs. The CNA-name prefix will also ease data migration when experimental properties are merged with the CVE schema, as the property name will indicate that property value is the same structure regardless of occurrence within the CNA-container or multiple ADP-containers. Using CNA-name prefix potentially eliminates the need to establish is static prefix or property registry.

081120

Note: CNAs and ADPs can write their IDR client side code using the [OpenAPI YML file](https://github.com/CVEProject/cve-services/pull/48) and the [IDR Docker container](https://github.com/CVEProject/cve-services/commit/a9d91fdecf4012f32253e029b70520ad0397ec2d)

Community decisions under discussion

We discussed 3 approaches for treating additional properties:

* Note: JSON schema and examples below are for discussion only, they are not intended to be syntactically valid not complete CVE examples.
* No Change – Rich Description in containers  
  A screenshot of a cell phone

  Description automatically generated
* Remove no additional properties from descriptions  
  A screenshot of a social media post

  Description automatically generated
* Nest additional properties under language array  
  A screenshot of a map

  Description automatically generated
* Change impact considerations:  
  A screenshot of a cell phone

  Description automatically generated

Power Point version posted to AWG github:   
<https://github.com/CVEProject/automation-working-group/raw/master/cve_json_schema/v5.x_discuss/change_discussion/enriched%20descriptions.pptx>

Relevant links

[Sprint Reviews](https://github.com/CVEProject/cve-services-doc/tree/master/sprint-reports)

[AWG Services](https://github.com/CVEProject/cve-services-doc)

[Entry Submission and Upload Service (ESUS) API](https://github.com/CVEProject/cve-services-doc/blob/master/api-endpoints.md)

[Demo of the ESUS service](https://www.youtube.com/watch?v=N_P3AwbCLH4&feature=youtu.be) (still in progress)

[JSON Schema](https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss)

[OpenAPI YML file](https://github.com/CVEProject/cve-services/pull/48) for IDR

[IDR Docker container](https://github.com/CVEProject/cve-services/commit/a9d91fdecf4012f32253e029b70520ad0397ec2d)

[Security Policy](https://github.com/CVEProject/cve-services/security)

Raw notes

* Lew: update on the ESR status
  + YML file and Docker are available
  + IDR code deployed and the ESR has been submitted
  + Once approved, it will take a couple days to deploy the code in the publicly available instance because it’s not a fully automated process. Best case, it will be available next Friday but that assumes everything goes perfectly
* Joe: worked through some internal environment issues and resolved them.
  + This did impact the unit tests, so work on those continues
* Joe: suggested JSON change from Chandan

Currently proposed schema has a line additionalProperties: false for the description entry:

<https://github.com/CVEProject/automation-working-group/blob/master/cve_json_schema/v5.x_discuss/cve513.schema#L601>

This will prevent possible future improvements or extensions tied to a CVE description.

I suggest we remove that line from the schema.

We could add the additional property at the container level rather than expanding the description block. This could lead to ambiguity, though. At a minimum there should be a textual description. If there are alternate formats, they should be with text itself, or a cross-link if they’re in different areas. Perhaps build something structured so that tools that are parsing the schema would know what the alternates are. We should probably do something in the short term that addresses the problem, but leave room for future changes. We could have another field like “alternate” that have a media type and a value. These fields would be natural language specific.

Should we or how we improve the quality of descriptions would be topics for QWG, but I prefer we keep the schema flexible here for future-proofing purposes.

An example of an improvement is to provide an alternate rich-text version of the same description:

{

          "lang": "EN",  
          "value": "The loadContentFromCookie function in core/Cookie.php in Piwik ...",  
          "md": "The `loadContentFromCookie` function in `core/Cookie.php` in \*\*Piwik\*\* ...",

}

Another hypothetical example is for an audio/voice rendering of the description text

{

          "lang": "EN",  
          "value": "The loadContentFromCookie function in core/Cookie.php in Piwik ...",  
          "spoken": "[http://example.org/audio/..](http://example.org/).",

}

Changing it at the descriptions level has a minimal impact, doing it at the container level has no impact. We want the community to have a sandbox to play in, providing a kind of constrained freedom.

A container could have alternates within a language, the alternates could be an array of objects with a media type and a value, but no additional properties allowed so there’d be an “alternates” property at the root of the container where semi-arbitrary content can go.

Containers can support additional properties and the current properties are scripted to work with the JOSN 4.0 format, but there’s no good way to roll additional content into the old format. We could simply drop that content if we are back-porting from the 5.x schema to the 4.0 schema.

We might want uniformity across CNAs description blocks (i.e. each CNA should be taking the same approach). We want the structure of the description block to be consistent because that will impact the usability of the additional properties. Concern that CVE consumers would be required to have in interpreter for each CNA.

Things enshrined under the description (lang and value) are the things we agree all parties should use uniformly. Additional properties added above and beyond that are left to the extension by the container owner and should be treated as specific to that container owner. That allows the container provider to have the freedom to do things their own way.

Modifying the schema at the CNA container level leads to different flavor of content within a container, but the minimum information that’s agreed upon will still be accessible. This might provide the balance we’re looking for between uniformity and freedom to experiment. CNAs could use this an opportunity to make incremental changes by experimenting with an idea first, then proposing changes to the schema to incorporate those ideas formally.

I'd [Matt B] even propose a slight modification that I think would work well:

**Text**

descriptions: {

en: [

{

media\_type: "html|json|link|whatever",

body: "the link, the json, the html"

}

],

ger: [

{

media\_type:,

body:

}

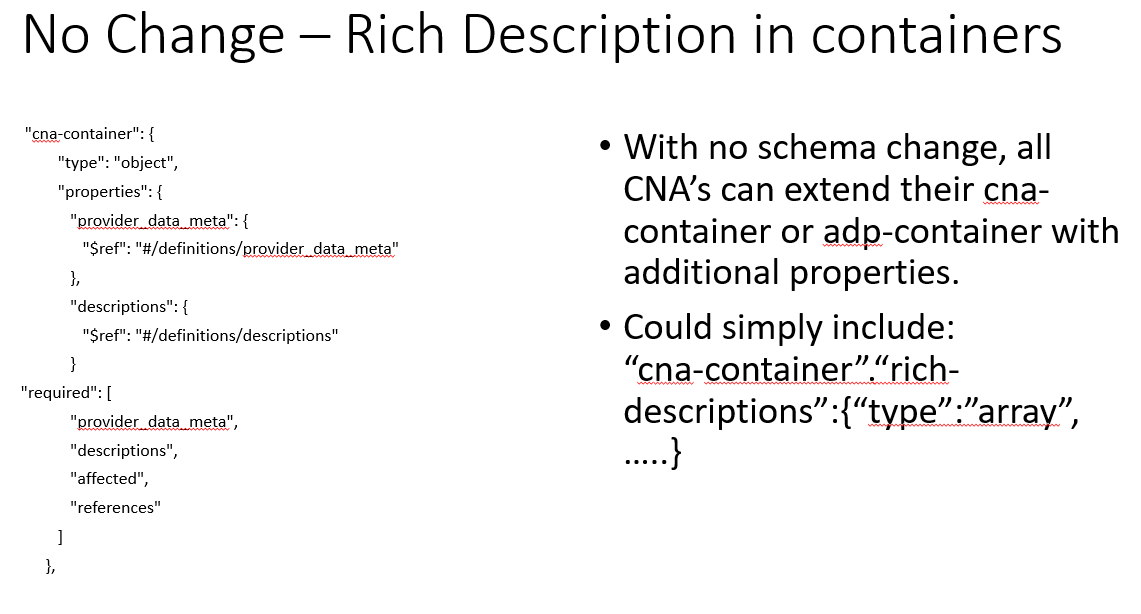
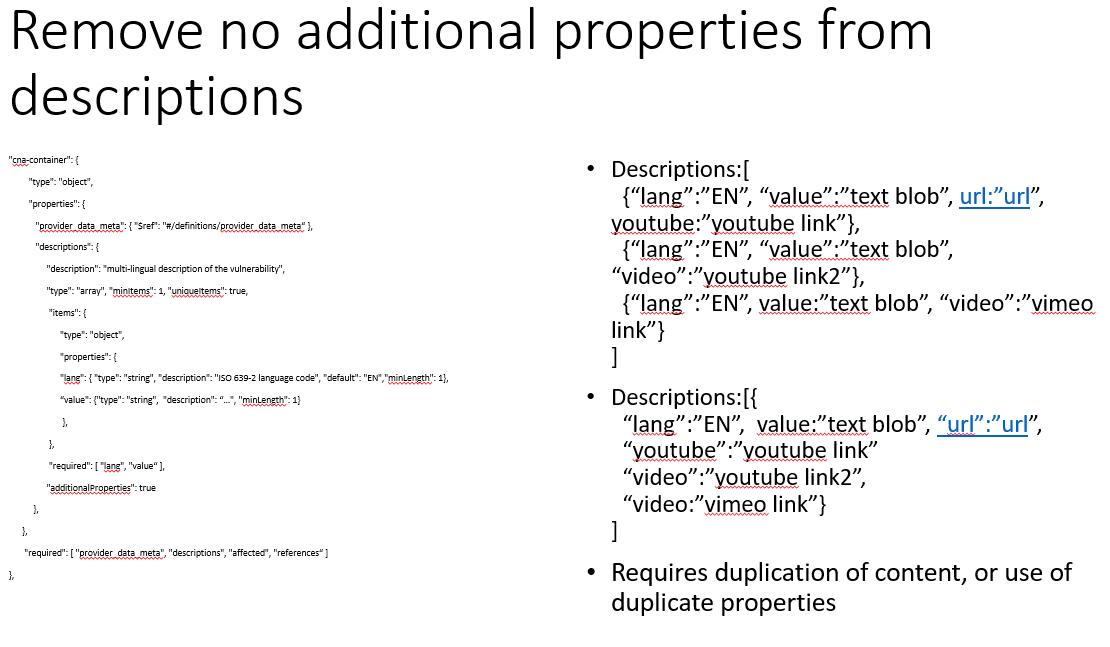
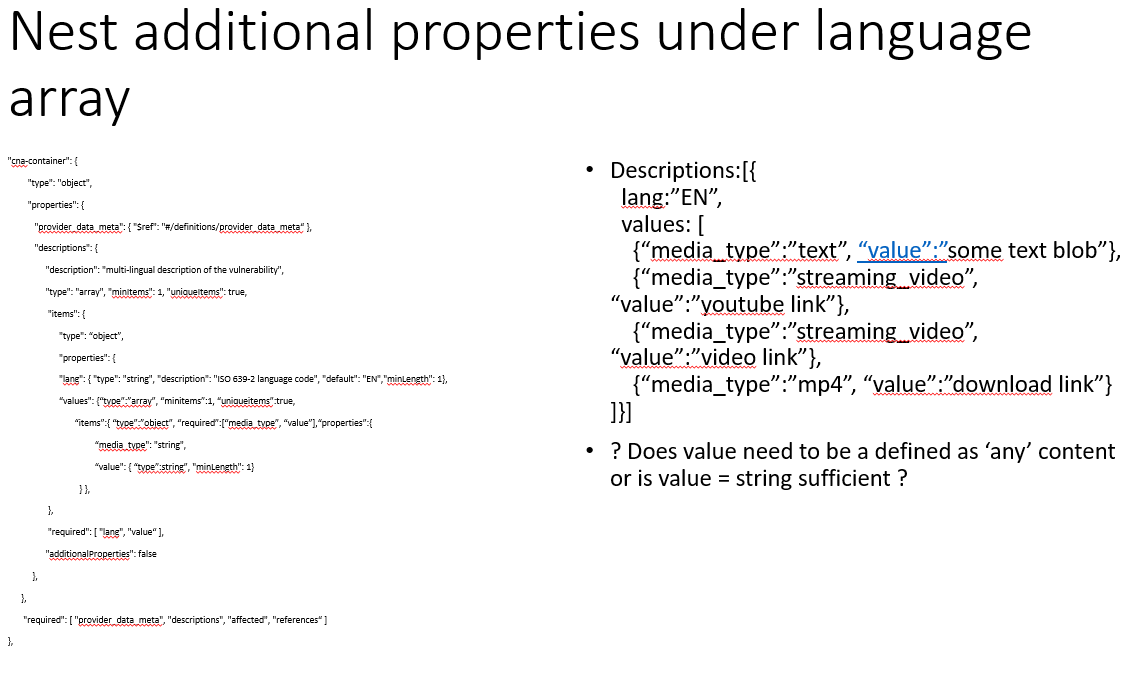
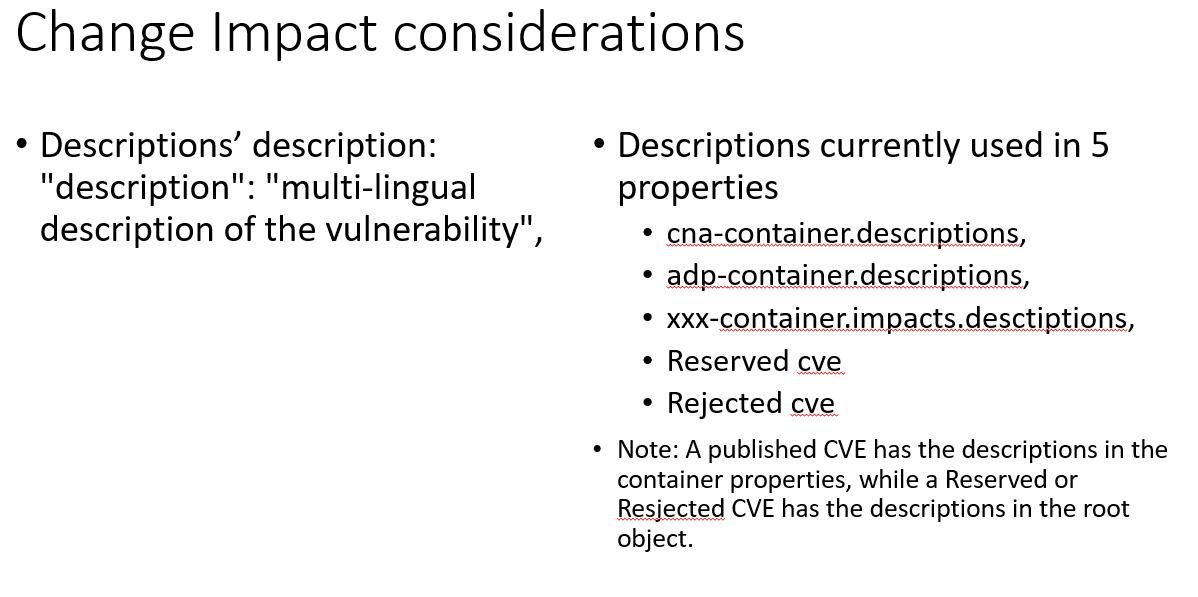
]

}

We could track the use of these variations so we would have a sense of when new approaches are being used, how often they’ve been used etc.

[Jsonapi.org](http://jsonapi.org/) has its own take on various topics and might provide some mechanisms for addressing this issue.

We discussed 3 approaches for treating additional properties:

* Note: JSON schema and examples below are for discussion only, they are not intended to be syntactically valid not complete CVE examples.
* No Change – Rich Description in containers  
  
* Remove no additional properties from descriptions  
  
* Nest additional properties under language array  
  
* Change impact considerations:  
  

Power Point version posted to AWG github:   
<https://github.com/CVEProject/automation-working-group/raw/master/cve_json_schema/v5.x_discuss/change_discussion/enriched%20descriptions.pptx>

Once we’ve worked out the options in the AWG, we will send it to the SPWG for selection.

* Matt: We will use fake credentials for the staging system, just to be extra cautious [three votes in email for fake credentials, and no objections to fake credentials in the last AWG meeting]
* Matt: Look for an email about getting your account set up
* Matt: user registry endpoints completed
* Matt: Discuss the static site generator (Jekyll or Hugo, or other?) added to Github site as a topic of conversation

080420

Questions for the community

* We can use fake credentials or real credentials.
  + Using production credentials during the testing phase could create risk, while using fake credentials for testing is less risk, but it means participants would have two credentials to keep track of.
  + The dev team is leaning towards the use of fake credentials for the staging system, just to be extra cautious, but what does the community have a preference?

Relevant links

[Sprint Reviews](https://github.com/CVEProject/cve-services-doc/tree/master/sprint-reports)

[AWG Services](https://github.com/CVEProject/cve-services-doc)

[Entry Submission and Upload Service (ESUS) API](https://github.com/CVEProject/cve-services-doc/blob/master/api-endpoints.md)

[Demo of the ESUS service](https://www.youtube.com/watch?v=N_P3AwbCLH4&feature=youtu.be) (still in progress)

[JSON Schema](https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss)

[OpenAPI YML file](https://github.com/CVEProject/cve-services/pull/48) for IDR

[IDR Docker container](https://github.com/CVEProject/cve-services/commit/a9d91fdecf4012f32253e029b70520ad0397ec2d)

[Security Policy](https://github.com/CVEProject/cve-services/security)

* Lew: Update on IDR status and schedule
  + Code deployed for ESR review this week
  + Hoping for approval in 1-2 weeks
* Matt: <https://github.com/CVEProject/cve-services/issues/54>
  + We’re working on a project site and posted this to provide some tasks to which the community contribute.
  + We should determine what framework to sue for a static website. [Jekyll](https://jekyllrb.com/) is built in, so that might be convenient, but there are alternatives.
* Matt: can post a form on the CNA site so people can fill in the info so we can create accounts for them
  + We can develop a Microsoft form and put a link to it on the Sharepoint site, if that works for everyone.
  + People can provide an email address and additional info and we can send a GPG key to that email address.
  + If you don’t want to interact with IDR, it won’t be necessary as the webform access will still be available.
* Matt: for IDR testing, we can use fake credentials or real credentials. Does the community have a preference? [add this to the list of questions for the email distro]
  + Using production credentials during the testing phase could create risk, while using fake credentials for testing is less risk, but it means participants would have two credentials to keep track of.
* Joe: IDR changes to CPS moving into the test and debug phase

072820

Relevant links

[Sprint Reviews](https://github.com/CVEProject/cve-services-doc/tree/master/sprint-reports)

[AWG Services](https://github.com/CVEProject/cve-services-doc)

[Entry Submission and Upload Service (ESUS) API](https://github.com/CVEProject/cve-services-doc/blob/master/api-endpoints.md)

[Demo of the ESUS service](https://www.youtube.com/watch?v=N_P3AwbCLH4&feature=youtu.be) (still in progress)

[JSON Schema](https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss)

[OpenAPI YML file](https://github.com/CVEProject/cve-services/pull/48) for IDR

[IDR Docker container](https://github.com/CVEProject/cve-services/commit/a9d91fdecf4012f32253e029b70520ad0397ec2d)

[Security Policy](https://github.com/CVEProject/cve-services/security)

* Lew: does the community want to participate in sprint planning and backlog grooming?
  + The attendees indicated a willingness to provide feedback on the process and the approach
  + Some participants are interest in development, but prefer to restrict their participation to that part of the process
  + Interest in selecting specific tasks from the workboard, complete the issue and move on
  + It would be good to have small shovel-ready tasks on the workboard [Matt has already started this process]
  + There is value in having community participation in the backlog grooming, but the time commitment could prevent some from participating
  + A model where we release often and frequently could help to encourage participation [this is what we’re aiming for]
  + Need crisp effective sprint planning, but it will take some time to get everything organized to support a community-based agile process
  + The readme file needs to be updated to include additional information about how to participate. This is what we’re doing, this is how you join, etc. [Matt working on updating the readme]
* Lew: bugs can be reported on the workboard, but we need a process for managing vulnerabilities in the AWG services that isn’t immediately public. Should we use the AWG list, or should we create a new email address for receiving vulnerability reports? We need something secure but simple. We will need a process for this that is posted on Github (should also be on the new website).
  + It was suggested that the Secretariat could create an email address used specifically for this purpose.
  + Broader question: if there’s a vulnerability and the fix is to patch it, it’s straightforward, but if there are backwards compatibility issues or client-side changes, then we need a communication mechanism that supports that
    - We could communicate directly with the CNAs to discuss such things
    - We need to be transparent and fast about dealing with such issues
  + We should stand up the email address for receiving such info, perhaps [security@cve.mitre.org](mailto:security@cve.mitre.org) until we have a new domain name.
    - We might be able to set up a notification policy in Github that alerts people when a vulnerability has been fixed
  + Is there an easier communication mechanism than creating a new email address? Maybe something through Github?
    - There appears to be a way to keep info private on Github, we would need to explore this to see what is offered and how we could set it up. The security disclosure policy might provide a mechanism in Github. [Matt started to stand this up [here](https://github.com/CVEProject/cve-services/security)]
  + Having a policy and an email address would provide a workable solution in the short term then transition to a more robust approach when time permits.
* Lew: Chris L asked that we add the Sprint review and planning process to the agenda
  + List artifacts and what they are intended to communicate, skeletal structure of sprint planning
    - **Architecture diagrams**
    - **Product vision statement:**a summary that communicates how the services support the program’s strategies and articulates the goals for each service.
    - **Product roadmap:** a high-level view of the product requirements needed to achieve the product vision. Provides a general timeframe for when the requirements will be developed and released.
    - **Product backlog:**The full list of what is in the scope for your project, ordered by priority. After you have your first requirement, you have a product backlog.
    - **Release plan:**A high-level timetable for the release of software.
    - **Increment:** The product functionality demonstrated to stakeholders at the end of the sprint.
    - **Sprint Retrospective**
* Discussion around incorporating CPE information
  + **CPE information in the API** – JPMC are doing some additional work to take the current CPE information in a CVE and expand it via the match CPE file in order to pull the full list of CPEs.
    - Is it possible to have a CPE endpoint to list all the CPEs in a given range in order for us to have the fill list of CPE URIs for a given vulnerability?
    - An internal team might be working on DB as a service, they have core technologies that they want information on, and the CPE is one such mechanism for doing that. Right now, a CVE will provide a high-level description (i.e. everything below this version is affected). It would be nice to have an API call that permits people to search for such info.
      * This is being discussed in the QWG, so this would be a good issue to raise in that venue.
    - **References to OWASP in CVE v5 Schema:** As a coincidence I sit on the board of the OWASP Foundation (Vice-Chari), and I’m happy to help there where possible happy to explore how we can help CVE.

072120

Sprint Reviews <https://github.com/CVEProject/cve-services-doc/tree/master/sprint-reports>

AWG Services <https://github.com/CVEProject/cve-services-doc>

Entry Submission and Upload Service (ESUS) API <https://github.com/CVEProject/cve-services-doc/blob/master/api-endpoints.md>

Demo of the ESUS service (still in progress)  
<https://www.youtube.com/watch?v=N_P3AwbCLH4&feature=youtu.be>

JSON Schema <https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss>

OpenAPI YML file for IDR <https://github.com/CVEProject/cve-services/pull/48>

* Lew: IDR in playground for ESR testing will be done between July 24 and July 31
  + We need to work on the CICD pipeline in parallel with the IDR deployment to avoid additional delays.
* Lew: uploaded markdown version of the new AWG charter to Github and renamed it to “AWG Charter” from “CAWG Charter DRAFT” <https://github.com/CVEProject/automation-working-group/blob/master/AWG_Charter.md>
* Matt: OpenAPI YML file was updated with some improvements and details
  + Includes a schema for JSON CVE objects and has additional details describing the CVE ID object that you’ll get. Feedback would be appreciated
* Matt: Docker file in the repo, might be helpful for folks trying to run the API
  + Matt: regarding instructions for the Docker, do people want instructions for the current Docker deployment?
  + Instructions for using the Docker container will be posted shortly
* Matt: update and status on the IDR work
  + Lookup endpoints are in, so if you populate your database with sample IDs, you can look those and are described in the YML file
  + We will be finishing up the work on the allocation code
* Matt: OpenAPI file YML file, they used a resource named “CNA” but this is a misnomer because ADPs might also use it. The question is what those interfaces and endpoints should be called. Would “org” be better or is there another term that’s preferred?
* Joe: quick update
  + Working on the conversion of internal tools to use IDR, it’s looking good so far but still working through it. Writing the test cases and testing the new software. When it’s going well, you start to wonder what you missed.
  + Thanks to the community for input on 5.0 schema. Before we release the update to 5.13 we will collect the comments and incorporate them all at once. If there are other questions please send them to the email list and we will respond when we can, but they won’t be rolled in until we have IDR implemented.
* In a few weeks we will begin Sprint planning
  + We need to ensure that our Sprint process is ironed out. How do we get stories in, how do we map requirements to features?
  + Matt: has the compilation of the MVP requirements for each of the services. We could post those to the CVE services document. These get translated to requirements that get posted to the work board.
* Sprint planning will take some time, 4-6 hours? Given that we want community participation, 4-6 hours is a lot.
* What kind of weeklong process works? 5 1-hour meetings? 2 2-hour meetings?
* We need to detail the set of artifacts that will be produced (architectural diagrams, tech stacks, design documentation, every service should have a checklist and instructions)
* We could establish backlog grooming meetings to help prepare for the sprint planning, this could make the sprint planning more efficient.
* Sprint planning could be conducted internally but communicated to the community in the AWG meetings. The MITRE team could prepare a summary of what was discussed and decided so that it could be communicated easily to the community
  + This would be effective, but doesn’t provide a participatory environment for the community
  + So, how do we conduct Sprint planning in a manner that makes it easy for stakeholder participation?
* Need a document that explains the context for the AWG work. What’s the goal? Why are we doing this? How does it relate to the program goals?
* Once the code is deployed, contributions from the community could be made by suggesting improvements, identifying bugs, etc.
* Once IDR deployed, maybe the devs could put together some proposals for Sprint Planning and propose those to the community. The idea would be to let the MITRE devs do the heavy lifting, while still giving the community an opportunity to participate. Devs identify artifacts, a recommended time length for the first sprint, understand where all the material will be posted and available and document risks. A transparent process that everybody can review and comment on. We should have checkpoints within a Sprint. Any interested in the community should be able to participate in a manner of their choosing.
* Perhaps the community members could propose ways that we could organize sprint planning to make it easier for people to participate and review the process in a transparent manner.
* Next meeting, we should add the Sprint review and planning process to the agenda.
* For next AWG meeting: List artifacts and what they are intended to communicate, skeletal structure of sprint planning

071420

* Sprint Reviews <https://github.com/CVEProject/cve-services-doc/tree/master/sprint-reports>
* AWG Services <https://github.com/CVEProject/cve-services-doc>
* Entry Submission and Upload Service (ESUS) API <https://github.com/CVEProject/cve-services-doc/blob/master/api-endpoints.md>
* Demo of the ESUS service (still in progress)  
  <https://www.youtube.com/watch?v=N_P3AwbCLH4&feature=youtu.be>
* JSON Schema <https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss>
* OpenAPI YML file for IDR <https://github.com/CVEProject/cve-services/pull/48>
* Lew: Vote on AWG Charter The AWG Charter was accepted because there were no objections to it
* Lew: overview of schedule

June 26

* + - External System Request (ESR) architecture diagram drafted
    - OpenAPI YML file for IDR <https://github.com/CVEProject/cve-services/pull/48>

July 10

* ID Library endpoints, User Registry endpoints, algorithm changes (last year allocation)
* ESR submitted, pentest not required [moved from July 10 because the ESR needs to include CPS interactions and changes, this will \*not\* impact CNAs ability to prepare for the release of IDR, work completed early, ESR submitted July 13, testing can commence when SW is deployed]
* Request code review from community
* Provide information to CNAs on how to get accounts for IDR testing Matt plans to start soliciting info on users and account. Input can be sent to us via the AWG list of directly to Lew or Matt. Matt’s working on the distro method for the API secrets, possibly encrypted email.

July 17

* IDR code and DocumentDB Deployed to playground [delayed by some issues with the download script, about 2 days of work left to implement]
* Planning for the CPS-IDR tests
* Schedule planning meetings for the other services [delayed until after IDR release by Chris/Dave by request]

July 24

* External System Request (ESR) response [moved from July 17th due to ESR submission delay] [production code, consider adding a comment to the JSON that marks it as test, not yet live, use unique URLs for the test and production instance]
  + Host a test/staging environment on AWS so people can test on IDA with dummy data to augment current automated tests, we will put out a call for participants [moved from July 17 due to dependency on ESR response, above]
* Conduct CPS-IDR tests

July 31

* IDA goes live and is available to the community for production use [potential delay from testing]
* CPS uses IDA instead of generating its own IDs [CPS changes could cause this date to change, but will not impact communities ability to prepare for the change]
* Matt: Request code review from community
* Matt: Provide information to CNAs on how to get accounts for IDR testing \Matt:

# AWG Notes

1. Tested code with MongoDB 3.6 and tests had no issues, indicates higher probability that DocumentDB transition should be smooth.
2. CVE "library" or "look-up" endpoint code pulled into dev.
3. Substantial progress on ESR documents for submission.
4. Work still being done on insuring ID allocation database transactions will run smoothly in production.
5. Development started on necessary User Registry endpoints for administration purposes

* Joe
  + We have been getting comments on the JSON 5.0 As we get through IDR please post input in the AWG folder for the schema.
  + we will review the comments all at once and update the schema

070720

PLEASE NOTE: the AWG that Charter vote is next week during the AWG meeting. Please plan to attend if you want to vote as this will be a voice-vote only (no email votes). Please review the charter that Kent sent out and, if you have edits, send them by Thursday July 9th so that they can be incorporated into the charter.

As always, in the meeting notes below, black font is used for text that was added as part of the agenda, while purple font indicates text that was added during the meeting.

* Lew: Remind AWG that Charter vote is next week
* Lew: in summary of meeting notes, remind email list that they should review the new charter and provide input prior to next week’s meeting, vote will be in person only. Changes should be sent by Thu July 9th.
* Lew: overview of schedule

June 26

* + - External System Request (ESR) architecture diagram drafted
    - OpenAPI YML file for IDR <https://github.com/CVEProject/cve-services/pull/48>

July 10

* ID Library endpoints, User Registry endpoints, algorithm changes (last year allocation)
* IDR code and DocumentDB Deployed to playground [delayed by some issues with the download script, about 2 days of work left to implement]
* Request code review from community
* Provide information to CNAs on how to get accounts for IDR testing

July 17

* ESR submitted, pentest not required [moved from July 10 because the ESR needs to include CPS interactions and changes, this will \*not\* impact CNAs ability to prepare for the release of IDR]
* Planning for the CPS-IDR tests
* Schedule planning meetings for the other services

July 24

* External System Request (ESR) response [moved from July 17th due to ESR submission delay] [production code, consider adding a comment to the JSON that marks it as test, not yet live, use unique URLs for the test and production instance]
  + Host a test/staging environment on AWS so people can test on IDA with dummy data to augment current automated tests, we will put out a call for participants [moved from July 17 due to dependency on ESR response, above]
* Conduct CPS-IDR tests

July 31

* IDA goes live and is available to the community for production use [potential delay from testing]
* CPS uses IDA instead of generating its own IDs [verifying impact to the CPS code later this week, this will not impact CNA ability to develop their IDA interface code but could cause this date to change]
* Matt
  + Matt added new endpoints for CPS to YML file: <https://github.com/CVEProject/cve-services/pull/48>
  + Cristina worked on library endpoints, Matt worked on User Registry endpoint and functionality speced out and put in Github. CPS-to-AWG data migration and interaction
* Joe
  + Re-worked the tasks in light of the focus on IDR
  + The 5.0 schema and the 5.0 up-convertor need to be reviewed
  + Spent last week redesigning the CPS-AWG integration plan and now we’re working on the code
  + CVE 5.0 Conversion Review Set: <https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss/CVE5.0_Conversion_Review_Set>

* + Latest Draft 5.0 schema (cve513.schema): <https://github.com/CVEProject/automation-working-group/blob/master/cve_json_schema/v5.x_discuss/cve513.schema>

* + Python Conversion Script: <https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss/support/CVE_4_to_5_converter>
  + Sample output log results included.
  + sample command:           python 4to5up.py -d orig\2020 -o 5\2020 > 2020run.txt

Suggest we reorganize Github repo around the JSON schema. Possibly rename schema to a stable name and track changes over the history of the schema. Joe: we will move to that approach in the future and will abandon the use of different files for each version. Schedule this for July 31 after IDR is released but this needs to occur prior to the ESUS release.

063020

* Kent: New AWG Charter
* Matt: we think issue 44 is set (<https://github.com/CVEProject/cve-services/issues/44>), if anyone else would like to review it and provide feedback in case there are concerns about how it’s being implemented
* Lew: When a request for IDs is made, we should also return their “balance” (i.e. you requested X and that leaves you with Y remaining IDs)
  + We can add this as a post-Phase 1 functionality
  + Does the community want this? yes, members want this
* Lew: We will need a day where no IDs are handed out so we can transition from CPS to IDR
* There will be an AWG charter vote in two weeks

**AWG Road Map**

June 26

* External System Request (ESR) architecture diagram drafted
* OpenAPI YML file for IDR

July 10

* ID Library endpoints, User Registry endpoints, algorithm changes (last year allocation)
* IDR code and MongoDB Deployed to playground
* Request code review from community
* Provide information to CNAs on how to get accounts for IDR testing

July 17

* ESR submitted, pentest not required [moved from July 10 because the ESR needs to include CPS interactions and changes, this will \*not\* impact CNAs ability to prepare for the release of IDR]
* Host a test/staging environment on AWS so people can test on IDA with dummy data to augment current automated tests, we will put out a call for participants
* Potentially with a load testing schedule
* Begin schedule planning for the other services

July 24

* External System Request (ESR) response [moved from July 17th due to ESR submission delay] [production code, consider adding a comment to the JSON that marks it as test, not yet live, use unique URLs for the test and production instance]
* Plan CPS-IDR tests with Hale, Joe, Matt,

July 31

* IDA goes “live” and is available to the community for production use [I
* CPS uses IDA instead of generating its own IDs [verifying impact to the CPS code later this week, this will not impact CNA ability to develop their IDA interface code but could cause this date to change]

062320

* Sprint Reviews <https://github.com/CVEProject/cve-services-doc/tree/master/sprint-reports>
* AWG Services <https://github.com/CVEProject/cve-services-doc>
* Entry Submission and Upload Service (ESUS) API <https://github.com/CVEProject/cve-services-doc/blob/master/api-endpoints.md>
* Demo of the ESUS service (still in progress)  
  <https://www.youtube.com/watch?v=N_P3AwbCLH4&feature=youtu.be>
* JSON Schema <https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss>
* Dave/Chris overview of open source contributions
  + How can we convey information better?
  + How can we make it easier for people to contribute?
    - Issue tracking using labels to identify low-hanging fruit (i.e. tasks that are relatively easy) is a good way to get contributions. The AWG will try to start implementing that now that we have solid requirements.
    - It doesn’t have to be coding, it could be requests for documentation or explanations of terms
    - Additional documentation on the Github site would be a good step forward
    - Classifications by skill (Python experience, or knowledge of a given library) helps people identify tasks that are aligned with their abilities and interests.
    - Are there any corporate constraints to contributions that we could try to alleviate? No issues were identified in the meeting.
* Matt: overview of ways to contribute
* Matt: provide context that we’re adding automation in phase 1 and in subsequent sprints we’ll move towards more self-service
* Lew: I need to add a few emails to the AWG email list, but I get a “Bad Gateway” message when I try. It’s an unintended side-effect of an upgrade they made last week. The infrastructure folks are aware of it and are working on the problem.
* Lew: Explain to the AWG how they can participate in defining the information architecture by participating in the next OCWG meeting on the 26th

ID Reservation System Schedule

June 26

* External System Request (ESR) architecture diagram
* OpenAPI YML file for IDR

July 10

* ID Library endpoints, User Registry endpoints, algorithm changes (last year allocation)
* Request code review from community
* Provide information to CNAs on how to get accounts for IDR testing

July 17

* Deployed to playground and ESR submitted pentest not required [moved from July 10 because ESR review needs to include CPS interaction]
* External System Request (ESR) response [production quality, test URL, consider adding a comment to the JSON that marks it as test, not yet live]
* Host a test/staging environment on AWS so people can test on IDA with dummy data to augment current automated tests, we will put out a call for participants
  + Potentially with a load testing schedule
* Begin schedule planning for the other services

July 31

* IDA goes “live” and is available to the community for production use
* CPS uses IDA instead of generating its own IDs [verifying impact to the CPS code later this week, this will not impact CNA ability to develop their IDA interface code but could cause this date to change]

Open questions / risks

* Need to verify DocumentDB API 3.6 vs MongoDB 4.2 [key decision point, July 2 confirm with Colin]

Kent will work on a draft of the AWG charter, based on the SPWG charter, and will provide that to Dave W. for review and discussion in the AWG.

061620

* Sprint Reviews <https://github.com/CVEProject/cve-services-doc/tree/master/sprint-reports>
* AWG Services <https://github.com/CVEProject/cve-services-doc>
* Entry Submission and Upload Service (ESUS) API <https://github.com/CVEProject/cve-services-doc/blob/master/api-endpoints.md>
* JSON Schema <https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss>
* Question: Do we need strong 2 Factor Authentication (2FA) for the AWG Services? If so, why?
  + We were asked if this is 2FA or strong 2FA. Answer: strong.
  + It was asked if there are any corporate guidelines requiring 2FA and no one in attendance reported any.
  + Assertion: we should plan for it in the future
* Lew: note the delay and let them know when IDA will be made available.  3 milestones:
  + fully functional code they can stand up locally (2 weeks)
  + a test instance in AWS (4 weeks)
  + production ready and deployed (cps changes, subject to pen tests, bugs, issues) and risks that could delay these. If there are no issues from the code review and pen test, this could coincide with the previous milestone; if issues are found, it could take a few weeks to address them
* Lew: ESUS put on the back burner until we get IDA deployed.
  + To ensure that the ID Allocation (IDA) service is deployed and available in a timely manner, we are pulling some resources from the Entry Submission and Upload Service (ESUS) to IDA
  + We will move developer resources back to ESUS at the start of August
* Joe: Schema translators are delayed by User Registry and reprioritization of IDA
* June 15 Joe email:
  + AWG -> change ASSIGNER from email to UUID from user data registry
  + UDR not online yet
  + ASSIGNER field required
  + Current conversions fail validation because of this.
  + Options:
  + - Change schema to be UUID or email (probably our best route)
  + - upload converter and 5.0 set as is with email in assigner field (and others swapped to UUID) for early review, then run conversion again when UDR is live for finally corpus conversion. Converter update would include the UUID look up. (probably the path we want to take)
  + Q: can we generate the UUID’s now and upload them into the User Registry when we’re ready? A: there might be a way to do it, worth exploring.
  + Three options:
    1. Upload converted set with email in place of UUID
    2. delay the upload until we can replace emails with UUID
    3. Adjust the schema to be email or UUID
  + It was suggested that we pursue the first two options in parallel, starting with option 1, then moving to option two when we can replace the emails with a UUID. This was acceptable to those in attendance.
  + We will send an email to the AWG, SPWG, and CNA email list as Joe makes progress on the JSON up-convertor and any changes to the schema.
* June 11 Joe email:

I think we poorly defined the problemtypes field in 5.0, and we need to change its schema. Simplified proposed solution included here:

5.0 ProblemTypes definition

We have the structure wrong

Think we want (proposed 5.0 change, would use definitions/descriptions for the descriptions property, and definitions/references for references):

[

    {

        descriptions:[

            { lang:eng,

            value: text} ],

        CWE-ID:ID STRING,

        references:[],

    }

],

additional properties: stuff,

Not this (current 5.0 schema):

[

    {

        descriptions:[

            {

                lang:eng, (required)

                description:string, (required)

                CWE-ID: cwe-id string,

                type: string,

                references:[],

            }

        ],

        additional properties: stuff,

    },

],

060920

Sprint Reviews <https://github.com/CVEProject/cve-services-doc/tree/master/sprint-reports>

AWG Services <https://github.com/CVEProject/cve-services-doc>

JSON Schema <https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss>

Questions from the meeting

* Should we require CNAs to make an explicit action to change the state of a CVE, or should we infer state changes from the content of the JSON? The inference could result in people accidentally making a CVE public before they were ready, while performing an explicit action requires them to do more
* How do people plan to use the GET CVE endpoint? What searches do you want? What should we implement first? We will start with CVE by provider data meta or by ID, but other use cases and queries would be helpful for us to know

Meeting Notes

* Matt: GraphQL questions for AWG
  + The benefit of GraphQL is providing a language for client and servers to communicate
  + We were hoping to tie it into the data model, but we don’t get that out of the box, so we will need to build it
    - Matt will create an issue on Github so people can post input there, and will be available on GITR chat at <https://gitter.im/cve-awg/community>
  + We will develop 3 endpoints: 1 for the entire schema, 1 for the CNA container and an endpoint for updating just the ADP container
  + Question: Should we require CNAs to make an explicit action to change the state of a CVE, or should we infer state changes from the content of the JSON? The inference could result in people accidentally making a CVE public before they were ready, while performing an explicit action requires them to do more
    - Input from participants: A consistency or validation check alerting the user to the fact that they are instituting a state change would be helpful. It’s a middle ground between an explicit action, like a button click, and simply inferring the state change from the JSON.
* Matt: Sprint update
  + Implementing GraphQL
  + Working on a CICD pipeline and setting up some tests
  + The Sprint was impacted by a Holiday and some developers who took time off
* Joe: We need a way to publish registries of valid tags and their associated definitions and guidelines for use. How do we keep that in synch with the schemas?
  + Three options
    - Define current tag values as strict options
      * Group felt this would be the best starting point
    - Don’t define it as a pick list so it’s an open field [it could become a dumping ground]
      * Audience didn’t favor this option
    - Perhaps a tag registry where we require tags to be registered before they can be used
      * This seemed like something we might grow into, but seems overly complex as a starting point
  + Additional info on the meta data tags can be found here: <https://partners.mitre.org/sites/CVE_CNA/QWG/Documents/Reference%20Tags.docx?d=w3e74ed45bf044b5f8ad2e8715eb8bc62>
  + This topic has been discussed in the QWG as well
* Joe: email regarding 5.0 schema
* As I was working on the upconverter I found “extra” data and a gap in the 5.0 schema.
* Extra data:
* Example CVE-2019-1547
* It contains an impact property. Impact is not part of the 4.0 schema, but it’s been added in 5.0.
* Looking for how we should handle this.
* Drop extra data (data loss, pruning)
* Attempt upconvert
* Something else?
  + - Preference was to keep the extra data and put it in a dumping ground.
* Gap:
  + Impacts property in 5.0 didn’t include a language property and specified the description as a single string instead of a descriptions array (to capture translations)

060220

* Links
  + Entry Submission and Upload Service (ESUS) API <https://github.com/CVEProject/cve-services-doc/blob/master/api-endpoints.md>
  + JSON 5.0 Schema <https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss>
  + Demo of the ESUS service (still in progress)  
     <https://www.youtube.com/watch?v=N_P3AwbCLH4&feature=youtu.be>
* Joe: upconverter update
  + Working on it now, will send the link to the AWG email list when done
* Matt
  + Sprint ends next week
* Other relevant events
  + Changes to the JSON are being discussed in the QWG. Joe and Lew attend the QWG meetings and will try to provide a heads-up if the window to integrate those changes is closing.

052620

* Joe: status of up-convertor, pushed to next week
  + Previously there was a discussion of changing the JSPON 5.0 schema so that it used a UUID instead of an email. Joe didn’t have time to implement that, but Dave W offered to draft the changes and provide them to Joe (thanks, Dave!).
* Matt
  + Test suites, graphQL on the GET endpoint
    - Cristina added additional tests for the code. We’re working on standing up a CICD pipeline so we can start maturing the process.
  + Summary of condensing ID/CVE state
    - Met with Tod Beardsley’s group to explain that we’re going to represent allocated, populated and rejected in the code. There are additional states that individual organizations might want to use, but that can be done independently of the internal representation
  + Summary of ESUS MVP meeting
    - We discussed letting ADPs and CNAs modify only their respective containers.
    - It’s better for us to avoid dealing with “scope” in the initial deployment of the services, but as the notion evolves we do plan to incorporate scope into the User Registry.
    - The current JSON schema 5.0 format supports adding additional fields, but there are some concerns that this could invite abuse. We should probably constrain additional fields to just the container they are able to modify. We should also put some limits on the size of the data that gets posted there.
* Lisa O asked about getting an early look at the ESUS service so that her developers could start planning
  + Matt pointed out that we have a demo video for ESUS that walks through the set up and usage. We will try and make that available on the youtube channel hosting the training videos. When it is available, Lew will send out a notice and the link to the email list.

051820

* Lew: we are exploring CVE states, Matt and Joe are meeting with Tod Beardsley’s group this week to solicit input from the CNA community
  + We only need to track allocated, populated and rejected in the AWG services. Other states might be useful for CNAs, but tracking them in the AWG services doesn’t appear to be necessary for those use cases.
  + We might put a limit on states changes in order to prevent abuse or malicious behavior. The Secretariat will be able to override such limits. Once an allocated CVE ID is rejected, the rejection is permanent, but a populated ID that is rejected can be undone. There’s a need for more discussion about which roles should be able to perform these operations. We could start by giving this permission to the Secretariat and if it turns out that other roles need it, make those changes later. Does anyone know of a better way to put some safety measures in place?
* Joe: upconverter 5/15 slipping to 5/30 but this will not have any additional impacts to the schedule
* Matt
  + Sprint status (current sprint ends on Memorial Day)
    - Building tests for the AWG services
  + We should address versioning in the Entry Submission and Upload Service (ESUS), so that the community can see the change history of a CVE Entry over it lifetime
  + To simplify compliance with GDPR, we will require assigned\_by and Requested\_by to be UUIDs in the User Registry

051220

* Joe upconverter 5/15 slipping to 5/30
* Lew: indicate that we are soliciting addition input from SPWG for the topics below:
  + Matt: Questions for the community from last week (review to ensure no issues)
    - For ID Allocation (IDA), if someone asks for more IDs than they are permitted to receive, do we reject the request or return the partial block of IDs for which they are approved? We planned to reject the request and provide an error message that lists the maximum number of IDs that can be requested, but if that’s problematic, please let us know so we can discuss it and reach a consensus
      * *[From* Paulson*] Here's a feature request for IDA.  For Juniper, we generally ask for a block once a year.  But, if we were to exceed that block, we’d need more, so asking for more seems fair to me. Perhaps instead, IDA can have a rate limiter (e.g. 10% of your base permitted) that will first kick off 10% more – if you get 100 in approval, then you can auto-approve 10 more, which then ticks up the counter for 110 in the future.  Then, if they ask again in the same year, it needs human-intervention by MITRE to increase the approval to another number, e.g. 125, and that resets the auto-approve ticker. This would give some leeway and accommodate future growth automatically without human intervention in most cases.* This is likely not an issue unless the original request was for the entire block to which they are entitled (i.e. they requested all of the IDs to which they are entitled and did not populate any of them)
    - How should the IDA handle requests for an ID assigned to the previous year?
      * IDA would only support allocating IDs against the current calendar year. Would this be a problem for anyone?
      * Information on how the year is used in a CVE ID is [here](https://cve.mitre.org/about/faqs.html#year_portion_of_cve_id)
    - Should IDA allow people to request IDs for the next calendar year?
      * *[From* Matt Paulson*] Emphatic yes for Juniper, and we would definitely hope that it is part of the initial roll-out. We ask for next calendar year blocks before end of year*

User Registry Root CNA Requirements from Chandan

**Rules and process for transferring CVE IDs between CNAs**:

1. CNA which has been allocated the CVE IDs and the recipient CNA must both agree for the transfer or CVE IDs and associated responsibilities. (This mutual agreement is necessary. No one should be able to hijack other CNA's CVE IDs nor be able to dump them on another).

2. Both the CNAs shall submit a request independently to their parent CNA or the program root CNA, listing the specific CVEs being transferred. (Proactive steps from both CNAs provides a simple means to prove the mutual agreement.)

3. The parent CNA or the program root CNA reviews the requests, makes necessary changes to allocation records, and confirms the completion of the transfer process to the requesting CNAs or responds to the request with reasons if the transfer cannot be completed.

We also discussed what happens when a CNA ceases to exist (business closure, loses CNA status), in such cases CVE IDs should be transferred to the parent CNA or the root CNA as part of cleanup administration.

What this translates to ESUS is that :

Need API for a CNA to

a) initiate transfer of it's CVE ID to another CNA.

b) receiving CNA to list and accept CVE IDs transferred to it.

c) the program root CNA or a parent CNA to change the CNA on record for an allocated ID.

We would like to introduce this functionality after the initial deployment. There will be a way to do it (i.e. MITRE can change any values as the RootCNA/CNA of last resort, Secretariat) but adding specific functionality just for this use case should wait.

This seems like it could be easier to accomplish for a Root CNA (or Secretariat?) to do this in a single step.

Open Discussion

It was requested that the AWG agenda be sent out in advance of the meeting. Lew will send it out the day before (every Monday).

The assigner field is an email address, would it be better to indicate the CNA using a short name and UUID? This allows the identifier to persist, even if email address changes.

There’s a need to identify the “assigner” when a CVE is created. Should we use the provide\_data\_meta field with a UUID and organization short name so we can specify both who submitted it and who maintains it. Would that be an improvement? [No objections were raised]

Discussion of the use of email addresses as identifiers. The use of a UUID won’t become obsolete as an email does, but a UUID is not human readable like a UUID is. The general consensus is that we should use a UUID, possible in conjunction with a companies short name, and discontinue the use of email addresses as an identifier. Are there any objections?

We want to improve the search capability and GET requests for the CVE list. We talked about an endpoint that would support various queries, but there seems to be a defacto standard around the use of GraphQL with an endpoint to support ad hoc quieries. Graphql.org provides additional information. Does any have issues with the use of GraphQL?

050520

* For ID Allocation (IDA), if someone asks for more IDs than they are permitted to receive, do we reject the request or return the partial block of IDs for which they are approved?
* Matt: IDA MVP
  + ## IDA
  + Check header authentication fields against User Registry to confirm credentials (API secret).
  + Only allocate first 20k (arbitrary number) IDs to requests for a single CVE ID.
  + > This is to incentivize requesting IDs on demand.
  + > Or a batch of non-sequential IDs.
  + Support the ability to request a batch of IDs, sequential and non-sequential.
  + Store allocated CVE IDs in a collection and recording their Assigned status.
  + > Make sure these states adhere to current CNA rules document. -> correct names for states.
  + > Make sure CVE ID JSON is in specified format. -> described in 5.0
  + Check with User Registry to make sure request would not exceed CNA's ID quota.
  + (allowed = hard\_limit - reserved\_not\_populated)
  + Someone with the Root CNA role can allocate a CVE ID on behalf of their sub-CNAs.
  + > Sub-CNAs ID quota still applies.
  + Allocator will need to have a stubbed CVE entry created with a Reserved state.
  + ### Some Risk but necessary
  + Allow CNAs to request IDs for past year and future year.
  + > Allow the IDA to have configuration variables to keep within a designated range of past IDs
  + > Past year allocations still go against id quota.
  + > IDA will need record of past ID allocations in order to be able to do this. This will require a data migration from CPS to services.
  + ## ID Library
  + Can perform a GET on a CVE ID to get its status, but anonymized.
  + CNA can update the status for a CVE ID. -> updates CVE Entry to be rejected as well.
  + CNA will be able to check its IDs that have been allocated.
* How should the IDA handle requests for an ID assigned to the previous year?
  + IDA would only support allocating IDs against the current calendar year. Would this be a problem for anyone?
  + Information on how the year is used in a CVE ID is [here](https://cve.mitre.org/about/faqs.html#year_portion_of_cve_id)
* Should IDA allow people to request IDs for the next calendar year?
  + It is technically feasible but requires more work, so we want to make sure there’s actually a need for this before we invest the development time in it. It could also be added after the initial rollout
* We will need to migrate data from the Content Production System to the new IDA service so that we can enforce the quotas currently in effect.
* We should list the metrics and their status on the github site

042820

* Lew: Authentication/User Registry schedule
* Joe: nothing at this time, but here to answer questions. In a few more meetings Joe will have another update.
* Matt: Summary of Matt’s User Registry doc. What we’re thinking now and what questions we need answered for the user registry.
  + CNAs can be entered
  + CNAs can add additional users to their org
  + ID quotas
  + Can we get use cases for the various use cases? Standard user and Super User (create accounts within org, etc.)
  + Roots and Secretariat roles are being stubbed out but we’ve made it possible to add those later as additional requirements are created.
* Matt: schedule for ID Allocation
* Matt: quick sprint review (for people who don’t follow SPWG): prerelease branch so people can review the current status of the code. We have a simple authentication mechanism as a backup in case the User Registry is delayed. OpenAPI YAML file is expected by 4/30. Ensuring that JSON validator returns concise error messages. We’re planning to release an in-progress version of the code that includes the mongoDB (originally this was to occur later in the schedule).
* Matt: Sprint issues in next sprint:
* Matt will post current status to Github and Lew will include in weekly meeting notes
* Need for additional error codes in the ESUS service [200 and 500, but need more, will try to get to this in today’s meeting, but it could get pushed]
* Issue: storing non-official fields could get out of control if we don’t have limits on it. Size limits, content, format, this is also a security vulnerability with potential for exploitation

041320

* Proposed that we move the AWG meeting from every other Monday 2:00-3:00 EST, to weekly on Tuesday’s 4:00 – 5:00 EST. No one on the call had an issue with the proposed date/time change, but please let me know if this change is problematic.
* TBD: bootstrapping the user registry with initial data for ESUS testing. We have thoughts but need to flesh them out, Matt can give a preview today.
* We have a schedule for ID Allocation and Authentication/User Registry but will present those later due to time constraints
* Joe provide JSON 5.0 transition: Goal is to provide more information in a more structured way to support automation
  + Products to support the transition
    - JSON validators
    - An Up-converter from JSON 4.0 -> 5.0, available on the AWG site
    - A down-converter from JSON 5.0 -> 4.0
  + 3/30/20 Post JSON [python and possibly NodeJS] schema validator(s) to AWG Github site and updated over time
  + 5/15/20 Convertor from 4.0 to 5.0 posted to Github and maintained
  + 6/19/90 Convert Github corpus from 4.0 to 5.0 and keep both formats on Github
  + 9/30/20 End of 90-day transition period, stop supporting JSON 4.0 entries [pushed from Aug to Sept.]
  + Need to determine how long to support 4.0 past the 90-day transition and identify the best way to do this. At end of 90-day window, put github in read-only mode and provide 4.0 for at least six months [6-month clock begins when we announce the start of the transition period. This needs to be advertised to the community. Add to website, CVE form, CVE twitter feed, others?]
  + Over time, we will update the schema validators to include additional checks for state transitions, checking permissions to change specific containers and fields
* Matt provide sprint status
  + Implemented JSON Validation code and User Verification code
  + General housekeeping
  + Sprint review in SPWG on the 13th or 20th depending on Kent’s agenda
  + Planning session for next sprint
* Matt Provide ESUS transition: Goal is to provide an Entry Submission and Upload Service to remove a bottleneck and enable automation and self-service; User registry and Authentication Services; and an ID allocation Service supporting non-sequential IDs and sequential ID blocks [IDA roadmap presented at a future meeting due to time constraints]
  + Products to support the transition
    - Stand-alone version with DB stubbed out
    - Online version with authentication/user registry and dummy data
  + 4/3/20 Implement JSON Validation code
  + 4/3/20 Implement User Verification code
  + 4/30/20 Provide ESUS code with DB stubbed out for CNA beta testing
    - [demo to SPWG and provide a webinar for CNAs and stakeholders]
  + Matt will post current status to Github and Lew will include in weekly meeting notes
  + Need for additional error codes in the ESUS service [200 and 500, but need more]
  + [add tasks to the posted sprint schedule, Matt will get this info on the board]
  + 4/30/20 Integration with CPS
  + 4/30/20 Complete Open API implementation [sooner is better]
  + 5/30/20 Cognito and User Registry configured and available [separate team working this]
  + 6/20/20 Add Authentication for internal ESUS Beta testing [waiting for ESR]
  + 6/30/20 External System Request permission to host AWG code on a public AWS instance
  + 6/30/20 Provide code in AWS instance with mongoDB and representative data for beta tests [Question for SPWG: does the 90-day transition period need to end by 9/30, or is it sufficient to have the services available by September? Answer: Not necessarily, but the services do have to be available and fully functional
  + 7/1/20 Deploy ESUS initial version connected to CPS for early adopters
  + 9/30/20 90-day transition ends, turn off Github submission service and stop accepting JSON 4.0
  + Q: start the 90-day transition clock when ESUS is in AWS with Cognito and real data [currently expected on 6/30/20]? A: yes
  + Q: How will initial provisioning of a CNA get handled? A: we will have internal code to allow the addition of permissions in the user registry. The MITRE team can add folks until we get a self-service module that lets CNAs add their own permissions
  + Q: would it be helpful to have webinar tutorials for ESUS to assist CNAs in making the transition? A: yes

033020

* JSON 5.0 status: what’s in and what got punted to the QWG
  + Email from Alex:
    - It looks like repository + references would accomplish the same thing. We're eager to have fields that are more tightly defined than references, primarily because both we and Google are interested in doing research and validation at scale that's harder to do if a human has to look at each CVE that comes in first to separate out the references into the repository, the fix revision, etc. That said, I'm newer to the CVE format and the general way of doing things so it may be that I'm missing some way that the references field could provide the same tight definition.
* JSON. 5.0 transition plan
  + we’ll do 1 time conversion to bootstrap a new github repo/branch/directory so there is a 4.0 and 5.0 set
  + CPS can push updates to both 4 and 5 for a time while consumers migrate
  + We turn off/disable github submission
  + the upload and submission service becomes the new process, it feeds to CPS, CPS records and then pushes to github.
  + to discuss at some point, what frequency do we really need to refresh the CSV and XML CVE catalog dumps (~200mb), these can also be pushed to github instead of going through tarball to get onto the current cve.mitre.org site.
  + Note – the 4.0 schema contains less content that the 5.0 schema supports so 4.0 loses data, but it can be generated from 5.0 content, and 4.0 contains enough to produce a valid 5.0 CVE (but not all the optional content).
  + 90-day transition period?
  + Q: After the transition to JSON 5.0, can we put GitHub in a read-only state so people can pull the cve corpus from it? Dave W mentioned that NVD needs to do that. I thought we also had a new endpoint that let folks download the corpus, is that correct? If so, could we transition to that service and retire github? Do we do that as part of the JSON 5.0 release, or stagger them?
* ESUS status and sprint schedule
  + We need a date for the first sprint review so Kent can schedule it in the spwg. I owe Kent as response.
* ESUS diagrams
  + [AWG Diagrams/AWG Upload and Submission Service Flow Diagram.png](https://mitre-my.sharepoint.com/personal/lloren_mitre_org/Documents/Microsoft%20Teams%20Chat%20Files/AWG%20Diagrams/AWG%20Upload%20and%20Submission%20Service%20Flow%20Diagram.png)
  + [AWG Diagrams/WS-upload-sequence.png](https://mitre-my.sharepoint.com/personal/lloren_mitre_org/Documents/Microsoft%20Teams%20Chat%20Files/AWG%20Diagrams/WS-upload-sequence.png)

031620

* Divide changes into easy ones that will be included in 5.0 and those requiring additional conversation, which go in 5.1. The problematic case, which I do not think we face, is one where a change requires additional discussion, but would break backwards compatibility or cause significant re-work.
* Update from Matt on sprint cycle
  + Try to get a cycle going where requirements one or two sprints out are reviewed by QWG and SPWG
* We need a transition plan for the JSON 5.0 schema

Notes: Joe posted 5.08 last week. CVE metadata block missed date/time field being optional, that’s been corrected.

Chandan posted updates. Unless objections, let’s roll in the definitions for configuration, work around and exploits. These were just a blob, Chandan changed those to arrays.

Timestamp/date/email addresses have proposed changes.

Validators have not made the conversion to a draft 7 validator. We could define date/time/email and provide a pattern, but the challenge is establishing a REGEX patter that is identical to the schema type (email definition is the challenge). Two approaches to validating opens the possibility that REGEX and field definitions produce different results.

Pushed to 5.1

Metadata tags. How are they applied and where in the schema should be they be? Metadata? ADP/CNA? If main body of container is affected, … As long as metadata tags are not required, it won’t choke the validation

NVD reference tags -> 5.0

CVE tags for QWG -> 5.1

021720

* Vote on JSON schema v 5.0 <https://github.com/CVEProject/automation-working-group/blob/master/cve_json_schema/v5.x_discuss/cve505_containerized.schema>
* Documentation on AWG API is here: <https://github.com/CVEProject/cve-services-doc/blob/master/api-endpoints.md>
  + An OpenAPI version of this will be posted later
* We are currently applying for permission to create a separate AWS instance for the AWG services and new website. Estimated time is ~8 weeks. This could be complicated by 4300A requirements

020320

* Asked Matt to explore the use of Open API for describing the AWG API: He was already planning to do that. Matt put textual description of the API on the AWG site for now
* Who else should get permissions for the AWG Github site? Chandan? Tod B.?
* JSON schema: <https://github.com/CVEProject/automation-working-group/tree/master/cve_json_schema/v5.x_discuss>

Comments from Madison and responses from Joe:

All,

CERT/CC reviewed the schemas and we wanted to offer our opinions and feedback. As discussed in yesterday’s meeting, we’re in favor of the containerized version over the consolidated version.

Overall feedback:

There are a number of places “to be decided” has been left in the schema. Will these TBDs remain in the final publication? Will they be removed before sending to MITRE for approval (which I believe is the next step after the AWG approves)? Are there specific dates or other input that we’re waiting for to make these decisions?   
  
These are pulled forward from the current v4 schema to prevent data loss going forward. These items are currently defined as optional JSON objects. (Mandatory in: none, Optional in: all containers, JSON data type: object ) Structured definitions for these TBD object do not exist, and are rarely used in practiced.

* More of an FYSA for the group until decisions elsewhere have been made – The Quality Working Group is discussing the implementation of two tags in a CVE, end-of-life and hosted-service. Art brought this up in the Quality Working Group mailing list as they’re the group currently working on CVE tagging so some of you may have seen the it. The discussion about the tag definitions and uses are happening currently which will (likely) lead to requirements for the AWG once those things have been decided. We’re suggesting that the tags go with the product definitions (with CPE and SWID) and looks like it would be simple enough to implement into the existing schema, so hopefully won’t lead to time-consuming changes.  
    
  I don’t have the background on this one, but this lends itself to being implemented as an optional array of string values in the meta-data, or the CNA container, with a set of enumerated strings if these are going to be true/false tags. If the tags have other values this would an object with propertied names being the tag names, and the values being the values. If this concept is solidified, and a location is chosen for placement, it can be inserted into the v5 schema.
* The proposed schema allows for non-English CVEs to be generated but will still default to English. Something to consider that may affect the schema (but I believe it currently supports) is that English CVEs will still be required even if a non-English CVE is generated (as per the new rule changes).

It allows for non-English content, but the schema, numerous fields, and the enumerated lists are assuming English values only. Schema and enumerated strings need to be a single language. Items like the CVE\_data\_meta\_public.TITLE(string) could be converted to arrays to support multiple languages. I am not familiar with any new rule changes in reference to language.

Specific feedback:

* Line 27, the email address pattern – is there a particular reason that the plus symbol “+” was excluded? We vote for it being included unless there are reasons to leave it out.

The ‘+’ is a tag prefix, and creates an opening for exploit and script injection. <https://www.cs.rutgers.edu/~watrous/plus-signs-in-email-addresses.html> This is great feedback and makes changes (assuming approved) easy to do. I don’t have any technical reason to reject this item as long as the resulting regex validates valid email address against the email address standard.

* Line 199, 203 – misspelling of registry

Thank you - updated

* Line 357 – this description looks like it was copied from elsewhere. This might’ve been deliberate but is worth reviewing.

Correct – loosely defined string field, v5 makes it support multi-language.

* Line 393  and 396 – we should specify how CWE should be entered. If it’s CWE-123, it looks like both “CWE-123” and “123” would technically be correct in this field. Earlier in the schema, we specify that the text “CVE” is required for CVE IDs, so we’d also vote that “CWE” should be required for CWEs in this section.

It’s an overloaded field. Not all problem types are covered by a CWE, so the description is left as generic string. The CWE-ID property was also left as a generic string, because the schema of CWE-ID are managed externally to the CVE program. Leaving this as a generic strings keeps the CVE schema decoupled from the CWE schema.

* Line 418 – we vote for supporting and suggesting base64 over uuencode unless there is reason to specify uuencode.

Uuencode is what is currently supported in v4, and was pulled forward into v5. Recommend reject use of base64, as it obscures the URL and breaks the ability for it being human readable. (example: <https://codebeautify.org/base64-to-json-converter> => aHR0cHM6Ly9jb2RlYmVhdXRpZnkub3JnL2Jhc2U2NC10by1qc29uLWNvbnZlcnRlcg==)   
Value encoding should result in human readable JSON safe strings.

Please feel free to reach back out to us if there’s any questions on our feedback. We’re very happy with all of the work that’s being done on this and are happy to help where we can.

012420

* Review of where we are: AWG was nearing agreement on the 5.0 schema. Some additional functionality was introduced, along with a slightly different approach to implementing it which creates a choice of implementations. Use containers to adp, cna, CVE-meta data, our approach creates and array structure for the new metrics block and our approach didn’t separate can/adp into distinct sections. Dave’s latest wraps the whole cve in either an ADP container or [separates CNA as one container, ADP container is a collection of contributions]
* Dave’s latest revision is recommended. 5.03 version into container structure. There’s a little clean-up work to do, but the structure looks correct. Note: this version will require changes to the minimum requirements. Up until now, we haven’t had a single consistent schema, so changes were highly likely.
* This is the first time we’ve had to make a major decision that didn’t have unanimous support, so I’d like to use this opportunity to try out some new procedures, like voting. I suggest we take a vote (voice for those in attendance, email for the next week? Two weeks?). We can certainly modify the procedures if people have ideas to improve it.

010620

* Review of JSON schema changes to CVSS score
* Dave W. suggested additional changes to the JSON schema and will send out a quick mock-up in a few days followed, a week later, by a JSON schema implementing the proposed changes.
* Time permitting, Joe W. will incorporate Dave’s suggestions and send out a JSON schema for review by the AWG.
* We will review the JSON changes at our next bi-weekly meeting, and hopefully we can agree on a final version at that time.

112519

* Joe B asked that we add a “planned disclosure date” to the JSON schema to keep their users more informed about CVE status.“ Does that cause any heartburn?
* Suggestion from Matt P: when NIST needs extra data, can they ask CNAs to provide it at a "Well-Known URI" as defined by <https://www.iana.org/assignments/well-known-uris/well-known-uris.xhtml> ?

111019

* Lew and Matt update group on how we will manage AWG privileges for Github (we want to strike a balance between giving everyone access and only allowing it for MITRE staff)
* Joe W and Chris T (and Chandan?) propose JSON changes so we can come to a consensus and finalize the schema
* We are looking to replace the Github process with a new submission service. Who is pulling info from Github and what replacement functionality will minimize the impact to the CNAs?
* The Board asked that we use [the](mailto:cve-board-auto-list@mitre.org) CVE Board email list to discuss issues via email and to provide a written summary of the discussions during the meetings, so that folks unable to attend the meeting remain informed.

102819

* Joe W did not see any issues in implementing Chandan’s CVSS changes into the JSON schema.
* Chris T. had questions about the Chandan’s initial cut, where does that stand?
* Do we need to provide notice to the community before implementing?
* Unknown dev asked about contributing to the code. Do we have a process for this? Do we need anything from them for verification etc.?
  + How are we checking code? Manual review by AWG members for acceptance? Static code analysis?
  + When we have a development process in place and code analysis processes, do we want to offer bug bounties?

101419

* Let folks know we’re always interested in feature, bug, or task requests. I don’t want the availability of MITRE devs to delay our work. If anyone has time to contribute, we’ll take all the help we can get
* Chandan or Joe W. should present the proposed changes to the JSON schema to include an array of the CVSS scores by version [and there was another change, maybe the effects field?]
* Someone [I didn’t note who] asked about what the MITRE analysts do, and I floated the idea of having an analyst present their workflow. Jonathan said he could work something out if folks are interested. Who asked about that and is that something that others would be interested in (i.e. is there enough interest from the community to justify the effort on Jonathan’s part to do this?)

093019

* Update on JSON CVSS update incorporating the first.org JSON treatment of CVSS
* Review Joe B’s requirement requests from last time, if he’s present
* Chandan planned to send me an email re the 270 JSON entries that got mangled during an upload. Nick thought he fixed 250 with a script, but I’d like to verify that status.

091619

Chris T asked about finalizing the CVSS Json schema, we should discuss in today’s biweekly

<https://github.com/CVEProject/automation-working-group/blob/master/cve_json_schema/DRAFT-JSON-file-format-v4.md#cvssv2>

Use the JSON that first defines here: <https://www.first.org/cvss/data-representations>

Just use the vector defined in the first.org description and the score. Use the required fields at the bottom of the first.org link, we could just that entire schema for our JSON checker.

Include support for v2, v3, v3.1 for CVSS version, at least one of these must be provided. Chandan suggests using an array for this info to support multiple CVSS scores in the JSON. Chandan feels it is redundant to include the version field since the version number is part of the CVSS score.

Attaching a description to the score is also a nice feature. Should NVD and NIST both have a place for their scores, so we know where the score originated (this will be an optional field). If you’re not providing it, someone else, like NIST, might want to add it.

How do we change the draft schema? Joe should update the AWG draft JSON schema so it reflects our use of the first.org JSON schema.

Requirements from Joe B.

* We don't want to have to reserve and manage a block of numbers.  Instead we'd like real time reservation of CVE numbers 1 at a time and, in case we submit multiple requests rapidly, we'd like them to not be sequential.  As a matter of fact, we'd like them to not be sequential at all, i.e. don't just increment one CVE at a time CVE-2019-0001, 0002, 0003,…..   Instead take a large block and dedicate it to random assignment so we might get CVE-2019-115520 and then CVE-2019-10222 on back to back requests.  This would lessen the ability to make a time correlation.
* We'd like to be able to communicate a planned public disclosure date as we give our customers pass through language like this -

Enhancement to address [CVE-20YY-NNNN] (<https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-20YY-NNNN>) anticipated to be published [MM/DD/YYYY]

So that they can put it in their release notes and release their updates prior to Intel publishing the CVE/Public Security Advisory.  We want to avoid what has happened a couple of times in that a CVE appears in the wild in someone's release notes and it looks like we are negligent in publishing the CVE. [mention next time so Joe can explain the reasoning, some others have suggested this could go in the description field so we’re not just increasing the number of fields]

* We'd also like to handle the typical rejection, update, etc..  Basically, we will have buttons and automated circumstances to accomplish all these tasks in our Service Now implementation.
* We would like to be able to request a CVE number that would be for next years’ time period. For example, if it was November or December 2019, we would be able to request a CVE-2020-001 instead of a CVE-2019-001 because our disclosure date would not actually be until sometime in 2020.

071019 AWG summary for Board Meeting

* Source code is posted on github, also available as exe in a docker container
* We’re working on standing up a publicly available workboard for AWG
* User Registry and Authentication services will be on the back-burner as we explore Amazon Managed Services, which do provide some authentication and permissions
* Upload Service – looking at vulnogram as a starting point. Vulnogram currently works with Github, we want to see if we can leave the fn intact and only modify the API calls to ensure that our API will provide all the functionalityn required to replace Github

070819

* AWS migration and amazon web services for Authentication and possibly user registry?
* AWG status from Matt
* Solicit requirements for Github replacement

062419

* Request status from Anthony Singleton on the corrupted JSON files from March
* Status update from Joe W. on JSON changes
  + Chandan suggested we also release a translator so CNAs can re-format JSON files that have been created but not submitted
* Let Chandan know looking at Vulnogram to provide a replacement for Github in in the queue (we’re going to develop some new endpoints and potentially modify Vulnogram to work with the new endpoints)
* Solicit requirements for Github replacement

061019

* Time to start ramping up the Entry Submission and Upload Service as a replacement for Github. What functionality do folks want?
  + Github let’s any random person see the flow and timeline for the process, so they get an understanding of what needs to be done and how long it takes
  + There’s a benefit of capturing pull requests from a CNA because it serves as a public record of input from the CNA (the pull request itself is a public record)
* Use new web site as the UI front end for services?
* Docker for AWG services from Nestor
* JSON change suggested from Joe W. to remove bloat

052019

* Do CNAs want the ability to change the “Assigner Value” in a CVE entry? It’s the contact email for a given CVE. Do CNAs need to update the contact info for all CVE entries (including the historical ones), only a subset of CVE’s, etc.? Ticket 1037

051319

* Schmitty planned to provide a walkthrough of the offsite
* during the meeting on April 29th, there were questions about whether Github would continue to be used and the role of the AWG; I agreed to look into the issue and respond with my findings. Currently, The SPWG is working on a position paper exploring the pros and cons of continuing to use Github in light of the EU’s General Data Protection Regulation (GDPR) and other factors. Once complete, that paper will be provided to the Board so that they can make a decision. Once the decision is made, the AWG will need to factor that decision into the “Entry Submission Service” which has yet to begin.
* Lew should have checked Chandan’s pull request with the changes to the JSON comments
* Lew: There was a MITRE commit a month ago that appears to have over-written fields from CAN submissions, replacing the content with “not available.” Joe Sain was looking into this.
* Lew: Has the decision made to replace Github been made, and if so when? Why wasn’t the AWG informed of this?

042919

* Code repos are now available at the CVE.mitre.org website under board/CVE Working Groups
* I requested that Chandan’s changes to the JSON description be approved: <https://github.com/CVEProject/automation-working-group/pull/76/files>
* Chandan pointed out that he sub mitted a JSON validation request that failed, but when he ran the online tool, it did not return the error. We need to ensure that the online tool uses the same code as the internal error checking. I suspect that at one time they were two identical code blocks that diverged over time. If at all possible, both the online tool and the internal JSON validation should use the exact same code (i.e. a singleton rather than two distinct, but identical, blocks of code).
  + Joe S: This sounds like it could be an AWG or QWG project designed to develop a standard JSON format.  Perhaps Chandan could lead it?
  + Anthony - This is something we will need to write up and share with the Community. We have several checks running in the CPS that are not transparent to the community. The issue Chandan mentioned is the use of multiple same language tags. These checks are not present in the documentation for the JSON Schema.

040119

* Schmitty planned to provide a walkthrough of the offsite
* I requested that Chanden’s changes to the JSON description be approved: <https://github.com/CVEProject/automation-working-group/pull/76/files>
* No new progress to report on the AWG development work, developers were attending to other tasks., which works out OK because we need to identify additional requirements for User Registry and Authentication

031819

* Record future conversations
* Invite Nestor and Matt so they can demo their work
* Schmitty planned to provide a walkthrough of the offsite
* Exchange from Chanden re JSON submission and CVSS scores
  + From Matt P: The Juniper CNA is complaining that the Red Hat CNA should use a different representation for some of the non-required information in the JSON file format. This is completely independent of the transition from the CMS to the CPS. Almost certainly, MITRE does not need to make any immediate code changes to block what Red Hat is doing. My feeling is that both Juniper and Red Hat have valid positions, and the concern could be resolved elsewhere (e.g., in the Automation Working Group) at a later date.
  + However the scores in IBM's JSON submissions are not consistent with official CVSS JSON format recommended by CVSS SIG (same as used in NVD JSON feed): <https://www.first.org/cvss/data-representations>
* Chandan Pull request status
  + It was discussed in Oct 15th automation working group meeting. My proposal to add "?" as character to denote unknown affected status was accepted unanimously. I was asked to submit a pull request to the documentation so documentation can be updated. This did not require a schema change (as schema allows any characters). I submitted the pull request the same day. It has been six months and still pending without any further progress.
    - The code change has not been accepted for two reasons:
    - 1. We are not in the review cycle for making changes to the JSON schema
    - 2. There must be agreement from the community to accept not just MITRE saying ok.
    - This could be a good opportunity for either a working group to be created to engage the community to further standardize the JSON schema or some sort of email chain to be created bringing the Schema back to the foreground.