FedRAMP

FedRAMP OSCAL Early Adopters

March 13th, 2024

GSA

info@fedramp.gov
fedramp.gov

Introduction

Purpose: To engage Cloud Service Providers, 3PAOs, tool vendors and other participants in FedRAMP's OSCAL Early Adopters Workgroup (OEAW) activities.

Outcomes:

- Shared understanding of current OSCAL issues.
- Shared understanding of current FedRAMP OSCAL Versioning strategies.

Agenda:

- Welcome
- Guiding Principles/Mission Review
- OEAW Updates
- FedRAMP Versioning Discussion
- Open Forum
- Next Steps & Closing

FedRAMP OEAW Guiding Principles

Keep the discussion respectful

Be curious, seek understanding

Speak from your own experience

Challenge through questions

Focus on ideas

Keep it technical

General Updates

FedRAMP Automation Community Updates

March 13, 2024

Revising OSCAL Guides

- FedRAMP automation team is continuing to work towards publishing HTML versions of the OSCAL guides to replace the current PDF versions.

Local Validation Tooling

- FedRAMP automation team is working on adding metaschema validation mechanisms in the OSCAL-CLI tool

Current Work

- A revised separation of duties proposal (#534)
- Guidance for printable documentation generation based on FedRAMP templates (#558)

GitHub Issues

- Prioritizing issues related to FedRAMP Guides and SP 800-53 Rev 5

https://github.com/orgs/GSA/projects/25/views/3

Closed PRs

- #540 Local version of SP 800-53 with zero padded labels; updated based on usnistgov/oscal-content#238
- #557 Container support for user guides

Review needed

https://github.com/orgs/GSA/projects/25/views/7

FedRAMP Versioning

FedRAMP OSCAL Versioning

Need a versioning strategy that can handle incremental changes to FedRAMP OSCAL requirements

- The FedRAMP version needs to:
  - Minimize disruption for implementers due to disruptive changes
  - Enable baselines, guides, and validations to move as a unit with a given release
  - Provide content and tooling producers a clear version target for implementation
  - Allow for OSCAL version changes that likely require tooling adjustments
- Semantic or schema versioning can be a useful way of signaling compatibility
  - https://semver.org/
  - https://snowplow.io/blog/introducing-schemaver-for-semantic-versioning-of-schemas/

Versioning - Current State

What are we versioning?

Source versions we are aligning to:

- SP 800-53 catalog - changes based on SP 800-53 catalog revisions
- OSCAL release - changes to OSCAL models
- FedRAMP release - baselines, guides, and validations, including FedRAMP extensions

Currently using:

- An overall version that combines FedRAMP and OSCAL versions, e.g., fedramp-2.0.0-oscal-1.0.2
- Baseline versioning based on SP 800-53 version, e.g., 5.1.1+fedramp-20240111-0

Versioning - Proposal

The following is a proposal for consideration.

Going forward we will:

- Use a single, FedRAMP version scheme for releases in FedRAMP baselines, guides, and validations
- In the release notes:
  - Declare the OSCAL MAJOR version and minimum MINOR/PATCH version required
  - Indicate the SP 800-53 version supported
- Use SchemaVer or SemVer for this version

FedRAMP OSCAL Versioning

Semantic Versioning (SemVer)

SemVer is a versioning convention designed to convey meaning about the underlying code changes in a software release.

Uses:

- Incrementing the MAJOR version indicates significant changes.
- Incrementing the MINOR version adds functionality in a backward-compatible manner.
- Incrementing the PATCH version indicates backward-compatible bug fixes.

Implications:

- A MAJOR OSCAL version change will occur with a MAJOR change in FedRAMP version
- A MINOR or PATCH OSCAL version change can occur with a MINOR change in FedRAMP version
- A non-compatible SP 800-53 change will occur with a MAJOR change in FedRAMP version
- A compatible SP 800-53 change can occur with a MINOR change in FedRAMP version

Example of FedRAMP using SemVer:

fedramp-3.0.1 (Major: 3, Minor: 0, Patch: 1)

FedRAMP OSCAL Versioning

Schema Versioning (SchemaVer)

SchemaVer is a versioning convention specifically designed for versioning data schemas, providing a means to minimize disruption to existing data and applications relying on those schemas.

Uses:

- MODEL version is a breaking schema change which will prevent interaction with any historical data.
- REVISION version is a schema change which may prevent interaction with some historical data.
- ADDITION version is a schema change that is compatible with all historical data.

Implications:

- A MAJOR OSCAL version change will occur with a MODEL change in FedRAMP version
- A MINOR or PATCH OSCAL version change can occur with a REVISION change in FedRAMP version
- A non-compatible SP 800-53 change will occur with a MODEL change in FedRAMP version
- A compatible SP 800-53 change can occur with a REVISION change in FedRAMP version

Example of FedRAMP using SchemaVer:

fedramp-3.0.1 (MODEL: 3, REVISION: 0, ADDITION : 1)

Versioning Discussion

Feedback

1. Does this proposed versioning approach make sense?
2. Will the proposed version approach be useful to you?
3. Should FedRAMP use SemVer or SchemaVer?
4. What else should FedRAMP consider when implementing a versioning scheme?

Open Forum

Thank you

Our next OEAW virtual meeting will be on

Wednesday, March 27th, 2024 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**

Collaboration Resources

FedRAMP Automation GitHub: https://github.com/GSA/fedramp-automation

- Open Issues: https://github.com/GSA/fedramp-automation/issues
- Open Pull Requests: https://github.com/GSA/fedramp-automation/pulls
- Active Work: https://github.com/orgs/GSA/projects/25/views/3
- Community Review Needed: https://github.com/orgs/GSA/projects/25/views/7

GitHub Resources:

- Issues: https://docs.github.com/en/issues
- Pull Requests: https://docs.github.com/en/pull-requests

OSCAL Resources

NIST:

OSCAL repo: 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
Early Adopter repo: https://github.com/GSA/fedramp-oscal-earlyadopters