Open Security Controls Assessment Language

What is OSCAL and Who Needs It?

Dr. Michaela Iorga,
OSCAL Strategic Outreach Director

March 10, 2022
Why are we all here today?

Information technology is complex & calls for automation

Regulatory frameworks are burdensome & Need interop auto GRC tools

Paper-based A&A doesn’t scale & Calls for auto updates

Security vulnerabilities are everywhere & Calls for auto updates

Risk management is hard & Experts need automation

DevOps & IaC is hard in multi-clouds & Calls for interoperability & standardization

Before the audit

During the audit

After the audit
What was needed?

A (Cyber) Machine-readable Esperanto that enables actors, tools and organizations to exchange information via automation:

OSCAL sets the foundation for automation and interoperability
OSCAL is the result of NIST and FedRAMP collaboration

- **OSCAL provides** a common/single machine-readable *language*, expressed in XML, JSON and YAML for:
  - multiple compliance and risk management frameworks (e.g. SP 800-53, ISO/IEC 27001&2, COBIT 5)
  - software and service providers to express implementation guidance against security controls (Component definition)
  - sharing how security controls are implemented (System Security Plans [SSPs])
  - sharing security assessment plans (System Assessment Plans [SAPs])
  - sharing security assessment results/reports (System Assessment Results [SARs])

- **OSCAL enables** automated traceability from selection of security controls through implementation and assessment
First OSCAL Release

OSCAL 1.0.0 WAS RELEASED ON JUNE 7, 2021

https://github.com/usnistgov/OSCAL/releases/tag/v1.0.0

“...First official, major release of OSCAL provides a stable OSCAL 1.0.0 for wide-scale implementation ...”
Few of the OSCAL Adopters

- FedRAMP
- Noblis
- HHS CMS
- National Renewable Energy Lab
- GovReady
- C2 Labs
- cFocus Software
- Shujinko
- Robers Bosch (EU|Germany)
- Telos
- KPMG
- IBM Research

2021 presenters

- AWS
- CSAM
- Easy Dymanics
- Volant Associates LLC
- Secureframe
- Red Hat
- Nirmata
- SunStone Secure

2022 new presenters

- US AirForce Platform One
- Booz Allen Hamilton
- eMASS
- Microsoft
- Coalfire
- Kratos
- Salesforce
- Oracle

2021-2022 other adopters
How is OSCAL different?

- No information needs duplication
- Custom granularity (controls can be decomposed into statements)
- Unique identifiers for parameters and statements
- Vendors can document their products
- Systems’ security implementation can be decomposed
- Capture assessment Plans and Activities with custom cadence, & only for selected components
- POA&M conveys open risks aligned with the SSP capabilities and controls
A Closer Look at OSCAL Models
What can you do with the OSCAL models?
OSCAL Models    >>>    OSCAL Content    >>>   OSCAL Tools

OSCAL Models

https://github.com/usnistgov/OSCAL

OSCAL Content

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

OSCAL Tools

https://github.com/usnistgov/oscal-tools
OSCAL Supports Continuous Authorization to Operate (ATO)
- System Assessment Automation -
Who Can Benefit & How?

Risk Management & OSCAL content

RMF steps: PREPARE  CATEGORIZE  SELECT  IMPLEMENT  ASSESS  AUTHORIZE  CON-MON

Who Can Benefit & How?
OSCAL Supports Complex Systems
Authorization to Operate (ATO)
Authorization to Use (ATU)
Common Control Authorization
**Common Control Authorization & Authorization to Use**

**Yes**

- **Cloud (SaaS on IaaS):** Several SaaS systems running on a separately authorized IaaS.
- **Data Center (System on GSS):** Several systems relying on a separately authorized storage array or other general support system (GSS).

**Yes**

- **Cloud (SaaS on IaaS):** Several SaaS systems running on a separately authorized IaaS.
- **Data Center (System on GSS):** Several systems relying on a separately authorized storage array or other general support system (GSS).

**No**

- **Cloud (SaaS on IaaS):** Several SaaS systems running on a separately authorized IaaS.
- **Data Center (System on GSS):** Several systems relying on a separately authorized storage array or other general support system (GSS).
- **External Service or Interconnection:** Interconnections or External Services are not leveraged authorizations.
  - Even if they have an authorization
  - SaaS A handles the Identity Management Service as a system component

**OSCAL** supports this, just not as a L.A.
OSCAL supports leveraged ATOs of complex stacked systems

**Leveraging System:**

The leveraging system's SSP should:

- identify what is inherited from a leveraged system
- identify any addressed responsibilities (as identified by the leveraged system)

**In addition to:**

- identifying what may be inherited by the leveraging system's customers
- any responsibilities the leveraging system's customers must address to fully satisfy a control
Blockchain-based Secure Software Assets Management = A (proof-of-concept) software assets (leasing) management service using AWS Managed Blockchain (Hyperledger Fabric (v2.2)) [NIST-DHS-UMBC Cooperative Research and Development Agreement (CRADA)]
OSCAL: the Open Security Controls Assessment Language

Automated Control-Based Assessment
Supporting Control-Based Risk Management with Standardized Formats

Providing control-related information in machine-readable formats.

NIST, in collaboration with industry, is developing the Open Security Controls Assessment Language (OSCAL). OSCAL is a set of formats expressed in XML, JSON, and YAML. These formats provide machine-readable representations of control catalogs, control baselines, system security plans, and assessment plans and results.
<table>
<thead>
<tr>
<th>Name</th>
<th>Provider/Developer</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compliance trestle</td>
<td>IBM</td>
<td>A python SDK and command line tool which manipulates OSCAL structures and supports transformation of data into OSCAL.</td>
<td>open source</td>
</tr>
<tr>
<td>OSCAL Java Library</td>
<td>NIST OSCAL Project</td>
<td>A Java-based programming API for reading and writing content conformant to the OSCAL XML, JSON, and YAML based models.</td>
<td>open source</td>
</tr>
<tr>
<td>OSCAL React Component Library</td>
<td>Easy Dynamics</td>
<td>A library of reusable React components and an example user interface application that provides a direct UI into OSCAL.</td>
<td>open source</td>
</tr>
<tr>
<td>OSCAL REST API</td>
<td>Easy Dynamics</td>
<td>An initial OpenAPI definition of an OSCAL REST API that describes how systems might manipulate catalogs, profiles, components, and SSPs.</td>
<td>open source</td>
</tr>
<tr>
<td>XSLT Tooling</td>
<td>NIST OSCAL Project</td>
<td>A variety of Extensible Stylesheet Language (XSL) Transformations (XSLT), Cascading Style Sheets (CSS), and related utilities for authoring, converting, and publishing OSCAL content in various forms.</td>
<td>open source</td>
</tr>
<tr>
<td>XML Jelly Sandwich</td>
<td>Wendell Piez (NIST)</td>
<td>Interactive XSLT in the browser includes OSCAL demonstrations.</td>
<td>open source</td>
</tr>
<tr>
<td>Xacta 360</td>
<td>Telos</td>
<td>Xacta 360 is a cyber risk management and compliance analytics platform that enables users to create and submit FedRAMP system security plans (SSPs) in OSCAL format. Future OSCAL capabilities are forthcoming as the standard evolves.</td>
<td>license</td>
</tr>
<tr>
<td>Atlassian Continuous</td>
<td>C2 Labs</td>
<td>Atlassian CE (release 2.0) runs in any environment and supports the development of OSCAL v1.0 content for Catalogs, Profiles, System Security Plans and Components. Additional detail can be found in this blog post: Atlassian Delivers Free Tools to Create OSCAL Content.</td>
<td>community edition</td>
</tr>
<tr>
<td>Atlassian Continuous</td>
<td>C2 Labs</td>
<td>Atlassian CE (release 2.0) runs in any environment and supports the development of OSCAL v1.0 content for Catalogs, Profiles, System Security Plans and Components. Additional detail can be found in this blog post: Atlassian Delivers Free Tools to Create OSCAL Content.</td>
<td>community edition</td>
</tr>
<tr>
<td>control freak</td>
<td>Risk Redux</td>
<td>This tool seeks to provide folks with a searchable and easy-to-navigate reference for NIST SP 800-53 Revision 5. It is an open-source application from the Risk Redux project, built using parsed content directly from the OSCAL repositories.</td>
<td>open-source</td>
</tr>
</tbody>
</table>
How to Contribute?

OSCAL is a community-driven effort.

Your participation directly impacts OSCAL’s success.

Integrate support for OSCAL in your tools
Implement OSCAL-based tools in your enterprise.
Contribute to the development of OSCAL on GitHub.

Attend the bi-weekly community meetings hosted by NIST.

https://github.com/usnistgov/OSCAL
https://github.com/usnistgov/OSCAL/blob/main/CONTRIBUTING.md
https://pages.nist.gov/OSCAL/contribute/#community-meetings
Publicly Available Resources

**Documentation:**
Catalog, Profile, Component, SSP, SAP, SAR, POA&M:
https://pages.nist.gov/OSCAL/documentation/

**Example:**
Generic examples:
https://github.com/usnistgov/oscal-content/tree/master/examples
NIST SP 800-53 R4 and Rev5 catalog and baselines (XML & JSON):
https://github.com/usnistgov/oscal-content/tree/master/nist.gov/SP800-53

**FedRAMP Automation:**
Repository (FedRAMP catalog and baselines (XML & JSON) included):
https://github.com/GSA/fedramp-automation
https://www.fedramp.gov/using-the-fedramp-oscal-resources-and-templates/

**Tools**
- OSCAL Java Library: https://github.com/usnistgov/liboscal-java
- XSLT Tooling: https://github.com/usnistgov/oscal-tools/tree/master/xslt
- OSCAL Kit: https://github.com/docker/oscalkit
- OSCAL GUI: https://github.com/brianrudges/OSCAL-GUI

Please visit Community’s:
OSCAL Club/awesome-oscal:
https://github.com/oscal-club/awesome-oscal
Questions?

Contact us at: oscal@nist.gov
Chat with us on Gitter: https://gitter.im/usnistgov-OSCAL/Lobby
Collaborate with us on GitHub: https://github.com/usnistgov/OSCAL
Join our COI meetings: https://pages.nist.gov/OSCAL/contribute/#community-meetings

Thank you!