Continuous, Automated Compliance with OSCAL

March 2nd, 2022

Contact Information
Conner Phillippi: conner@secureframe.com
Apostolos Delis: apostolos@secureframe.com
About Us

Conner Phillippi
Product

Apostolos Delis
Engineering Lead
Agenda

1. Introduction on Secureframe
2. Compliance Challenges
3. Secureframe Platform & OSCAL Applications
4. Group Q&A
Intro on Secureframe

- Founded in January 2020
- Headquartered in San Francisco and Denver
- Automating security compliance for 500+ companies
- Helping customers stay secure, meet regulatory obligations, and break through customer & vendor security requirements
- Support SOC 2, ISO 27001, HIPAA, PCI DSS, CCPA, and GDPR compliance initiatives
- Entering the Federal compliance space in 2022 and will be able to support ANY framework
Compliance Challenges
Federal Compliance Challenges & Solutions

### Federal Challenges

1a. Federal compliance & ATO reviews are complex, obscure, and costly for vendors

1b. High federal compliance barriers cause a smaller pool of available vendors

2. Agencies are unable to truly monitor vendor compliance “continuously”

3. Internal compliance is tedious for agencies, especially when they are subjected to several compliance frameworks

4. Agencies struggle to objectively interpret controls and assess their implementation

### Secureframe Solutions

- Simplify compliance requirements into an automated & actionable roadmap. Minimize vendor compliance readiness & audit costs. Validate SSPs to reduce time to ATO issuance.

- Notify agencies in real-time when their authorized vendor(s) fail controls (as opposed to requesting static deliverables) with set tolerances.

- Abstract all frameworks and controls via Secureframe and OSCAL. Push agencies to think in terms of security and risk rather than compliance to framework-specific controls.

- Normalize NIST 800-53, HIPAA, PCI DSS, SOC 2, etc. compliance data via Secureframe and OSCAL; promote global control implementation consensus
User Story A

Continuously Assess Your Own Compliance

- Automatically assess internal framework compliance
- Output compliance data in a standardized format for ingestion by other vendors, agencies, and assessors
1. Connect GitHub Account

Integrations

Connected Integrations

- **AWS**
  - Account ID: 267613903019
  - Last Sync: Feb 23, 2022 6:03am
  - Connected

- **Azure**
  - Subscription ID: b746d763-d330-4a8d-a71f-7f0accf3d679
  - Last Sync: Feb 23, 2022 6:15am
  - Connected

- **GitHub**
  - Connection #1
  - Last Sync: Feb 23, 2022 6:17pm
  - Connected

- **Office 365**
  - Organization ID: 694c08a2-cc5e-439a-a15c-d59079c4c151
  - Last Sync: Feb 23, 2022 6:32am
  - Connected
2. Configure GitHub Integration
3. Review Code Change Compliance Gaps

NIST Readiness Report

CM-3 3 Configuration Change Control
a. Determine and document the types of changes to the system that are configuration-controlled; b. Review proposed configuration-controlled changes to the system and approve or disapprove such changes with explicit consideration for security and privacy impact analyses; c. Document configuration change decisions associated with the system; d. Implement approved configuration-controlled changes to the system; e. Maintain records of configuration-controlled changes to the system.

1. Pull request approvals
   - Government Agency configures its version control tool to require at least 1 independent approver prior to pull request entering production.
   - Branch "release-1.3" from repository "Secureframe-dev-demosBackend" is not configured to require merge request approvals.

2. Branch merge request approvals
   - Government Agency requires branch merge requests to be independently reviewed and approved prior to merging to main, unless approved by an administrator due to a required emergency change.
   - Pull request "Add 2.0 file" from repository "Secureframe-dev-demosBackend" is not independently approved or labeled with an emergency label.
4a. Remediate Code Approval Gap

Branch “release-v1.3” is unprotected

Branch “release-v1.3” is protected
4b. Remediate Code Approval Gap

“Add 2.0 file” is missing approval/emergency label

“Add 2.0 file” is given emergency label
5. Secureframe Validates Remediations

CM-3 Configuration Change Control
a. Determine and document the types of changes to the system that are configuration-controlled; b. Review proposed configuration-controlled changes to the system and approve or disapprove such changes with explicit consideration for security and privacy impact analyses; c. Document configuration change decisions associated with the system; d. Implement approved configuration-controlled changes to the system;
6. Export API-Retrieved Evidence (.csv)

<table>
<thead>
<tr>
<th>Vendor</th>
<th>Repository</th>
<th>Audit Scope</th>
<th>Branch Name</th>
<th>Required Approved Reviews Count</th>
<th>Author Approval Enabled</th>
</tr>
</thead>
<tbody>
<tr>
<td>Github</td>
<td>Secureframe-dev-demo/Backend</td>
<td>TRUE</td>
<td>main</td>
<td>1</td>
<td>FALSE</td>
</tr>
<tr>
<td>Github</td>
<td>Secureframe-dev-demo/Backend</td>
<td>TRUE</td>
<td>release-v1.3</td>
<td>1</td>
<td>FALSE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Vendor</th>
<th>Repository</th>
<th>Private Repo</th>
<th>Emergency Label</th>
<th>Has Label</th>
<th>Base Branch Name</th>
<th>Pull Request Name</th>
<th>Merged</th>
<th>Pull Request Author</th>
<th>Independent App Opened At</th>
</tr>
</thead>
<tbody>
<tr>
<td>Github</td>
<td>Secureframe-dev-demo/B</td>
<td>TRUE</td>
<td>emergency</td>
<td>FALSE</td>
<td>main</td>
<td>Configure server settings</td>
<td>TRUE</td>
<td>AlecR</td>
<td>2022-01-19 1</td>
</tr>
<tr>
<td>Github</td>
<td>Secureframe-dev-demo/B</td>
<td>TRUE</td>
<td>emergency</td>
<td>TRUE</td>
<td>main</td>
<td>Add hello.py</td>
<td>TRUE</td>
<td>AlecR</td>
<td>2022-02-24 0</td>
</tr>
<tr>
<td>Github</td>
<td>Secureframe-dev-demo/B</td>
<td>TRUE</td>
<td>emergency</td>
<td>TRUE</td>
<td>release-v1.3</td>
<td>Add 2.0 file</td>
<td>TRUE</td>
<td>AlecR</td>
<td>2022-02-24 0</td>
</tr>
</tbody>
</table>
7. Export Report in OSCAL for Global Portability
8a. Reference (Catalog and Profile)
8b. Reference (System Security Plan)

Dynamic SSP Controls

Dynamic SSP Assertions
User Story B

Monitor Vendor Compliance & Configuration Drift

- Inform vendor risk assessments & evaluations
- Automate configuration drift detection for vendors/CSPs
1. Select a Vendor for Evaluation (AWS)
2a. Upload Agency Baseline & AWS OSCAL Files
2b. Reference (Agency Baseline and AWS SSPs)
3. Secureframe Creates a Diff of the Files
4. Review Control or Assertion Deviation

**Agency CSP Baseline SSP**

**AWS SSP**

---

Diff

- *(Red)* Identify the types of events that the system is capable of logging.
Group Q&A

Questions or Inquiries? Contact us at oscal@secureframe.com
Appendix
How It Works: Control Assessment

1. Create or Import Control from Framework

CM-3: Code changes must undergo security testing, integration testing, and independent approval prior to entering production

[SOC 2 Control Import]

2. Automatically Scope Control and Resources

1 GitHub repo, 1 primary branch, and 100 pull requests (PRs) are detected over the past 3 months

[Fetch time window of assessment]

[Fetch list of all version control integrations & components with “SOC 2“ audit scope tag]

3. Automate Control Testing with Prebuilt or Custom Tests

Test A: SAST occurred on all PRs before primary branch merge

Test B: Dependency checks ran on all PRs before primary branch merge

Test C: CI checks ran on all PRs before primary branch merges

Test D: 1 independent approval occurred on all PRs before primary branch merge

“Fix 123”: Missing independent approval

Test E: Independent approval is enforced as a branch configuration setting.

“Branch 123”: Branch protection disabled

4. Automatically Assess Control Implementation

CM-3: Partially Implemented

5. Automatically Collect Assessment Evidence and Generate SSP

Control CM-3: Test_A_E.zip Test_A_E.csv

Export SSP as: .pdf .json (OSCAL)

Test A: PASS
Test B: PASS
Test C: PASS
Test D: FAIL
Test E: FAIL
Commercial Compliance Time Savings

1 Week - Platform Onboarding
Onboard to the security compliance vendor’s automated platform

1 Week - Connect Integrations
Connect all integrations and receive automated readiness report

2 - 3 Weeks - Fix Gaps
Reconcile gaps via product guidance and vendor expertise

1 Week - Audit Ready
Perform ISO 27001 internal audit as applicable

2 Weeks - Complete SOC 2 Type 1 & ISO 27001 Stage 1
Start SOC 2 Type 2 audit window as applicable

7 Weeks - Complete ISO 27001 Stage 2
Typically a 2 month lapse between Stage 1 & 2

Secureframe

SOC 2 Type I
SOC 2 Type I Audit is complete and the report is received

ISO 27001
ISO 27001 Stage 2 Audit is complete and the certification is received

No Automation
General Challenges of the Compliance Process

1. **Collecting Audit Evidence is Manual & Time Consuming**
   - Internal applications do not automatically communicate with the auditing applications & process
   - Hundred screenshots are required in this process

2. **Maintaining Compliance Eats into Other Company Initiatives** — With the amount of cross-functional work required to initially complete and continuously maintain compliance, primary company initiatives are often impeded

3. **Understanding Compliance Requirements is Hard** — Compliance is a blackbox industry. Having guidance and direction is key for implementing the correct policies, technology, and internal processes.