Source Code Integrity and Code Signing

Source code integrity is a fundamental aspect of DevSecOps [1], ensuring that software remains unaltered and originates from a trusted source. Code signing is a cryptographic technique that affirms the authenticity and integrity of code by attaching a digital signature, thereby preventing unauthorized modifications and enhancing trust in software distribution.

Incorporating code signing within DevSecOps pipelines is crucial for several reasons. It safeguards against supply chain attacks by verifying that code has not been tampered with during development or distribution. This verification is vital in today's landscape, where threats like the SolarWinds incident have highlighted vulnerabilities in software supply chains [2]. By embedding code signing into Continuous Integration/Continuous Deployment [CI/CD] processes, organizations can automate security checks, ensuring that only verified code progresses through the development lifecycle.

Several tools facilitate the implementation of code signing in DevSecOps:

* Keyfactor Code Assure: Offers secure code signing with policy-based controls, ensuring that only authorized code is signed and deployed.
* PrimeKey SignServer: Provides a centralized solution for code signing, supporting various formats and integrating with existing infrastructure.
* Sigstore: An open-source project that simplifies code signing for developers by automating the process and maintaining a public, immutable log of signatures, enhancing transparency and trust [3].
* Microsoft SignTool: A command-line utility for signing Windows applications, integrating seamlessly into Windows-based development workflows.
* jarsigner: A tool for signing Java Archive (JAR) files, ensuring the authenticity of Java applications.

Implementing these tools within DevSecOps practices not only fortifies the software development process against tampering and unauthorized changes but also fosters a culture of security and trust. By ensuring that every piece of code is signed and verified, organizations can confidently deliver secure and reliable software to end-users.

References:

1. Encryption Consulting. (2025). *How Code Signing Aligns With DevSecOps*. Retrieved from <https://www.encryptionconsulting.com/how-code-signing-aligns-with-devsecops/>
2. Certauri. (2023). *Discover Top Code Signing Tools for Secure Software in 2024*. Retrieved from <https://www.certauri.com/discover-top-code-signing-tools-for-secure-software/>
3. Wired. (2022). *GitHub Moves to Guard Open Source Against Supply Chain Attacks*. Retrieved from https://www.wired.com/story/github-code-signing-sigstore