Oregon State Police

Secure Software Development

Audit-Ready Vendor Engagement for Sensitive Data

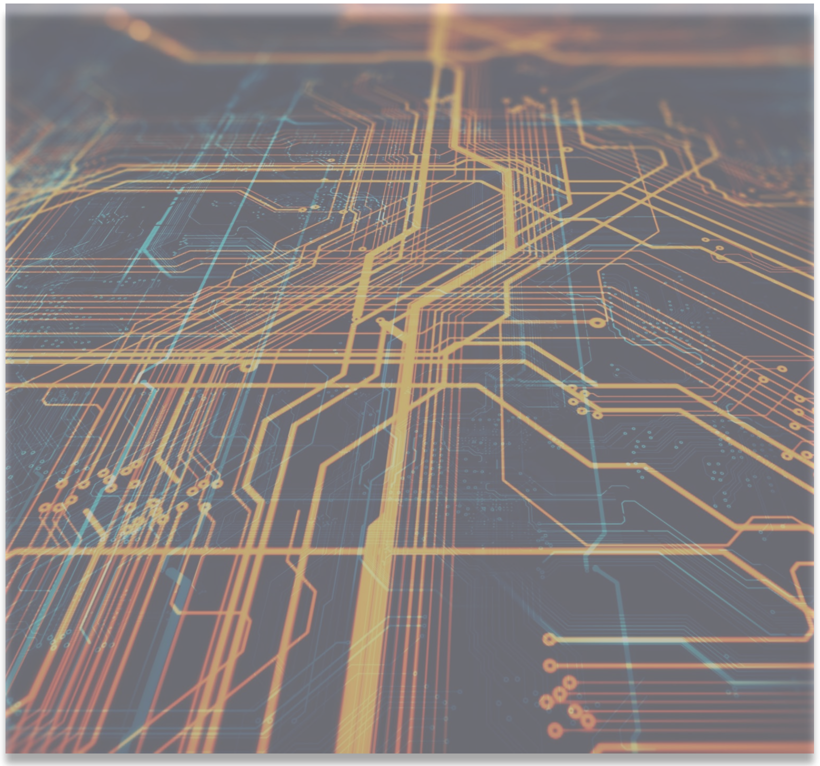
Nicholas Harris – CJIS Information Security Officer

UNCLAS (FOUO) – TLP GREEN

Welcome to the 2025 Cyber Resilience Summit!

Thank you for attending this year's, Summit!

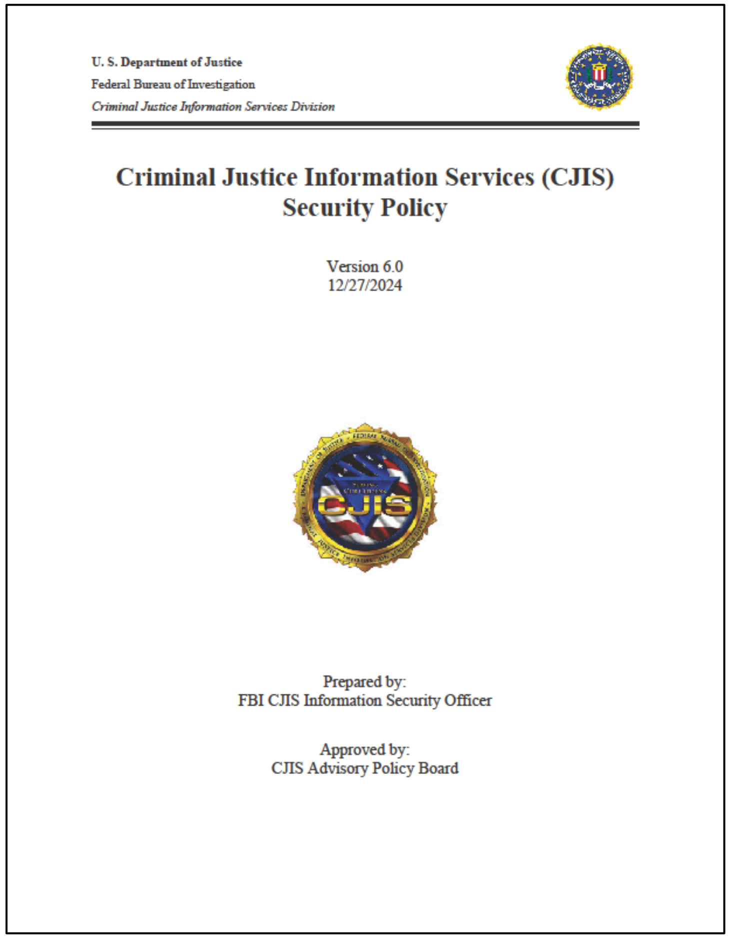
- Collaboration is key!
- Sharing information with the community!



Introduction & Objectives

- Importance of CJIS Security Policy and the NIST Special Publication 800-218
  - Learning outcomes:
    - Understanding the importance of Secure Software Development Framework.
    - Identifying the four core practices.
      1. Preparing your organization (PO)
      2. Protect Software (PS)
      3. Produce Well-Secured Software (PW)
      4. Respond to Vulnerabilities (RV)
  - Implementing SSDF into your procurement process.
  - How to Manage your software after implementation "Audit"
  - *This is about audit defensibility and vendor accountability!!*

Identify your authority to audit!



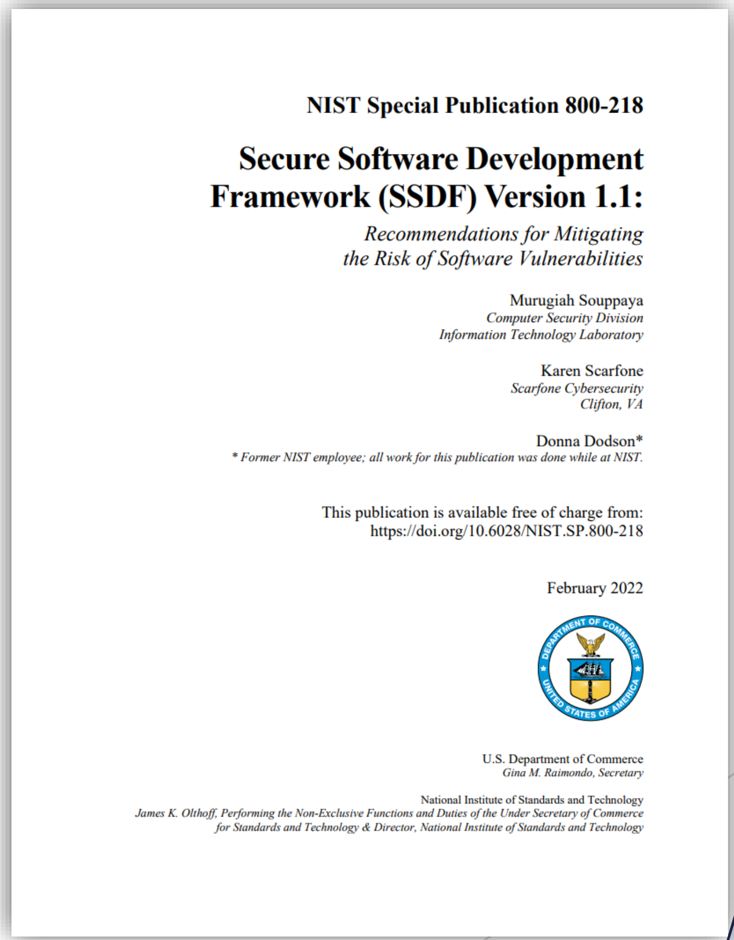
U. S. Department of Justice
Federal Bureau of Investigation
Criminal Justice Information Services Division

Criminal Justice Information Services (CJIS)
Security Policy

Version 6.0
12/27/2024

Prepared by:
FBI CJIS Information Security Officer

Approved by:
CJIS Advisory Policy Board



NIST Special Publication 800-218

Secure Software Development Framework (SSDF) Version 1.1:

Recommendations for Mitigating the Risk of Software Vulnerabilities

Murugiah Souppaya
Computer Security Division
Information Technology Laboratory

Karen Scarfone
Scarfone Cybersecurity
Clifton, VA

Donna Dodson*
** Former NIST employee; all work for this publication was done while at NIST.*

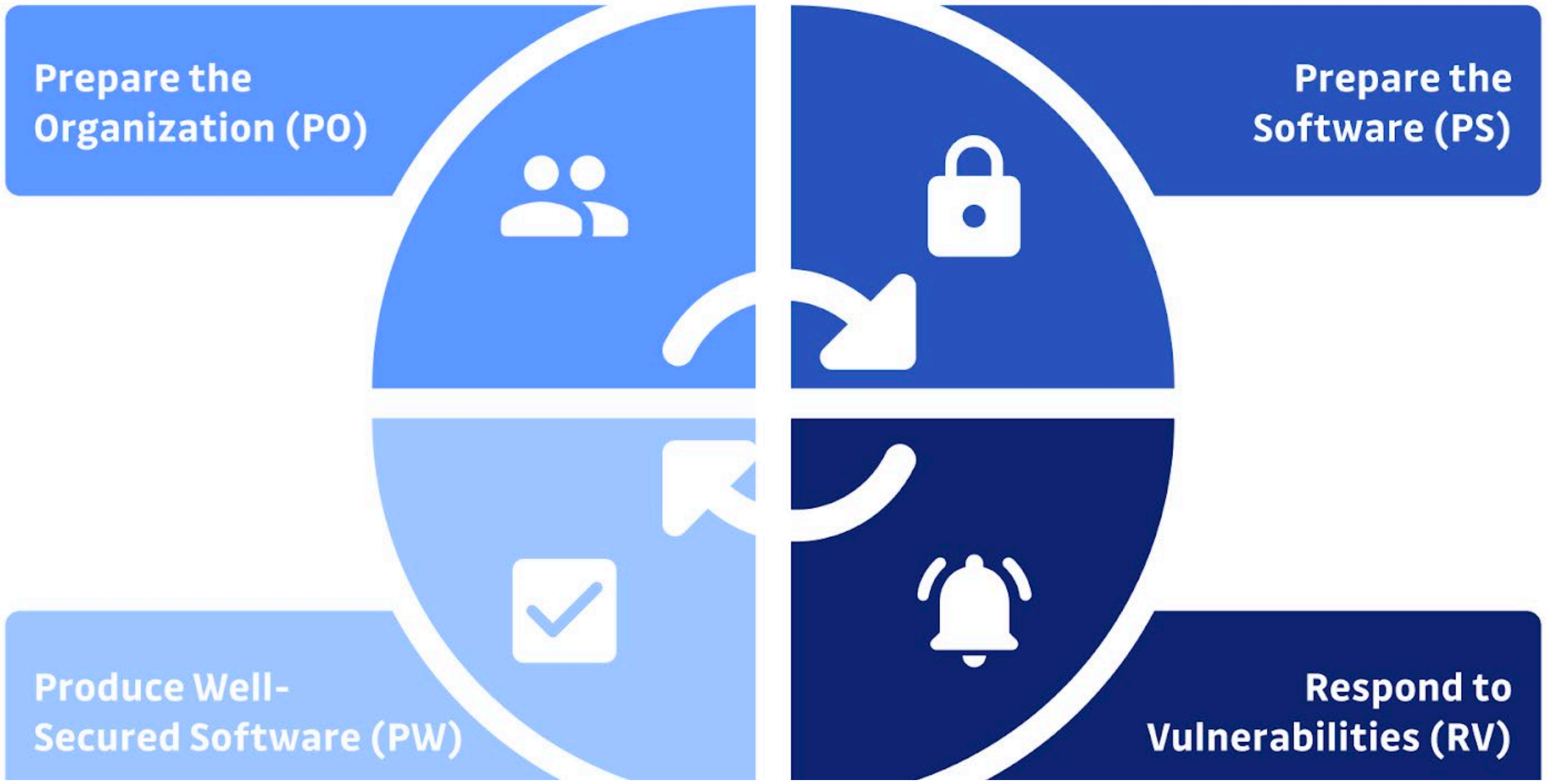
This publication is available free of charge from:
https://doi.org/10.6028/NIST.SP.800-218

February 2022

U.S. Department of Commerce
Gina M. Raimondo, Secretary

National Institute of Standards and Technology
James K. Olthoff, Performing the Non-Exclusive Functions and Duties of the Under Secretary of Commerce for Standards and Technology & Director, National Institute of Standards and Technology

SSDF Strategy..


Prepare the Organization (PO)
Prepare the Software (PS)
Produce Well-Secured Software (PW)
Respond to Vulnerabilities (RV)

The practices - Our battle plan for *cybersecurity….*

Image provided by GoDucks

SSDF Practices

PO: Prepare the Organization

- Establish security policies and governance
- Assign accountable roles and responsibilities
- Vet personnel and vendors for trustworthiness

SSDF Practices

PS: Protect the Software

- Enforce secure coding standards
- Apply encryption and authentication controls
- Manage dependencies and patching

SSDF Practices

PW – Produce Well-Secured Software

- Secure builds from commit through release
- Integrate testing and validation for vulnerabilities
- Protect secrets and enable audit-ready logging

SSDF Practices

RV – Respond to Vulnerabilities

- Detect, triage, and remediate flaws quickly
- Capture lessons learned for continuous improvement
- Demonstrate incident response readiness to auditors

Case Study: Vendor X and End-of-Life Apache Tomcat

- **Background:**
  Vendor X, a third-party service provider supporting a state law enforcement agency, hosted a web application that processed Criminal Justice Information (CJI). During an internal review, it was discovered that the application was running on **Apache Tomcat 7.x**, a version that had reached **end-of-life (EOL)** and was no longer receiving security patches.

Case Study: Vendor X and End-of-Life Apache Tomcat

- **Audit Gap:**
  The agency’s annual compliance audit did not flag this issue because Vendor X provided only a high-level system description and attested to “using supported software.” No technical evidence (e.g., version scans, patch management logs) was requested or reviewed.

Case Study: Vendor X and End-of-Life Apache Tomcat

- **Implications if Undetected:**
- **Security Exposure:** EOL Tomcat contained multiple known vulnerabilities (e.g., remote code execution, privilege escalation) that could be exploited by attackers to gain unauthorized access to CJI.
- **CJIS Violation:** Running unsupported software directly violated CJIS Security Policy Section **System and Services Acquistion (SA) System and Communications Protection (SC) SI-2 (Flaw Remediation)** and **CM-2 (Baseline Configuration)**.
- **Operational Risk:** A successful exploit could compromise sensitive law enforcement data, disrupt investigations, and erode public trust.
- **Audit Consequences:** If discovered post-incident, the agency would face findings of **non-compliance**, potential federal oversight, and reputational damage for failing to enforce vendor accountability.
- **Vendor Accountability:** Without evidence-based validation, Vendor X's self-attestation created a false sense of security, leaving the agency exposed to risks that could have been mitigated with proper patch management and audit scrutiny.

Case Study: Vendor X and End-of-Life Apache Tomcat

- **Lesson Learned:**
  This case underscores the importance of **evidence-driven vendor audits**. Agencies should require vendors to provide verifiable artifacts—such as vulnerability scans, patch logs, and configuration baselines—rather than relying solely on attestations. Integrating SSDF practices like **Respond to Vulnerabilities (RV)** and **Protect the Software (PS)** ensures that unsupported software is identified early, reducing the likelihood of systemic risk.

Closing Thoughts & Q&A

- Key takeaways:
  - SSDF is essential.
  - NIST 800-53 require oversight in managing application security.
  - Protects the security, integrity and availability of sensitive data such as Criminal Justice Information (CJI).

Follow up

Contact info for follow-up

- Nicholas.harris@osp.Oregon.gov
- (503)934-2335

GO DUCKS!! BEAT INDIANA!!