FedRAMP

Developer Data Bites

March 16, 2023

GSA

info@fedramp.gov
fedramp.gov

Introduction

Purpose: To create space for dialogue between developers who use OSCAL and the FedRAMP® automation team.

Outcomes:

- Shared understanding of how to reference leveraged authorizations in OSCAL
- Productive discussion around OSCAL for FedRAMP

Agenda:

- Welcome
- Pre-Submitted Q&A
- Referencing Leveraged Authorizations in OSCAL
- Open Forum
- Next Steps & Closing

Data Bites Guiding Principles

Keep the discussion respectful

Be curious, seek understanding

Speak from your own experience

Challenge through questions

Focus on ideas

Keep it technical

Pre-Submitted Questions

Pre-Submitted Questions

Question 1: When will OSCAL-generated documents be required to be submitted for ATO?

A CSP Transition Plan will be made available once Rev 5 baselines are approved and released. Timelines will be determined based on your annual assessment date.

FedRAMP Automation GitHub Updates

Update on FedRAMP automation repository

10x Flexion Transition

- 10x/Flexion team has transitioned the majority of Schematron validations support but is still available for consultation to FedRAMP team for limited period of time.
- A new resource has been added to the FedRAMP validations team: Dimitri Zhurkin

Issues/Ticket Tags

- **Newly opened issues this cycle**
  - #386 Prop value for PIA (Privacy Impact Assessment)
  - #385 Add Rev 5 ruleset to web documentation
- **Closed issues this cycle related to Schematron Validations**
  - #387 sample-ssp.xsl has flawed Schemaron xml-model processing instruction
  - #381 Risk associated with multiple POAMS
  - #375 Reading HTML code within Multi-line Markup

Referencing Leveraged Authorizations in OSCAL

Referencing Leveraged Authorizations in OSCAL

What is a Leveraged Authorization?

A leveraged authorization exists where:

- A leveraged system passes responsibility for control satisfaction to one or more leveraging systems (Customer Responsibility); and/or
- A leveraging system inherits security control satisfaction from a separately leveraged system (Inherited Control).

Examples:

- Several SaaS systems running on a separately authorized IaaS.
- Several systems relying on a separately authorized storage array or other general support system (GSS)

** Reference: OSCAL Presentations - Leveraged Authorizations

Referencing Leveraged Authorizations in OSCAL

Control Satisfaction and Inheritance

In fully satisfying a given control:

- Some controls must be satisfied independently by each system
  - **Example**: FedRAMP does not allow policies to be inherited. Each system owner must satisfy policy requirements independently.
- Some controls are only fully satisfied if each individual system does their part.
  - **Example**: Logical access control must be implemented on all components in “the stack.”
- Some controls are fully satisfied at a lower level, thus fully inherited higher in the stack.
  - **Example**: Usually an IaaS takes care of all physical controls. Each SaaS has no ability to implement physical controls and fully inherits those controls from the IaaS.

Referencing Leveraged Authorizations in OSCAL

Controls in the SSP: Two Approaches

Component Approach (Preferred)

- Each control response is broken down to the individual components involved. Enables a more robust response to controls.
  - **Example**: The access control implementation that satisfies AC-2, part (a) is described separately for the firewall, the router, the platform, the web server, etc.

System Approach (Legacy)

- Enables initial conversion of a document-based SSP to OSCAL with minimum reorganization of control responses. Except for leveraged authorization content, each control response is tied to a single component: “This System.”
  - **Example**: A legacy SSP has a single space for AC-2, part (a), which has a free-text description the access controls within the system. This single description is associated with “This System” component in an OSCAL SSP.

Referencing Leveraged Authorizations in OSCAL

3 Scenarios

You are submitting an OSCAL SSP (leveraging system)

1. **OSCAL SSP / With Access**
   - The leveraged system is using an OSCAL SSP, and the leveraging system is permitted to access it.
2. **OSCAL SSP / No Access**
   - The leveraged system is using an OSCAL SSP; however, the leveraging system is not permitted access it.
3. **Legacy SSP**
   - A leveraged system is still using a legacy SSP. A legacy Customer Responsibility Matrix (CRM) is used.

Referencing Leveraged Authorizations in OSCAL

Scenario 1: OSCAL SSP / With Access

The leveraged system is using an OSCAL SSP, and the leveraging system is permitted to access it.

Benefits:

No Customer Responsibility Matrix (CRM) is needed!

- Tools can identify which statements in the leveraged system's SSP have a customer responsibility,
- Tools can further identify the leveraged system's components associated with these customer responsibility statements
- Automated process little room for error or misalignment of control inheritance/responsibilities.

Drawbacks:

- Leveraged SSP must be OSCAL before the leveraging system.
- If multiple leveraged systems, then all SSPs should be in OSCAL.

Recommendation(s):

- If automating with OSCAL, transition your leveraged systems first.
- Use “this-system” approach on leveraged system OSCAL SSP.

Scenario 2: OSCAL SSP / No Access

The leveraged system is using an OSCAL SSP; however, the leveraging system is not permitted access it.

Benefits:

- A CRM will be required and used by FedRAMP.
- The leveraging system's SSP must ensure they fully satisfy every customer responsibility statement in the CRM, which requires at least one entry within the cited statement.

Drawbacks:

- Potentially FedRAMP does not have access to collateral materials.
- Similar to current manual submission and review process but a little better.
- Less automation, which puts the reviewers back in a more manual review mode.
- Slows down the authorization.

Referencing Leveraged Authorizations in OSCAL

Scenario 3: Legacy SSP

The leveraged system's SSP is not expressed in OSCAL, or its CRM is not.

Benefits:

- FedRAMP can generate a Word version of the leveraging SSP from OSCAL.

Drawbacks:

- Every responsibility statement in the leveraged system's legacy SSP/CRM must be addressed by the leveraging system's SSP within the cited control statement.
- Completely manual review process.
- Slows down the authorization.

Referencing Leveraged Authorizations in OSCAL

Discussion Thoughts on Process

- If leveraged system is FedRAMP Authorized (OSCAL and Legacy), then will review team have access to all collateral materials for the process?
- (Scenario 2) What situations exists where where a leveraging system does not have access to a leveraged system OSCAL SSP?
- (Scenario 3) How should the FedRAMP PMO review teams handle this? Real World situation or not, if automating?

Open Forum

What topics do you want to see in the future?

Quick Poll

What do you want to see?

1. Demo of NIST Deepdiff tool
2. Demo of GSA Public portal prototype
3. Something else, I will submit my ideas in the chat.

Thank you

Our next Developer Data Bites virtual meeting will be on **Thursday, April 13, 2023 at 12p ET**.

Submit questions and future discussion topics to OSCAL@fedramp.gov

Learn more at fedramp.gov

@FEDRAMP

How to Submit Issues with FedRAMP

Ensuring your outstanding issues or questions are received:

Issues can be submitted in several ways:

Preferred	Alternate
Open an issue on fedramp-automation github so that it will benefit the NIST/FedRAMP community. https://github.com/GSA/fedramp-automation/issues	Email us at oscal@fedramp.gov

OSCAL Resources

NIST:

https://pages.nist.gov/OSCAL/

Learning Resources: https://pages.nist.gov/OSCAL/learn/

Current release: https://github.com/usnistgov/OSCAL/releases

Development version: https://github.com/usnistgov/OSCAL/tree/develop

Content repo: https://github.com/usnistgov/oscal-content

FedRAMP:

Current repo: https://github.com/GSA/fedramp-automation

Current issues: https://github.com/GSA/fedramp-automation/issues

Validations work: https://github.com/18F/fedramp-automation/tree/master/src/validations

Web based validation tool:

https://federalist-2372d2fd-fc94-42fe-bcc7-a8af4f664a51.app.cloud.gov/site/18f/fedramp-automation/#/documents/system-security-plan

Useful Links

3rd Party Tools

- https://www.oxygenxml.com/xml_editor/download_oxygenxml_editor.html (Oxygen XML Editor)
- https://github.com/usnistgov/OSCAL/tree/main/xml/schema (NIST xsd schemas)
- https://github.com/usnistgov/oscal-content/tree/main/nist.gov/SP800-53/rev4/xml (NIST profiles)
- https://github.com/GSA/fedramp-automation/tree/master/dist/content/rev4/baselines/xml (FedRAMP profiles)
- https://code.visualstudio.com/download (Visual Studio Code)
- https://marketplace.visualstudio.com/items?itemName=qub.qub-xml-vscode
- https://marketplace.visualstudio.com/items?itemName=redhat.vscode-xml

Rev 5 Update

Rev 5 Timeline

Estimated Release Date of XXXXXXX

- Phase 1 Release: XXXXXXX
  - Rev 5 Baselines
  - CSP Transition Plan
- Phase 2 Release: XXXXXXX
  - Templates
  - OSCAL Docs
- Phrase 3 Release: XXXXXXX
  - Everything else (+45 documents)