Topic Area: Guidelines for software supply chain integrity and provenance

Contact Information: Daniel Riedel, SVP, Copado, DRiedel@copado.com, 415-264-1997

Auditable & Immutable Software Development Lifecycle, SBOM and DevSecOps:

As society builds new software we must build greater traceability and attribution within the software development process. There are too many interconnections of software components and that software now connects to physical assets that have potential to cause great harm or even death. Because of legacy software it may be impossible to know the identity of every person involved in the software development process. To manage that legacy software and gain a better understanding of all the components of software, the National Telecommunications and Information Administration (NTIA) software bill of materials (SBOM) is crucial to reduce the risk of unknown actors, potential defects, and security vulnerabilities that can cause harm.

Just as all of our civil engineering standards enforce strict guidelines for how we license and manage architectural and building standards, we must follow suit in our software development process as it is as important to the proper functions of society as our bridges, hospitals, power stations, water supply, banks, roads and other critical infrastructure that we rely on as a nation.

Auditable & Immutable Software Development Lifecycle - Value Stream Management

NIST should take advantage of the expanding Value Stream Management (VSM) for DevSecOps market by introducing standards of creating a centralized immutable auditable record within the value stream. The goal is to deliver all future software development using supply chain assurance by creating immutable records around the entire software development lifecycle (SDLC). This extends beyond traditional log management and security information and event management (SIEM) environments and looks to the future state software development of an entire enterprise. It is paramount that software developers record every action within the development process before code is sent into production. This includes binding an identity to all contributed assets delivered within the SDLC. Each object, action, and subcomponent is recorded in a centralized separate record system to build a comprehensive map and process flow. This practice is very similar to Value Stream Mapping in product development and hardware assembly. It is critical, as mentioned in Mr. Riedel's Senate Testimony in 2017[1],that VSM implementations use the strongest identity standards and trusted data utilizing the latest cryptographic standards to assure immutability, trust and transparency.

This creates an easy to decipher, visual representation of the software development lifecycle that is digestible across an organization, including the C-suite. More importantly, it provides transparency into how internal or external software is developed that translates to more secure and resilient software. It provides tracking and accountability of all actions. In addition, it enables immutable evidence for forensic examination after-the-fact, and to identify other software touched by a suspicious identity. All of this adds up to deterrence against malicious behavior and decreased effects by sophisticated adversaries. Management also receives extensive benefits, as this process will reveal inefficiencies, pain points, and errors in a product lifecycle.

By linking Value Stream Mapping and Secure Software Development it is possible to build future alignment within the enterprise between delivery and security. This alignment allows companies to consolidate budgets and creates an incentive model that aligns with the productivity of the organization while building in stronger security standards. In collaboration with NIST, Copado could bring this to fruition and significantly help in the creation and adoption of stronger security standards.

[1] https://www.energy.senate.gov/services/files/33E20741-F24D-4F26-B430-29AC585A9FC3

Software Bill of Materials (SBOM)

SBOMs are required for any legacy software and future software additions. SBOMs will be a significantly large set of data that will be required to be assembled by DevSecOps best practices. Additionally SBOMs are key components to be included in Value Stream Mapping, and tied to an immutable record linked to the identity of the author. SBOMs give transparency for all software assets included in the software supply chain. It is critical that SBOM's be implemented with DevSecOps best practices to maintain integrity while not creating a costly burden on the software maker. Properly implemented SBOM's create the ability to effectively manage SBOMs at enterprise scale. While numerous technical details will be sorted in the coming months, all decisions need to map back to the top line goal of identity, integrity, and transparency.

There are two critical areas of SBOM implementation that will maximize transparency and value. Collaboration with NIST would allow for further advancements on these important topics.

- **Legal, Compliance, and Governance:** SBOMs tackle unique challenges in understanding the provenance of code. An engineer can link to code taken from other sources and the true origin of code can become lost creating attribution and compliance issues. SBOMs create the potential of automating approval or rejection of code from restricted sources.

- **Dependencies:** SBOMs illuminate tangled webs of dependencies. An engineer could have little insight regarding the impacts of slight changes to code. The value in applying mapping and process graphing not only allows for the creation of the SBOM, but also eliminates barriers to transparency.

DevSecOps For Low Code/NoCode Environments

DevSecOps has matured significantly for traditional development environments, but best practices for Low Code/No Code are at their infancy. Low Code/No Code can be more dangerous than more traditional software development environments because of the assumptions of the personas developing within them. This risk, and the growing market of Low Code/No Code software environments, requires a new set of DevSecOps best practices be built. Low Code environments will still need rigorous testing and validation of proper controls to keep data stored in those environments safe.These practices need to set standards for identity management, role management, and testing standards for compliance and security.

NIST should consider these frameworks to deliver a holistic implementation of identity, integrity, and transparency into enterprise software development. Copado looks forward to collaborating with NIST to advance these important topics.

New Context, recently acquired by Copado, has been working on secure software development and DevSecOps since 2013. Copado has worked with the Department of Energy and the California Public Utilities Commission to bolster national cybersecurity standards Structured Threat Intelligence eXpression (STIX)[2], OpenC2[3] and CACAO[4] to secure the United States of America.

---

[2]https://www.dhs.gov/blog/2015/07/23/dhs-leads-effort-transition-automated-cybersecurity-information-sharing

[3] https://openc2.org/

[4] https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=cacao