FedRAMP OSCAL Early Adopters Workgroup Knowledge Capture

September 27, 2023

Meeting Details

Purpose: Weekly working session for the participants of FedRAMP OSCAL Automation Early Adopters work group (by invitation only).

Outcomes

- Shared understanding of Charter and Mission of the Workgroup
- Shared understanding of SAR Schematron Rules and rule levels.

Slide Deck

Meeting Notes

- Introduction
  - Tom Sr. reviewed purpose and outcomes for the workgroup, guiding principles, workgroup's charter/mission, provided some OEAW updates, current issues on the early adopters github
- Aaron discussed/demoed validation updates
- Phase 2: General Artifact Submission – Continued
  - Reviewed basic ATO package requirements & current challenges & compressed package submission examples

Participant Questions/Conversation:

- Valinder: Is FedRAMP 2.0.0 implementation and guidebooks being updated for OSCAL 1.1.1?
  - Tom: YES, we are working with Rene on this. Ahead of the game on the portal. Minor updates, but some side bar in depth issues he'll have to update with regards to that.

- Matthew Coughlin: Have you published the strucutre for compressed files you are recommending? And that's the legacy structure?
    - Tom: We have not. And that is correct.
    - Dave Waltermire: I think that's an ideal approach. There's a lot of peril about being prescriptive with folder naming because it's difficult to enforce. Case issues, misspellings, spaces, etc. We have to think through this so we can avoid those problems.
    - Tom: I think your idea about an open source tool to assemble the package is a great idea. It can interact with the rest API the way we need them.
    - Dmitri: You can build those structures in and they wouldn't have to worry about them too.
    - Tom: My suspicion is that CSPs have a repo that they push their stuff to and that's why the structure is the way it is now.
    - Greg: Have you thought about running a repo? And just having all this stuff in repos?
    - Tom: In other words, FR hosts the repo?
    - Greg: Yes and people commit individual files to a repo
    - Tom: I wish we could do that easily. There are challenges. This POC for the portal and Rest API, FR is trying to get hosting for this and a GRC tool, and basically we have tried to build these black boxes with the REST API. Our team is plugging these holes right now so we have a viable way to move forward once we have a GRC tool and repo a year from now.
    - Dave: There's a lot of technical challenges around if a repo was a path to take. How to manage access, how do you manage updates, etc.
    - Greg: I'm not suggesting there is one giant repo. Every SSP could have its own repo. And all the files associated with that SSP could be in that repo. There's your file structure, XML files, and it's all in a repo. If that repo, if 300-400 repos were hosted in Gitlab or Github at FR, essentially every CSP would have access to each repo for each of their services. The appropriate parties at the PMO would have access to them as well and could be put in GRCs or whatever but the file transmission process would be a commitment to the repo..
    - Dave: Something to think about. Thank you.
    - Tom: OSCAL brings a different dimension to that. It allows us to put that bouncer at the door, reducing back and forth between CSPs and review team.
    - Greg: The rules validation is incredibly interesting and powerful. It is a huge checklist. Forces everyone to up their game in terms of content and saves weeks and months in time.
    - Dave: The feedback can be synchronous. In a repo, it would have to be asynchronous. I think synchronous is better personally.

        - Greg: We get synchronous feedback from our repo all the time. Happy to talk about it another time.
        - Dave: Is there a commit hook?
        - Greg: CICD pipeline.
        - Dave: It would be asynchronous in that case? We can talk about it later.
    - Valinder: Our suggestions on the package submission. We get nervous when we have to zip everything up and move it over the wire. We would like the concept of initiating a package, uploading individual artifacts. While uploading we can get feedback, great. Submitting smaller validated chunks at a time is better. And then closing out the package and submitting as a final step would be great.
        - Tom: That's where we started with this portal.
        - Dave: CSPs are going to have different levels of maturity. For some, a rest API will be easier. If we provide rest APIs, it's possible to support both.
        - Bill: I think we need to support both. Valinder's point is great to submit in chunks. But some people want to submit everything at once.
        - Valinder: I agree with you. The only challenge with a large package is the size but also the synchronous feedback. It's bound to become asynchronous. Going to be challenging to run with synchronous feedback. Decomposition and running the validations will be easier.

Chat copy:

(N/A)

Actions

Owner	Action Item
	□
	□
	□
	□