Rules: September 2022 Edition

Why rules? Where's the gap?

SC-28 PROTECTION OF INFORMATION AT REST

Protect the [Selection (one or more): confidentiality; integrity] of the following information at rest: [Assignment: organization-defined information at rest].

RMF 800-53 Search

SC-28 PROTECTION OF INFORMATION AT REST

- What do we protect and how do they relate?
- What do confidentiality, integrity, and availability mean?
- How do we define encryption?
- How and where do we encrypt at rest?

Control: What and why you protect the C.I.A of systems and information. (high-level)

Method: How you act to analyze evidence of control's requirements. (high-level)

Rule: What aspect of a part of system you evaluate. It may or may not relate to a control. (granular)

Test: How do you specifically evaluate an aspect of a rule. It may or may not relate to a method for analyzing if control requirements are met. (granular)

Who does what?

Is it always the same people?

Caroline Component Supplier

- Works in SuperCSP security team to:
  - oversee security monitoring for stakeholders and customers
  - document secure configuration for customers
- Needs structured info to:
  - accelerate customers secure configuration
  - provide structured test procedures to quickly test configuration

Evelyn Evaluator

- Works at EvaluatorCorp and writes:
  - hardening guidelines for SuperCSP and open-sources software
  - automated tools to test for hardening guidelines
- Has tools and guidelines disconnected from assessment
- Needs tools and guidelines connected to assessment

Dina Developer

- Works at BigCorp to:
  - develop a BigCorp SaaS product built with SuperCSP services
  - design and implement with security in mind
  - maintain parts of SSPs for BigCorp services
  - provide updated technical evidence of SSPs requirements

Owen System Owner

- Works at BigCorp to:
  - act as business owner of a BigCorp SaaS product
  - scope assessments for authorizations
  - approve and owns product documented in SSP
  - sponsor BigCorp’s authorization of product for customer use

Arthur Architect

- Works at BigCorp to:
  - guide developer teams to design systems well
  - lead developer teams of different products to build a combined BigCorp architecture
- Needs to scale best practice to developers for architecture
- Needs methods for developers to test products for best practice

Oscar Operations

- Works at BigCorp to:
  - maintain a BigCorp SaaS product
  - manage product security monitoring program
  - oversee product vulnerability management program
- Needs automatable ways to generate evidence
- Needs repeatable ways to summarize evidence for management
- Needs to respond to POA&M inquires from management

Andy Assessor

- Works at AssessorCorp to:
  - consult with customers like BigCorp on security
  - advise on regulatory compliance
  - apply expertise in risk and security management
  - understand challenges and solutions for heterogenous architectures
  - analyze customers and recommend actionable improvements
- Needs better ways to parallelize repeat tasks

Not All Personas Are Equal

- Personas are not always consistent, persistent
- Some real-world roles blend personas (DevOps)
- Some personas exist all in one organization, sometimes in 2 or 3
- The persona's role in organization(s) changes how they interact with:
  - rules
  - tests
  - bindings of rules and tests
  - linkage to controls' or control statements' requirements as evidence

Caronline Component Supplier

- “I understand SuperCSP products and write rules for them.”
- “Customers ask us about other tools for security testing, I need to show the customer how to relate our tests as evidence to their rules.”
- “I understand SuperCSP tools for tests.”
- “Some customers use our rules for hardening but want to use their own tools to test. I need to show them how they relate.”

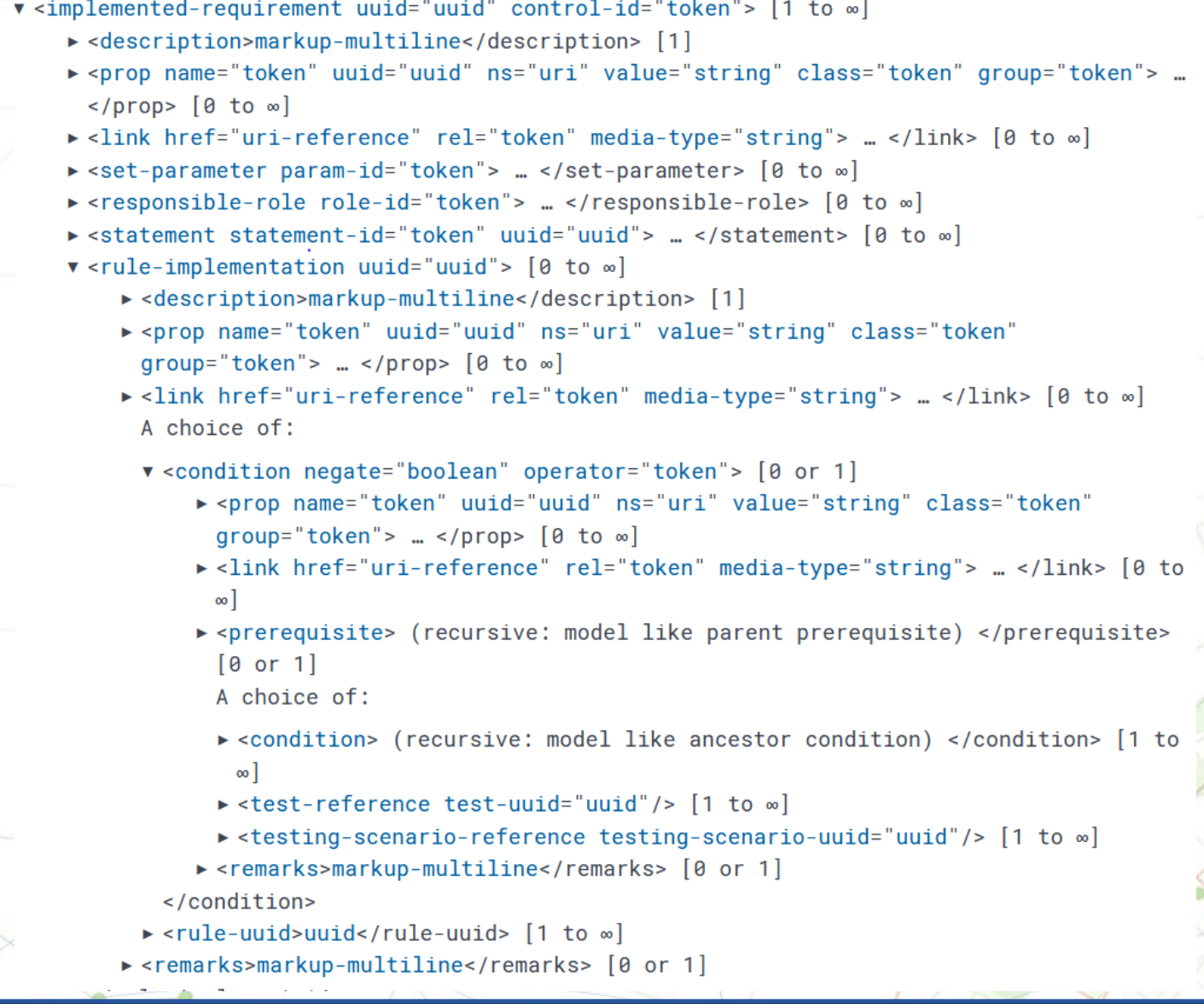
Evelyn Evaluator

- “I document hardening guidelines for EvaluatorCorp also using commercial and governmental standards.”
- “I know all the EvaluatorCorp tests. I can add them to a component-definition.”
- “Sometimes I know how to bind rules and tests together for EvaluatorCorp. Otherwise only the customer can know.”

```
▼ <component-definition uuid="uuid"> [1]
    ▶ <metadata> … </metadata> [1]
    ▶ <import-component-definition href="uri-reference"/> [0 to ∞]
    ▼ <rule uuid="uuid"> [0 to ∞]
        ▶ <title>markup-line</title> [1]
        ▶ <description>markup-multiline</description> [1]
        ▶ <prop name="token" uuid="uuid" ns="uri" value="string" class="token" group="token"> … </prop>
          [0 to ∞]
        ▶ <link href="uri-reference" rel="token" media-type="string"> … </link> [0 to ∞]
        ▶ <remarks>markup-multiline</remarks> [0 or 1]
      </rule>
    ▶ <test uuid="uuid"> … </test> [0 to ∞]
    ▶ <testing-scenario uuid="uuid" rule-uuid="uuid"> … </testing-scenario> [0 to ∞]
    ▶ <component uuid="uuid" type="string"> … </component> [0 to ∞]
    ▶ <capability uuid="uuid" name="string"> … </capability> [0 to ∞]
    ▶ <back-matter> … </back-matter> [0 or 1]
  </component-definition>
```

```
▼ <component-definition uuid="uuid"> [1]
    ▶ <metadata> … </metadata> [1]
    ▶ <import-component-definition href="uri-reference"/> [0 to ∞]
    ▶ <rule uuid="uuid"> … </rule> [0 to ∞]
    ▼ <test uuid="uuid"> [0 to ∞]
        ▶ <title>markup-line</title> [0 or 1]
        ▶ <description>markup-multiline</description> [0 or 1]
        ▶ <prop name="token" uuid="uuid" ns="uri" value="string" class="token" group="token"> … </prop>
          [0 to ∞]
        ▶ <link href="uri-reference" rel="token" media-type="string"> … </link> [0 to ∞]
        ▶ <remarks>markup-multiline</remarks> [0 or 1]
      </test>
    ▶ <testing-scenario uuid="uuid" rule-uuid="uuid"> … </testing-scenario> [0 to ∞]
    ▶ <component uuid="uuid" type="string"> … </component> [0 to ∞]
    ▶ <capability uuid="uuid" name="string"> … </capability> [0 to ∞]
    ▶ <back-matter> … </back-matter> [0 or 1]
  </component-definition>
```

```
▼ <component-definition uuid="uuid"> [1]
    ▶ <metadata> … </metadata> [1]
    ▶ <import-component-definition href="uri-reference"/> [0 to ∞]
    ▶ <rule uuid="uuid"> … </rule> [0 to ∞]
    ▶ <test uuid="uuid"> … </test> [0 to ∞]
    ▼ <testing-scenario uuid="uuid" rule-uuid="uuid"> [0 to ∞]
        ▶ <prop name="token" uuid="uuid" ns="uri" value="string" class="token" group="token"> … </prop>
          [0 to ∞]
        ▶ <link href="uri-reference" rel="token" media-type="string"> … </link> [0 to ∞]
          A choice of:
          ▶ <condition negate="boolean" operator="token"> … </condition> [1]
          ▶ <test-reference test-uuid="uuid"/> [1]
        ▶ <remarks>markup-multiline</remarks> [0 or 1]
      </testing-scenario>
    ▶ <component uuid="uuid" type="string"> … </component> [0 to ∞]
    ▶ <capability uuid="uuid" name="string"> … </capability> [0 to ∞]
    ▶ <back-matter> … </back-matter> [0 or 1]
  </component-definition>
```



National Institute of Standards and Technology
U.S. Department of Commerce

ITL/CSD/OSCAL Team
Model Meeting 20220902

Questions? Comments?

github.com/usnistgov/OSCAL/issues/1058

gitter.im/usnistgov-OSCAL/Lobby

oscal@nist.gov