NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
U.S. DEPARTMENT OF COMMERCE

DevSecOps and the Impact of Agentic Super Intelligence (SI)

October 28, 2026

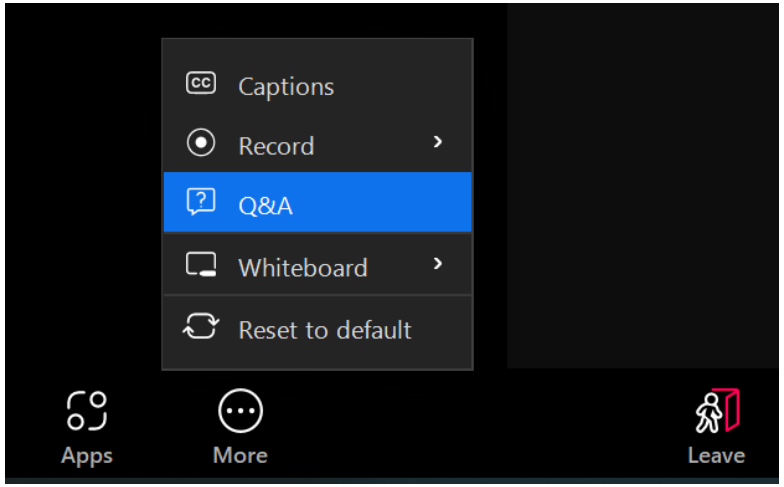
1:00 P.M. – 3:00 P.M. ET

This webinar is being recorded

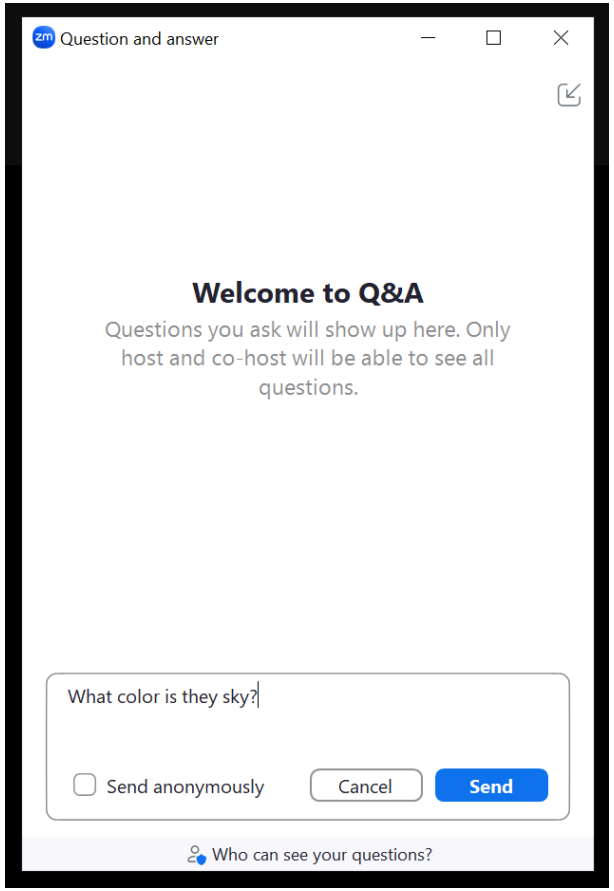
Submitting Questions

Please use the Q&A function to enter your questions.

We will do our best to answer all questions during the Q&A.



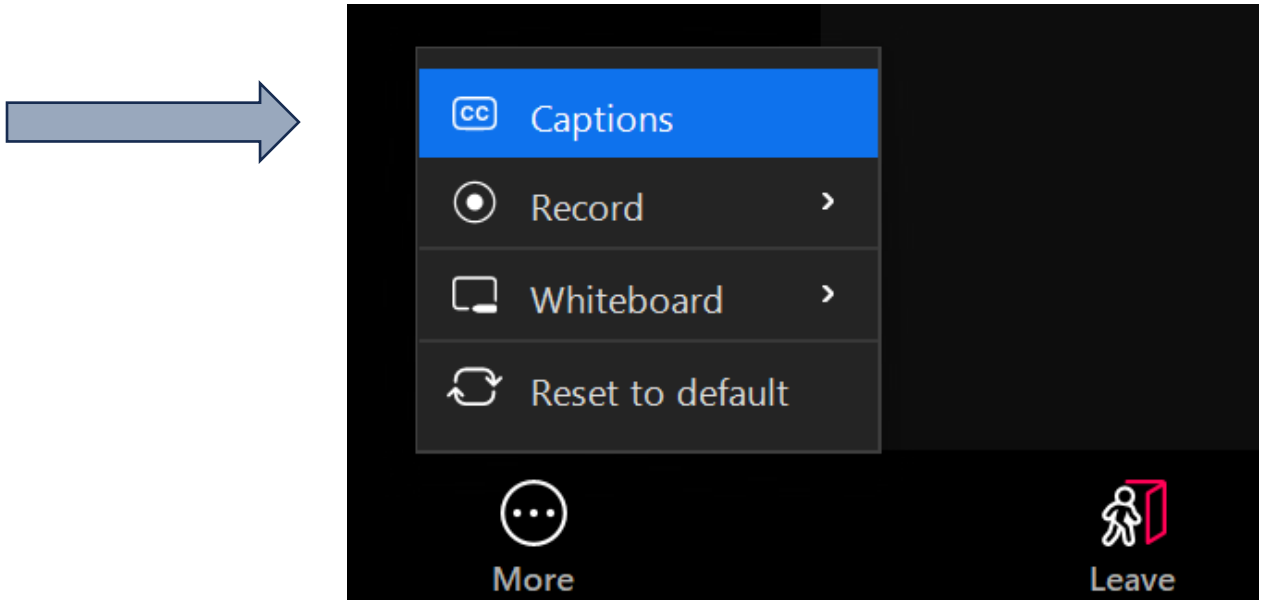
1. To open the Q&A function, click on “More” at the bottom of the screen and select the “Q&A” option.



2. Type your question in the text box and click Send

Enabling Closed Captions

To open enable captioning during the event, click on “More” at the bottom of the screen and select the “Captions” option.



Opening Remarks

Cherilyn Pascoe, Director, NCCoE (NIST)

Agenda

Time	Session
1:00 p.m. – 1:05 p.m.	**Welcome** *Speaker: Cherilyn Pascoe, NIST*
1:05 p.m. – 1:30 p.m.	**NCCoE DevSecOps Practices Project Overview** *Speaker: Alper Kerman and Michael Ogata, NIST*
1:30 p.m. – 1:50 p.m.	**NCCoE Project Update on Identity and Authorization of Software and SI Agents** *Speaker: Bill Fisher, NIST*
1:50 p.m. – 1:55 p.m.	**Break**
1:55 p.m. – 2:35 p.m.	**Panel Discussion: Impact of SI Agents in DevSecOps** *Moderator: Bill Fisher, NIST* *Panelists:* • *Bryan Ritchie, GitLab* • *Al Huizenga, Google* • *Segu Riluvan, Microsoft*
2:35 p.m. – 3:00 p.m.	**Q&A** *Moderator: Ryan Galluzzo, NIST*

NCCoE DevSecOps Practices Project Overview

Alper Kerman and Michael Ogata, NIST

Background and Motivation

NIST NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

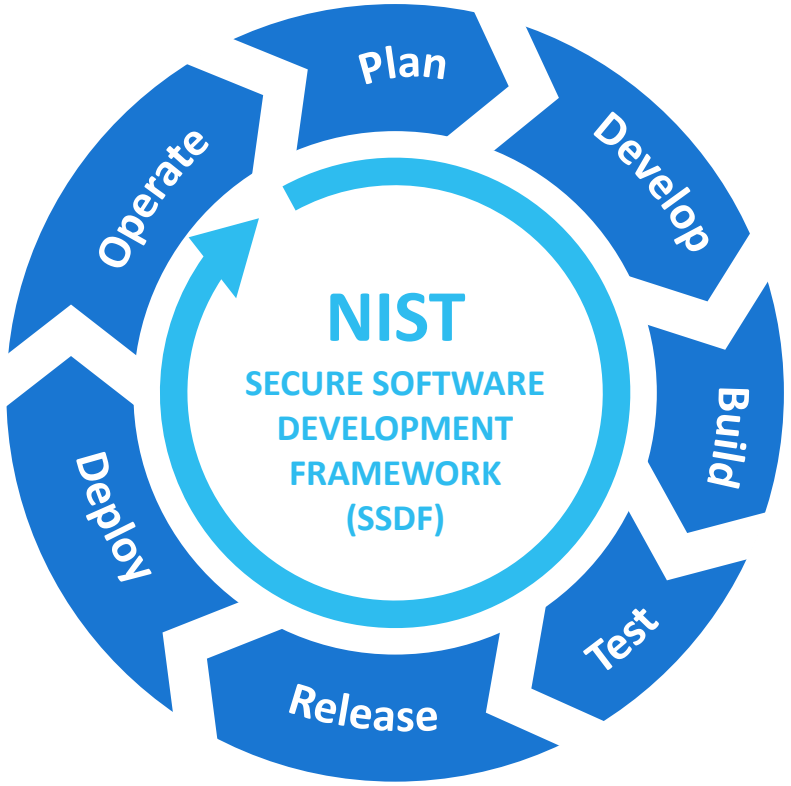
OBJECTIVE:

Develop and document an applied risk-based approach and recommendations for secure software development, security, and Operations (DevSecOps) practices consistent with:

- Secure Software Development Framework (SSDF)
- Other NIST, government, and industry guidance

SCENARIOS:

Demonstrate DevSecOps practices in multiple proof-of-concept use case scenarios that involve software development environments integrated with different industry technologies, **including security practices associated with the use of SI capabilities and Zero Trust Architectures**.



***This project was initiated in March 2025 to support the directive outlined in **Section 2(c)(i) of Executive Order 14306**

Secure Software Development Framework (SSDF)



4 Groups | **19 Practices** | **42 Tasks**

Outcome Oriented and Process Agnostic

Prepare the Organization (PO): Ensure the organization's people, processes, and technology are prepared to perform secure software development at the organization level and, in some cases, also for each individual project.

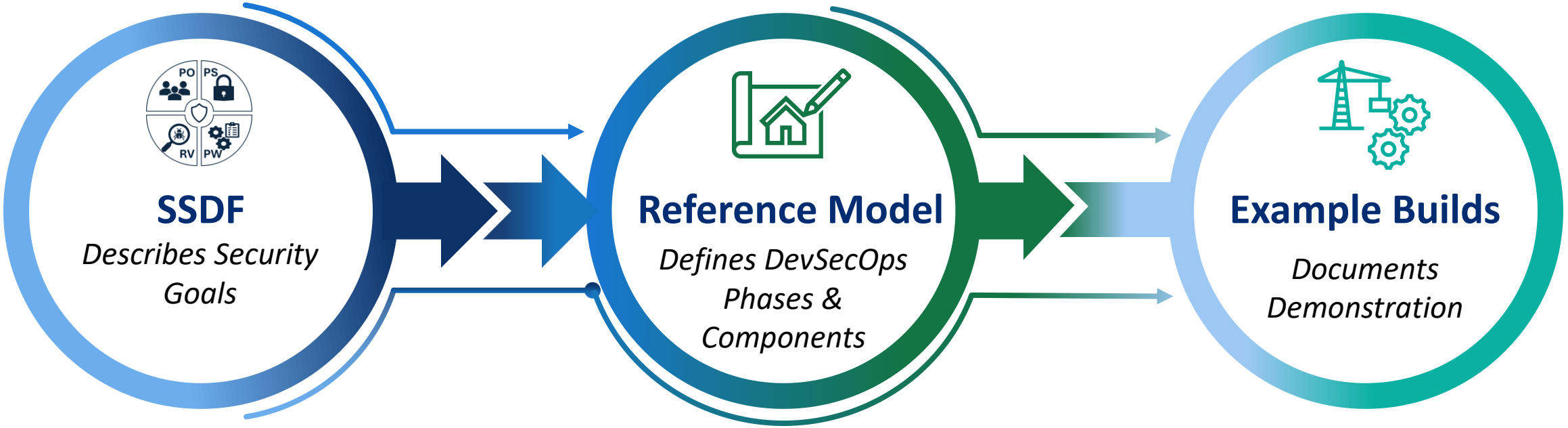
Protect the Software (PS): Protect all components of the software from tampering and unauthorized access.

Produce Well-Secured Software (PW): Produce well-secured software that has minimal security vulnerabilities in its releases.

Respond to Vulnerabilities (RV): Identify vulnerabilities in software releases and respond appropriately to address those vulnerabilities and prevent similar vulnerabilities from occurring in the future.

https://csrc.nist.gov/Projects/ssdf

Project Philosophy



Notional Reference Model

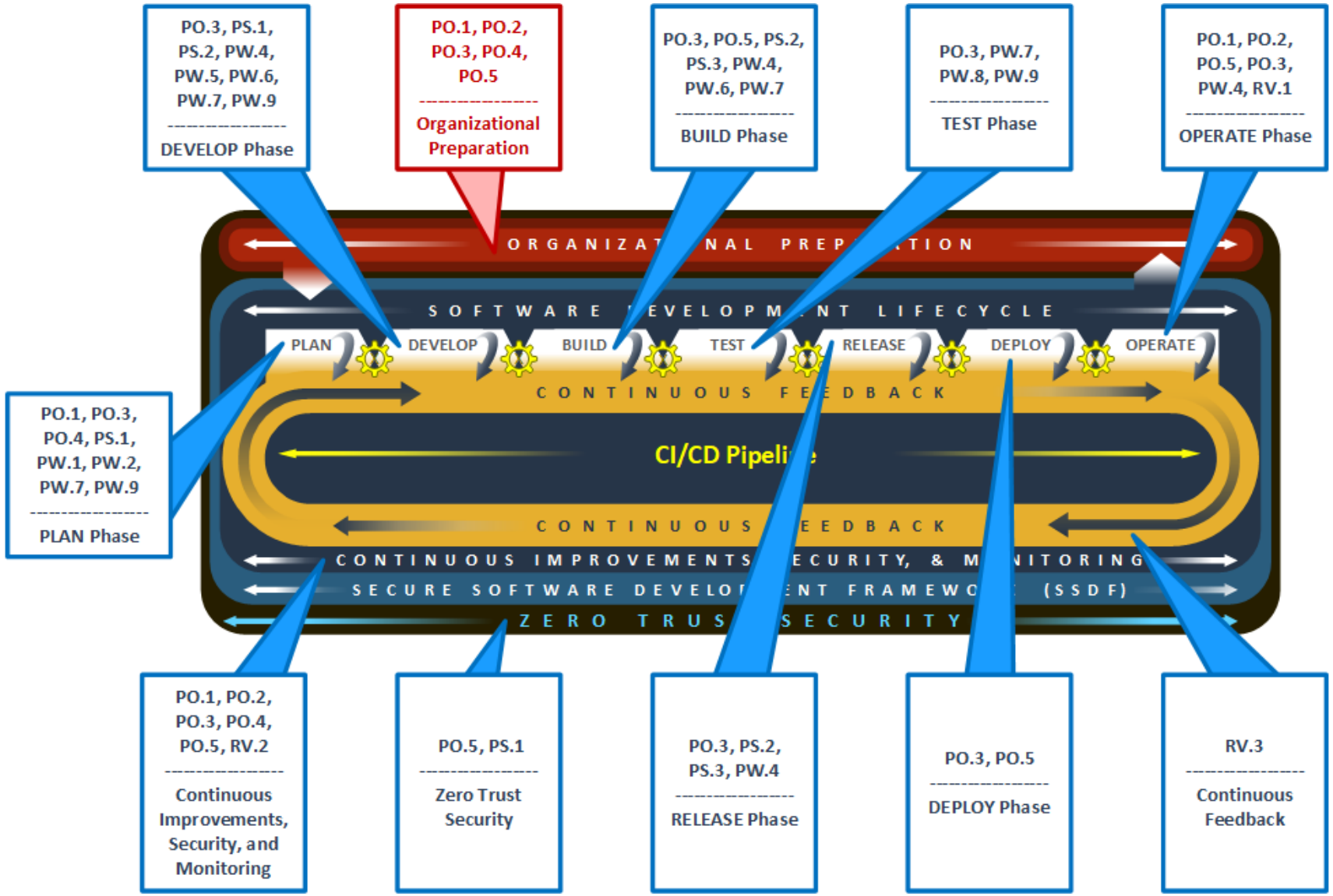
NIST | NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

Development, Security, and Operations (DevSecOps)



SSDF Mapping

NIST NATIONAL CYBERSECURITY CENTER OF EXCELLENCE



Over 60 Tool Categories Mapped to SSDF v1.1

The full taxonomy, including specific capability assignments for PO, PS, PW, and RV is documented in the

Live Document

System, CI/CD Execution, Test and Se… Policy Verification Tool, CI/CD Pipeline, Configuration Management …em, Container Image Scanner, Credential Management System, Cyb… telligence, Threat, and Security Metadata Feeds (e.g., OpenSSF, Na… lnerability Database (NVD), Open Source Vulnerabilities (OSV), and Co… Vulnerabilities and Exposures (CVE)/ Common Weakne… loyment Management Syste… and Canary), Developer Tools (e… ces, Fuzz Test Tool, Hardware Security … tual HSMs), IaC Scanner, IaC Script… Tool, Operations Monitoring System… anagement, and Performance Moni… oduct Management System, Project Ma… eam Collaboration, and Training), Prov… Regression Test Tool, Release Man… ent System, Risk Management Syste… AST System, SCA System, SCM Syste… nd Branch/Merge Protection), Secret… curity Monitoring System (e.g., Vulnerability Management, Incident Management, Security Information and Event Management (SIEM), and Security Orchestration, Automation, and Response (SOAR)), Smoke Test Tool, Software Libraries (e.g., Internal and External), Team Collaboration Tools, Threat Modeling System, Ticketing System, Unit Test Framework, Zero Trust Security System.

Evolution of Architectural Timeline

How Each DevSecOps Implementation Shifted Our Baseline Security Approach

NIST | NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

AUG 2025 – FEB 2026

Example Implementation 1 (includes Gen SI)

COMPLETED & PUBLISHED

WHAT WE BUILT

- Established the cloud computing environment and automation (IaaS/SaaS)
- Deployed foundational cloud-based platforms and tools
- Implemented source code signing
- Integrated Zero Trust, Gen SI, security, and project management platforms

THE NCCoE INSIGHT: Generative SI assist with code generation and analysis to accelerate project management, software development, and code review. Also, integrating security tools and platforms into CI/CD automation enhances supply chain risk management by enforcing policies and best practices on credentials, software dependencies, and automated pipelines.

COLLABORATORS: Microsoft, Black Duck, CyberArk, DigiCert, Endor Labs, Sagittal AI

MAR 2026 – SEPT 2026

Example Implementation 2 (includes Gen SI)

COMPLETED & PUBLISHED

WHAT WE BUILT

- Expanded Example Implementation 1 foundation with new SCM and CI/CD platforms, security tools, and ZTA components
- Deployed GitOps-managed Kubernetes (K8s) microservice architecture
- Integrated automated certificate lifecycle management for K8s applications
- Expanded C-SCRM including more SBOM and SLSA capabilities

THE NCCoE INSIGHT: In addition to Example Implementation 1 insights, integration of certificate management components, ZTA enforcement on microservices, and CI/CD platforms provide trusted compute for automation and deployed applications.

COLLABORATORS: GitLab, Microsoft, CyberArk, DigiCert, NextLabs, Resilience Cyber Security, Sagittal AI

SEPT 2026 - TBD

Example Implementation 3 (includes Agentic SI)

IN PROGRESS & ACTIVE

WHAT WE ARE BUILDING

- Designing next version of Notional Reference Model for software development with Agentic SI
- Implementing next-generation identity, authentication, and access management for agentic SI in cloud platforms
- Developing autonomous dual-loop SDLC (inner/outer) with multi-agent sandboxing

THE ANTICIPATED NCCoE INSIGHT: Autonomous agents converge Develop, Build, and Test phases into a machine-paced inner loop by pairing probabilistic agents with deterministic automation. Human roles elevate from manual code review to defining specifications, security guardrails, and adjudicating high-consequence policy exceptions.

COLLABORATORS: Google, Palo Alto Networks, and other collaborators (TBD)

Integrating Generative SI and Zero Trust Across Example Implementations 1 & 2

NIST | NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

OVERARCHING SECURITY STRATEGY: ZERO TRUST ARCHITECTURE

Strict Access & Least Privilege
Users and systems are granted minimal privileges; unapproved resources (repos, infrastructure) remain strictly inaccessible.

Preventing Lateral Movement
Continuous verification of every access attempt isolates workloads and strictly prevents lateral network movement.

Isolating & Securing Identities
Pipeline environments and actors (users/systems) are assigned unique, independent, zero-trust adhered identities.

Securing Pipeline Actions
All pipeline actions and code repository actions (e.g., merge requests) must be initiated and approved by authorized entities.

GENERATIVE SI CAPABILITIES (Operating Securely Within the Environment)

Planning & Task Management
- Requirements generation and organization
- Task decomposition and organization
- Ticketing and issue generation

Code Development & Testing
- Code generation and modification
- Pull and merge request support
- Test generation

Security & Infrastructure
- Cloud resource and service configuration
- Code scanning & security analysis
- Vulnerability detection and remediation

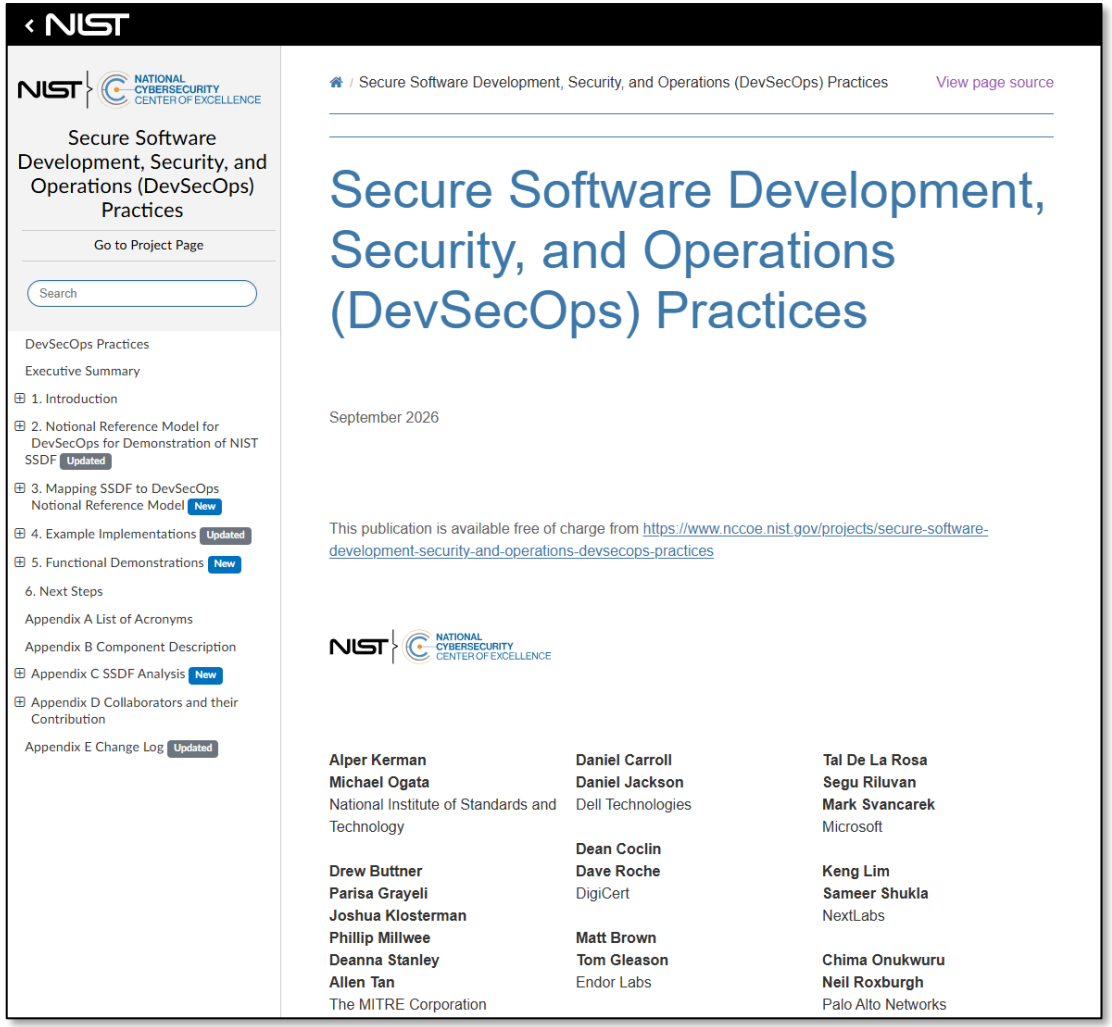
Device Posture Verification
Access to pipeline resources is granted only if the initiating device's security posture meets predefined organizational policies.

Securing Artifacts & Containers
Zero Trust policies are applied downstream by enforcing strict source provenance, attestation, and artifact signing requirements.

Continuous Diagnostics
DevSecOps infrastructure and systems are continuously monitored, triggering automated mitigations for policy violations.

The Zero Trust Architecture acts as the overarching framework, completely enveloping all human, machine, and SI Actors

Project Output: The Live Document



‹ NIST

NIST NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

Secure Software Development, Security, and Operations (DevSecOps) Practices

Go to Project Page

Search

DevSecOps Practices
Executive Summary
1. Introduction
2. Notional Reference Model for DevSecOps for Demonstration of NIST SSDF Updated
3. Mapping SSDF to DevSecOps Notional Reference Model New
4. Example Implementations Updated
5. Functional Demonstrations New
6. Next Steps
Appendix A List of Acronyms
Appendix B Component Description
Appendix C SSDF Analysis New
Appendix D Collaborators and their Contribution
Appendix E Change Log Updated

/ Secure Software Development, Security, and Operations (DevSecOps) Practices View page source

Secure Software Development, Security, and Operations (DevSecOps) Practices

September 2026

This publication is available free of charge from https://www.nccoe.nist.gov/projects/secure-software-development-security-and-operations-devsecops-practices

NIST NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

Alper Kerman, Michael Ogata, National Institute of Standards and Technology
Daniel Carroll, Daniel Jackson, Dell Technologies
Dean Coclin, Dave Roche, DigiCert
Matt Brown, Tom Gleason, Endor Labs
Tal De La Rosa, Segu Riluvan, Mark Svancarek, Microsoft
Keng Lim, Sameer Shukla, NextLabs
Chima Onukwuru, Neil Roxburgh, Palo Alto Networks
Drew Buttner, Parisa Grayeli, Joshua Klosterman, Phillip Millwee, Deanna Stanley, Allen Tan, The MITRE Corporation

DevSecOps Practices Live Document:

Current Draft Version Release Date: September 2026

Core Architecture & Framework

Establishing the foundation for secure software lifecycles:

- **Executive Summary & Introduction:** Describes DevOps to DevSecOps evolution, key drivers for adoption, alignment with NIST SSDF, project background, scope, challenges and how to use this guide.
- **Notional Reference Model:** Unveils the 7 key phases of the SDLC (Plan, Develop, Build, Test, Release, Deploy, Operate) against CI/CD pipelines, Continuous Monitoring, and Feedback loops.
- **SSDF to DevSecOps Mapping:** Aligns the architecture with the four core pillars of the Secure Software Development Framework (SSDF):
  - **Prepare the Organization** (PO)
  - **Protect the Software** (PS)
  - **Produce Well-Secured Software** (PW)
  - **Respond to Vulnerabilities** (RV)

Applied Scenarios & Appendices

Practical Implementations, testing, and technology integration:

- **Functional Demonstrations:** Extensive, scenario-based guides (Scenarios A through H) covering actionable workflows such as: Threat Modeling & Bug Tracking DAST / IAST / API Testing SBOM Management & Validation Infrastructure as Code (IaC) Analysis.
- **Appendix C (SSDF Analysis):** A granular decomposition of SSDF practices, detailing precise security checks like verifying software release integrity and archiving secure releases.
- **Appendix D (Collaborators):** Real-world implementations highlighting commercial technology partners (e.g., Microsoft Azure, IBM Cloud, Black Duck, CyberArk, etc.).

Project Collaborators

NIST | NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

- AMI
- Black Duck
- CyberArk
- Dell Technologies
- DigiCert
- Endor Labs
- GitLab
- Google
- IBM
- Microsoft
- NextLabs
- Palo Alto Networks
- Resilience Cyber Security
- Sagittal AI

Project Resources and Contact Information

Access live guidance, join the Community of Interest, and connect with the project team

NIST | NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

EXPLORE PROJECT INFORMATION	CONNECT DIRECTLY
Project Landing Page https://www.nccoe.nist.gov/projects/secure-software-development-security-and-operations-devsecops-practices • **Browse Live Document** • **Join COI** • **Track Events**	**Project Team Email** **Email:** NCCoE-DevDecOps@list.NIST.gov • **Technical feedback** • **Collaboration Qs** • **Speaker requests**

Federal & Industry Collaboration | National Cybersecurity Center of Excellence (NCCoE)

Slido Question #1

We update our live document regularly and want to reflect your real-world needs.
What specific example implementations and scenarios should we consider demonstrating next?

To participate in Slido, scan the QR code or go to Slido.com and enter the access code: #DevSecOps

Slido.com
#DevSecOps

Slido Question #2

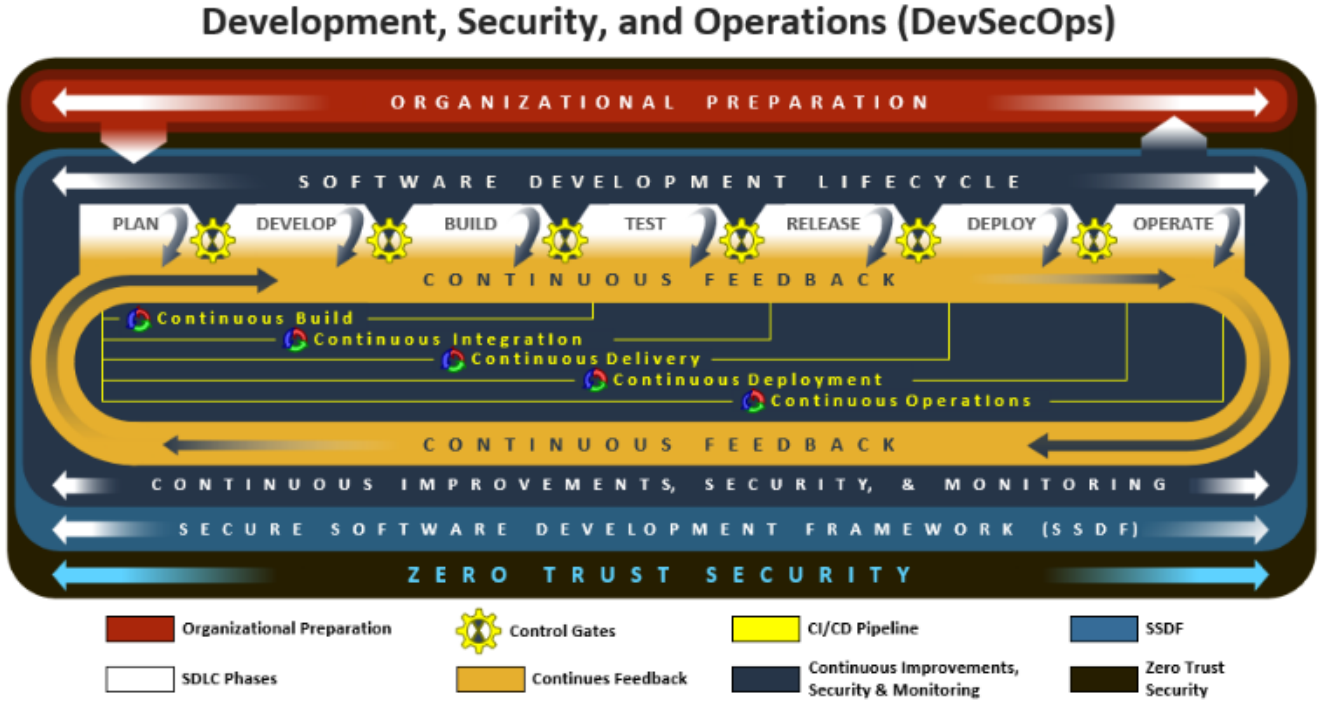
How do you see Super Intelligence (SI) agents changing the SDLC?

Slido.com
#DevSecOps

NCCoE Project Update on Identity and Authorization of Software and SI Agents

Bill Fisher, NIST

Demonstrating Agents in the SDLC

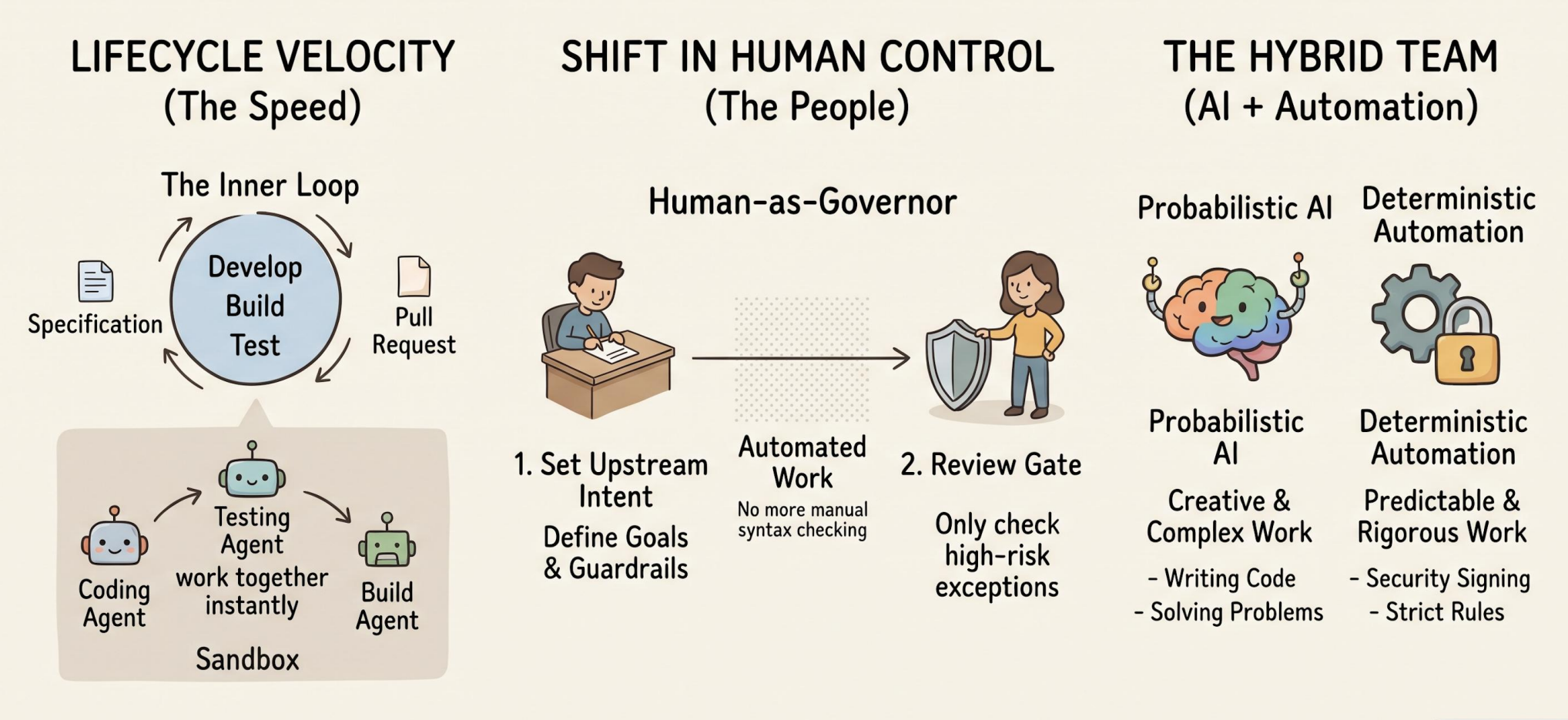


Phase 1: Enterprise SI Agents in DevSecOps

- Project will highlight how agents change the DevSecOps lifecycle and demonstrate agentic use
- Identity and Authorization principles will be included as part of architecture design and zero trust principles
- Project Scoping Live on the Online Resource Hub

How Do Agents Change the SDLC?

NIST | NATIONAL CYBERSECURITY CENTER OF EXCELLENCE



Note: SI-generated diagram using Gemini

The Actors

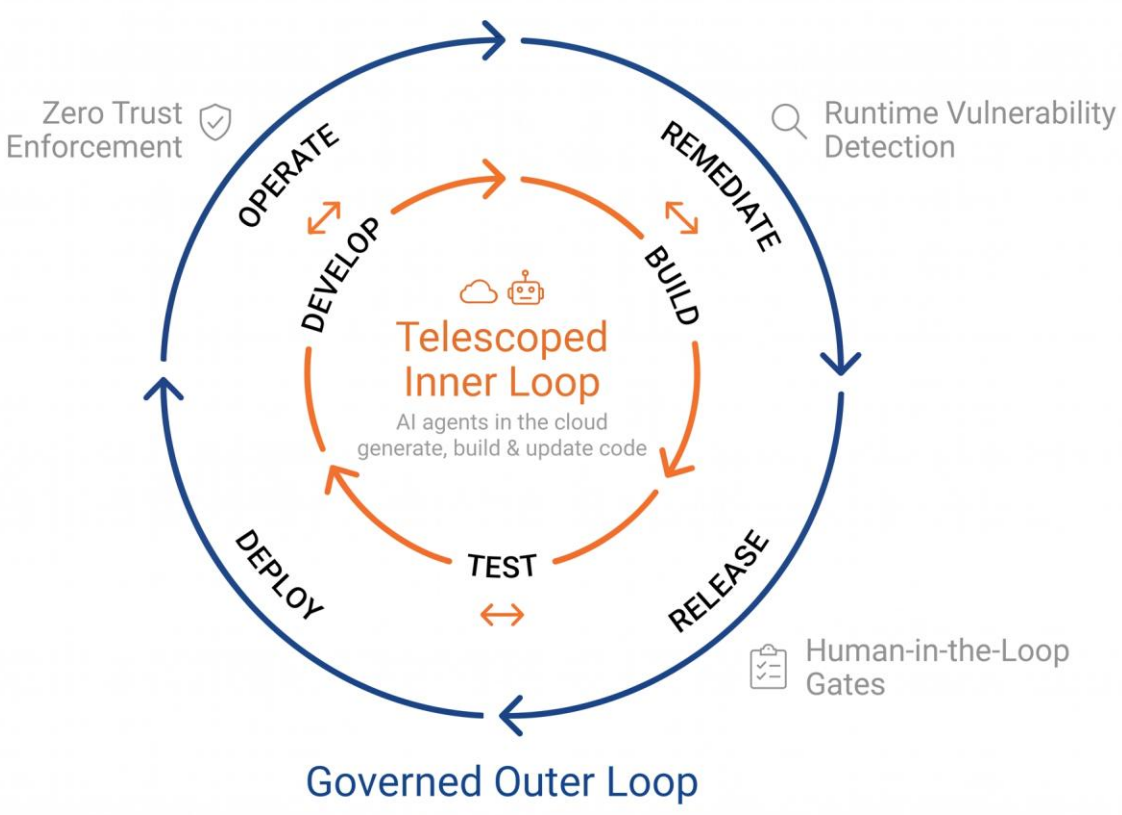
NIST NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

Actor	Primary responsibility	DevSecOps examples
Human	Set objectives, architecture, policy, risk tolerance; approve consequential changes; own accountability.	Security architect approves threat model; risk owner accepts residual risk; release owner makes Go/No-Go.
Agentic AI	Understand context; decompose goals; analyze across artifacts; select approved tools; execute permitted actions; evaluate outcomes; escalate.	Investigate a SAST finding, patch in a branch, run tests/scans, create PR, summarize evidence, and request approval.
Deterministic Automation	Execute deterministic, repeatable operations and enforce explicit policy.	CI/CD runs SAST/SCA/DAST, tests, signing, SBOM generation, policy checks, deployment, monitoring, and rollback rules.

Demonstrating Two Loops

Example Implementation will demonstrate two loops:

1. Inner loop – the fast paced Develop -> Build -> Test
    - Agentic actions include: Analyze planning data, generate code, unit testing & evaluation, generate patches, generate pull requests, generate Human-in-the-loop requests
2. Outer Loop – Release -> Deploy -> Operate
    - Agentic actions include: Analyze vulnerability reports, conduct vulnerability assessments, build environments to test and locate vulnerabilities, generate code for patches, validate patches, generate Human-in-the-loop requests, output summary of actions.



Note: SI-generated diagram

Securing the Agentic Lifecycle

NIST | NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

Demonstrating agentic identity capabilities aligned with broader zero trust strategy:

AGENTIC IDENTITY

- Will need to bind workload identity, operating instance, and active authority.
- Requires distinct non-human identities to differentiate agents from human users.

PERSISTENCE & EPHEMERALITY

- Stable hardware/software trust anchors bound to underlying runtime environment (SPIFFE/SPIRE)
- Highly ephemeral, task-scoped credentials feature auto-expiration and attenuation.

TASK SCOPED & CONTEXTUAL AUTHORIZATION

- Shifts from static inherited permissions to dynamic, continuous Zero-Trust evaluation.
- Evaluates authorization at tool call/moment of action to limit blast radius and privilege escalation.

AUDITABILITY BEYOND ACTION LOGGING

- Captures reasoning context, decision policies, delegations, and intent alongside actions.
- Implements tamper-evident, verifiable evidence chains and cryptographic receipts.

DELEGATION AND ACCOUNTABILITY

- Preserves authorization context across multi-hop, cross-domain agent chains.
- User signed tokens conveying delegation scope, constraints, and audit evidence.

DETERMINISTIC VS. PROBABILISTIC AUTHORIZATION

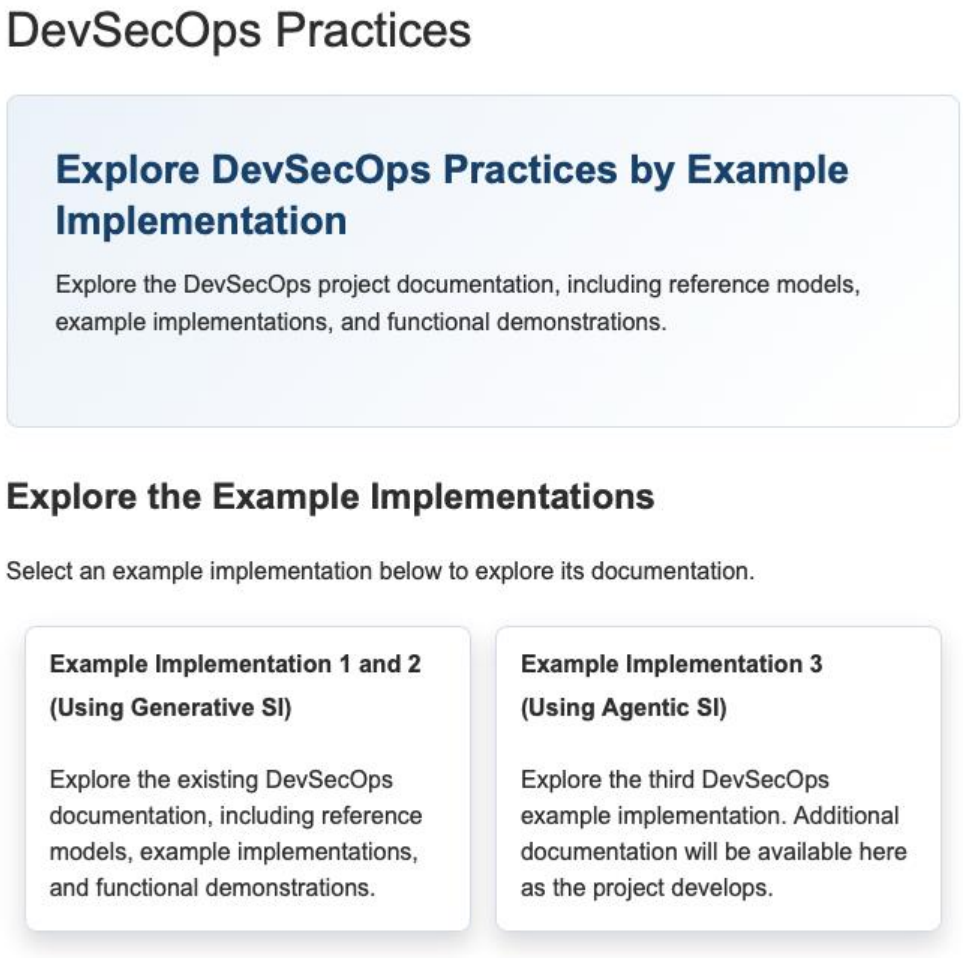
- Deterministic authorization is a priority
- Probabilistic (LLM-generated) information as a contextual risk signal.

Project Outputs

NIST NATIONAL CYBERSECURITY CENTER OF EXCELLENCE

Resources & Deliverables:

- Blogs and informative materials to drive discussion, education, and awareness
- Iterative and informative resources – architectures, profiles, and considerations that will inform the final practice guides
- NIST SP 1800 series practice guide with build details, reference architectures, and laboratory lessons learned.



Break

We will resume at 1:55 p.m.

Slido Question #3

What are the biggest security challenges facing organizations looking to deploy agents in the enterprise?

Slido.com
#DevSecOps

Panel Discussion:
Impact of SI Agents in DevSecOps

Moderator:

Bill Fisher, NIST

Panelists:

Bryan Ritchie, GitLab
Al Huizenga, Google
Segu Riluvan, Microsoft

Q&A

Moderator: Ryan Galluzzo, NIST

Send us your comments!

Public comment period is open through November 9, 2026

https://www.nccoe.nist.gov/projects/secure-software-development-security-and-operations-devsecops-practices

