SAFECode Comments on Secure Software Development Attestation Common Form

Comments to the Cybersecurity and Infrastructure Security Agency (CISA)
ICR Reference Number 202311-1670-001

Steven B. Lipner
Executive Director, SAFECode
15 December 2023

SAFECode has reviewed the November draft of CISA's Secure Software Development Attestation Common Form. We observed that there are very few changes between the April 2023 draft and the November draft.

We are particularly concerned that the selection of attestation requirements from the NIST Secure Software Development Framework continues to overemphasize code integrity and underemphasize secure development (SDL) and that several important secure development requirements are still omitted by Attestation Requirement 4 in the Appendix. In making this choice, CISA foregoes and opportunity to operationalize its emphasis on encouraging vendors to develop products that are "Secure by Design." We have attached our comments on the original (April 2023) draft since they are still applicable.

SAFECode Comments on Secure Software Development Attestation Form and Secure Software Development Attestation Form Instructions

Comments to the Cybersecurity and Infrastructure Security Agency (CISA)
Docket # CISA–2023–0001

Steven B. Lipner
Executive Director, SAFECode
22 June 2023

GENERAL COMMENTS

Introduction

SAFECode appreciates the opportunity to comment on CISA's Secure Software Development Attestation Form, which will be used by industry to provide the federal government with assurance that software used by agencies has been securely developed in alignment with Section 4 of Executive Order 14028 (EO).

SAFECode is a nonprofit industry organization that provides a global industry forum where business leaders and technical experts come together to exchange insights and ideas on creating, improving, and promoting scalable and effective software security programs. We believe that secure software development can only be achieved with an organizational commitment to the execution of a holistic assurance process, and that sharing information on that process, as well as the practices it encompasses, is the most effective way for software providers to help customers and other stakeholders manage software security risk.

SAFECode has a long history of advocating for Secure Development Lifecycle (SDL) and Code Integrity best practices within industry. We participated in the early industry-government working sessions after the EO was issued, and we have been involved with the development of the NIST Secure Software Development Framework (SSDF) beginning with co-leadership of a working session at the RSA Conference in 2018 and continuing by providing input to both the original and updated versions of NIST SP 800-218.

We are strong supporters of the government's decision to rely on vendor attestation to assure the adoption of secure software development and supply chain practices. The wide variety of software technologies, tools, and development processes in use across the industry means that each vendor must create its own secure development process. Providing vendors with high-level guidance, as specified in the SSDF, and then requiring that they attest to adopting that guidance as appropriate is much more practical and cost-effective for both government and vendors than attempting to create a more detailed or prescriptive standard and/or a common third-party evaluation process.

Balancing the Requirements

Section 4 of the EO seeks to accomplish three objectives:

1. SDL: to assure that vendors apply secure development processes as they create software products and services (the SDL requirements),

2. SBOM: to assure that vendors apply appropriate due diligence in selecting and supporting third-party software components that they integrate (the supply chain and Software Bill of Materials – SBOM – requirements), and
3. Code Integrity: to assure that vendors protect their software development and release systems from malicious attacks that could undermine the integrity of software under development (the Code Integrity requirements).

In reviewing the Attestation Form, we are concerned that there is an overemphasis on the Code Integrity requirements and an underemphasis on SDL requirements. SAFECode believes that the SSDF strikes the right balance among the three domains (SDL, SBOM, Code Integrity) identified in the EO. In the experience of SAFECode and its members, vulnerable software code contributes to a far greater percentage of cyber-attacks than attacks on code repositories or build systems.

We believe it is critical to better balance the focus of the attestation. Secure Development goes far beyond running vulnerability scanning tools. SSDF practices and tasks that are critical to achieving "security by design" such as threat modeling and root cause analysis of discovered vulnerabilities were omitted from the attestation. The SSDF does a much better job than the Attestation Form of balancing the three objectives of Section 4 of the EO and of representing modern software security practices. Given that the SSDF was positioned as the primary vehicle for attesting to the requirements of Section 4 of the EO in NIST's earlier guidance, we believe it should be more comprehensively represented in the Attestation Form.[1]

Code Integrity Practices: Mandates or Examples

With regard to the Attestation Form's treatment of Code Integrity requirements, we were surprised to see that Requirement 1 of the form asks vendors to attest to a series of minimum practices pertaining to development environments. The practices listed appear in paragraph 4 (e) (1) of the EO as examples ("…such as…") while the Attestation form appears to mandate those specific practices. The SSDF, similar to the EO, lists specific practices as examples. We believe that the Attestation Form should align with the approach in the EO and the SSDF, and thus allow vendors to take a risk-based approach in selecting specific controls and practices appropriate to their development systems and practices.

Agency Requests for Evidence and Artifacts

Our final area of general concern is the ambiguity around supporting artifacts that agencies may require beyond the Attestation Form and the Software Bill of Materials (SBOM). The paragraph on "Additional Information" in the Attestation Form opens up the possibility that each agency can demand information of its own choosing beyond the Attestation Form and SBOM. This lack of uniformity can lead to considerable cost and effort on the part of vendors as they seek to meet the differing requirements of numerous agencies. Such cost and effort would be better dedicated to creating and delivering secure software. Predictability of compliance artifact requirements will be essential to developer organizations' ability to comply with the attestation requirement at scale.

SAFECode also believes that agencies should not request artifacts that contain vendor intellectual property or that include sensitive information that could be used to facilitate attacks. Given the diverse

[1] Software Supply Chain Security Guidance Under Executive Order (EO) 14028 Section 4e (2022), https://www.nist.gov/system/files/documents/2022/02/04/software-supply-chain-security-guidance-under-EO-14028-section-4e.pdf

nature of modern software development processes and technology, SAFECode believes that suppliers should have some authority to determine which low-level artifacts they share and in what format, so as to prevent IP theft or leaking of sensitive security information. Furthermore, to minimize the compliance burden being placed on suppliers beyond the attestation and SBOM, additional clarity is needed around a process for sharing these low-level artifacts in a seamless and confidential manner.

SPECIFIC COMMENTS

- The table of attestation requirements on pages 4-6 fails to provide guidance for the case of partial compliance where a requirement cites an SSDF practice that includes multiple tasks. Must at least one or must all SSDF tasks of such a practice be implemented to claim compliance? We have assumed "all" in our review of the table.
- Requirement #1: As discussed in the general comments, specifying requirements for the development environment "at a minimum" is inconsistent with the use of "such as" in the EO and with the use of "Notional Implementation Examples" in the SSDF and may lead to ineffective or overly costly measures in specific cases. We recommend providing suppliers with flexibility to manage risk and make appropriate choices of controls in meeting the intent of the requirement by replacing the second sentence with "Those environments were secured by actions whose effectiveness is comparable to the following actions."
- Requirement #1a: The definition of this action is unclear. Please clarify whether the intent of the segmentation requirement is focused on separating development and test environments from production environments or whether it is intended to require segmentation of development environments to limit the scope and impact of potential intrusions or insider malfeasance.
- Requirement #1e: The definition of this action is unclear. The encryption of credentials is a best practice, but encryption of source code (at rest or in transit) may also be a best practice or may overburden organizations that have other controls in place, depending on the development environment and risk. Please clarify.
- Requirement #4: As discussed in the general comments, a number of SSDF tasks and practices that are important to the creation of software that is "secure by design" are omitted from the draft attestation form. We recommend the inclusion of the following practices and tasks from the SSDF to ensure that attestation reflects the vendor's adoption of an effective secure software development process:
    - "Defining security requirements" for software development to ensure that those requirements are addressed throughout the SDLC (PO.1. and PO.1.2)
    - "Implement roles and responsibilities" to assure that the development organization is committed and prepared to deliver on its responsibility for creating secure software. (PO.2)
    - "Implement supporting toolchains" to improve the quality, accuracy and security of software throughout the SDLC (PO.3). Note that while tasks from PO.3 are listed in requirements 1f, 2, and 3, an effective secure development toolchain is critically important to the creation of software that is free from vulnerabilities. Thus PO.3 (or its subordinate tasks) should also be listed for requirement 4.
    - "Design software to meet security requirements and mitigate security risks" (PW.1) The tasks of practice PW.1 are fundamental to the creation of software that is secure by

design. SAFECode considers this change to the attestation form to be especially important.

  - "Reuse existing, well-secured software when feasible instead of duplicating functionality" to reduce development cost and minimize the introduction of new vulnerabilities Although the attestation form addresses the use of secure third-party components as detailed in tasks PW.4.1 and PW.4.4, it omits the requirement for the vendor to create and manage well-secured components in-house. The availability of such components is important to an efficient and effective secure software development process. (PW.4.2)
  - "Analyze vulnerabilities to identify their root causes" to enable continuous improvement by identifying and improving SDL issues that can prevent future similar vulnerabilities. (RV.3) In the experience of SAFECode members, analysis of the root causes of discovered vulnerabilities is fundamental to creating a secure development process that incorporates continuous improvement and effectively addresses emerging threats.

- Section III – attestation and Signature – is inconsistent with the Minimum Attestation References table. Item 3 in Section III is a duplicate of Item 2a in Section III and should be removed to make the organization of Section III consistent with the table.