KPMG

LINEAJE

Securing your digital supply chain ecosystem



October 2024

kpmg.com/in

KPMG. Make the Difference.

Securing digital supply chain is key to operational resilience

The interconnected nature of the digital supply chain ecosystem and increasing reliance on technology, evidenced in the CrowdStrike incident, have underscored the critical need for operational resilience in our digital supply chain. The occurrence of just one unexpected event or update has the potential to trigger a chain reaction across the entire digital world ecosystem, emphasizing the importance of being prepared for any possible repercussions.

In this context, securing digital assets and associated supply chain has become vital for the government, regulators, senior management, and board due to:

- **Increasing software supply chain attacks and changing attack patterns**
- **Reliance on interconnected systems and third-party software components**
- **Increasing dependency on software products to ensure resiliency of critical business processes**

Recognising the growing risk, governments and regulatory authorities across the globe have released requirements to secure the software supply chain pipeline. One such requirement from the US federal government is self-attestation released under OMB memorandum- M-22-18[1] in September 2022, wherein software producers need to formally declare their adherence to secure software development practices.

Following the US federal government requirement, CISA[2] published the self-attestation requirements on 28 April 2023, highlighting some of the key asks from software publishers where they had to submit self-attestation form by 12 June 2023, for all the critical software and on/by 14 September 2023, for all the other software.

This paper delves into the requirements for self-attestation, key focus areas, and the role of technology in implementing mature software supply chain security practices. It primarily guides software publishers on implementing robust controls to secure their products and supply chain, ensuring customer trust. Additionally, it empowers software consumers to assess third-party software and components before onboarding.



1. OMB memorandum- M-22-18: Office of Management and Budget memorandum – M-22-18
2. CISA: Certified Information Systems Auditor

© 2024 KPMG Assurance and Consulting Services LLP, an Indian Limited Liability Partnership and a member firm of the KPMG global organization of independent member firms affiliated with KPMG International Limited, a private English company limited by guarantee. All rights reserved.

Software self-attestation and its coverage

Software self-attestation is the process of attesting that the software publisher has considered and enforced security practices while developing software products and components such as secure development practices and environment, maintenance of provenance data for internal and third-party code, good-faith effort to maintain trusted source code supply chain, vulnerability management, etc.

The initial focus of the software publisher is to attest and submit a signed self-attestation (by CEO or assigned designee / 3PAOs[3]) form for a software product(s) and components supplied to the US government to meet EO-14028[4] requirements (OMB memorandum – M-22-18(September 2022)).

Which software is covered under EO-14028 self-attestation?

- Software developed after 14 September 2022
- Software with major version changes after 14 September 2022 (e.g., Version 2 to Version 3)
- Software receiving continuous updates (e.g., software-as-a-service)
- Software products and components developed by federal agencies and software freely obtained (e.g., freeware and open-source software) directly by a federal agency are not in scope.



3. 3PAO: Third-Party Assessment Organisations
4. EO14028: Executive Order 14028

© 2024 KPMG Assurance and Consulting Services LLP, an Indian Limited Liability Partnership and a member firm of the KPMG global organization of independent member firms affiliated with KPMG International Limited, a private English company limited by guarantee. All rights reserved.

Top 3 SSCS implementation challenges

01. Limited software supply chain visibility:

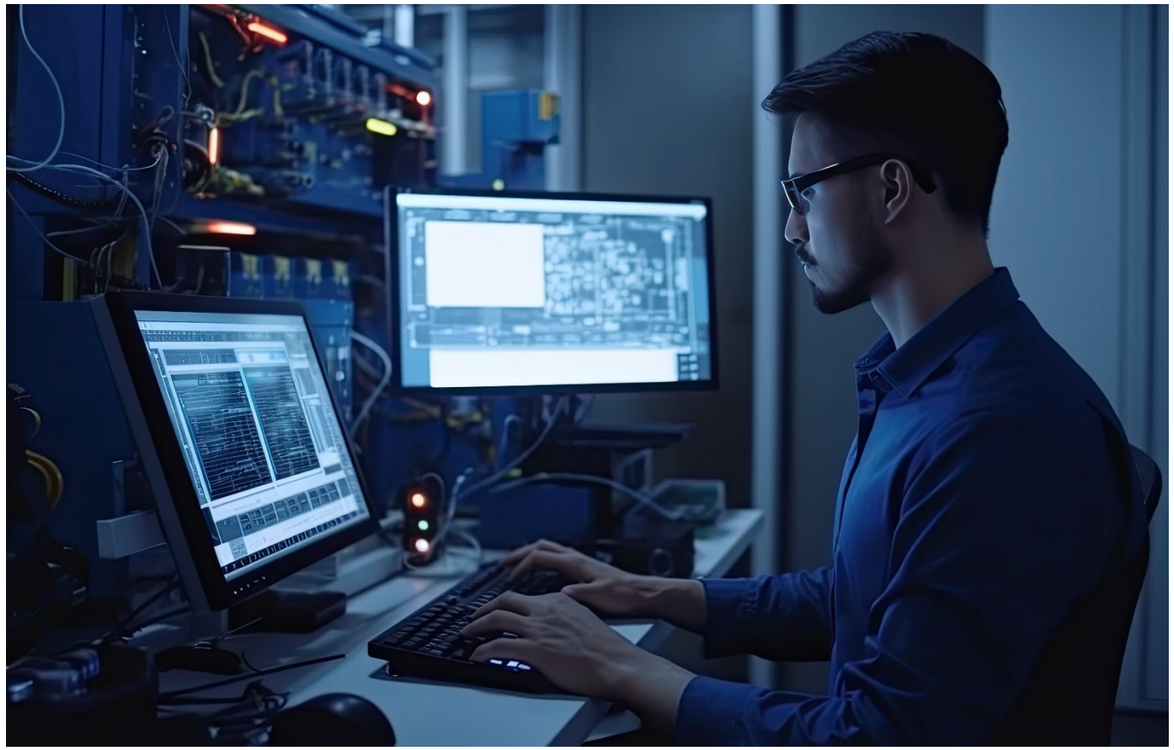
One of the major threats to the software supply chain is the trust organisations place in the Open-source Software (OSS) and Third-party Software Systems (TPSS) without having visibility on software components' authenticity and source of origin while developing software products.

02. Lack of standardized implementation guidance:

Organisations need to decide on steps required to be taken to demonstrate compliance with self-attestation requirements. Since this is an emerging area, industry-wide standards on how to implement specific self-attestation requirements are not present.

03. Evidence collection and maintenance:

Collection and maintenance of comprehensive self-attestation evidence from disparate systems, environments, and associated supply chains becomes a tedious and difficult task due to the absence of respective tools and platforms.



© 2024 KPMG Assurance and Consulting Services LLP, an Indian Limited Liability Partnership and a member firm of the KPMG global organization of independent member firms affiliated with KPMG International Limited, a private English company limited by guarantee. All rights reserved.

Practical guidance to demystify SSCS expectations

Self-Attestation areas	Description	Key expectations
1. Secure development and build environment	**1(a) Segregated environment:** Separate and secure environments leveraged in developing and building software products and components.	1. Establish physical/ logical network segmentation 2. Implement zero trust architecture 3. Secure configuration for tools in each environment 4. Establish one or more software repositories to host sanctioned and vetted open-source components with ability to attest authenticity and inalterability from source.
	1(b) Access controls: Access to development and build environments is strictly controlled and monitored using multi-factor authentication.	1. Restrict connections for entering and exiting each software development environment using a strong authentication mechanism 2. Minimize direct human access to toolchain systems 3. Configure each development endpoint to provide the least functionality needed by users and services 4. Configure multi-factor authentication for all environments and components within the environment.
	1(c) Logging and monitoring: Continuous logging, monitoring, and audit trust relationship used for authorized access to development and build environments, and associated components.	1. Identify events across components/ environments to be logged including: a. privileged account activities b. component configuration changes c. anomalous events (ex. Shutdown 2. Establish a mechanism for performing periodic reviews and act on deviations identified, if any 3. Perform periodic audits of access and authorization mechanisms for each (including between components) development and build environments.
	1(d) Minimizing usage of risky software: Steps are taken to avoid using software that poses security threats within development environments.	Implement process for acquisition of new software products (used in any of the environments) including: a. business case review and approval b. security risk assessment and testing.
	1(e) Data protection: Ensuring sensitive data is protected at rest using encryption wherever possible.	Enable FIPS-compliant encryption of all sensitive data at rest and in transit.
	1(f) Cybersecurity practices: Continuous monitoring and response measures are in place to identify and address security incidents.	1. Use code-based configuration for toolchains (e.g., pipelines-as-code, toolchains-as-code) 2. Continuously monitor tools present in the environment and tool logs for potential operational and security issues, including policy violations and anomalous behavior 3. Continuously monitor all software deployed in each environment for new vulnerabilities and respond to vulnerabilities 4. Establish a security incident management program 5. Establish a remediation management framework to log, track, monitor, and close the identified vulnerability with tools and software components.

© 2024 KPMG Assurance and Consulting Services LLP, an Indian Limited Liability Partnership and a member firm of the KPMG global organization of independent member firms affiliated with KPMG International Limited, a private English company limited by guarantee. All rights reserved.

Practical guidance to demystify SSCS expectations

Self-Attestation areas	Description	Key expectations
2. Trusted source code supply chains	Use automation wherever possible to maintain trusted source code supply chains and have a process to address the identified risks associated with internal and third-party components and manage related vulnerabilities.	1. Deploy and configure the technology platform to: a. Establish policy defining trust levels based on parameters such as source of origin, version, signature, and package URL verified b. Generate and maintain SBOM[5] with Nth-level dependencies c. For each software component, establish a process to identify and mitigate (if needed) violations to the trust levels 2. For third-party or open-source components, configure a tool for assessing and managing existing vulnerabilities: a. Generate alerts for known vulnerable components used in the software product(s) b. Maintain the version history of the repository to track all changes made to the code with accountability to the individual account c. Store all source code and configuration-as-code in a code repository and restrict access to it based on the nature of the code 3. For internal code: a. Store all source code and configuration-as-code in a code repository and restrict access to it based on the nature of the code b. Perform SAST and DAST on a periodic basis to identify vulnerabilities 4. Define and enforce policy for securing software development infrastructures and their components, including development endpoints, throughout the software supply chain pipeline 5. Establish and enforce security practices such as: a. Commit signing for code repositories b. Have the code owner review and approve all changes made to the code by others c. Use code signing to help protect the integrity of executables d. Use cryptography (e.g., cryptographic hashes) to help protect file integrity 6. Establish a remediation management framework to log, track, monitor, and close the identified gap(s) associated with software supply chain environment(s) 7. Securely archive the necessary files and supporting data (e.g., integrity verification information, provenance data) to be retained for each software release) 8. Educate affected individuals on impending changes to requirements.

5. SBOM: Software Bill of Materials

© 2024 KPMG Assurance and Consulting Services LLP, an Indian Limited Liability Partnership and a member firm of the KPMG global organization of independent member firms affiliated with KPMG International Limited, a private English company limited by guarantee. All rights reserved.

Practical guidance to demystify SSCS expectations

Self-Attestation areas	Description	Key expectations
3. Software provenance	Maintain provenance data for internal and third-party code and components to the greatest extent feasible.	1. Generate, maintain, secure, and share provenance data for all components of each software release in accordance with the organisation's policies, preferably using standards-based formats. (e.g., in a software bill of materials [SBOM]) 2. Collect provenance information for third-party software components (e.g., SBOM, source composition analysis, binary software composition analysis), and analyze and assess the associated risk 3. Maintain a list of organisation-approved commercial software components and component versions along with their provenance data 4. Build binaries from the source code after verifying the source code's integrity and provenance in case the integrity or provenance of acquired binaries is unknown 5. Store the release files, associated images, etc. in repositories and allow read-only access to authorized personnel 6. Store and protect release integrity verification information and provenance data by keeping it in a separate location from the release files or by signing the data 7. Implement an automation tool to review provenance and software composition data for all software components to identify any new vulnerabilities (if any) 8. Define security contractual clauses for third-party software products and incorporate them into the standard contract template; the criteria can include the third party's vulnerability disclosure program and product security incident response capabilities, SBOM sharing, VDR/VEX report, self-attestation report, etc. 9. Define and enforce the exception process to perform periodic reviews of all non-compliance cases to the defined SSCS policy/practices 10. Establish and enforce software acquisition process covering: a. Third-party and open-source components b. Acquisition process, security requirements, risk assessment requirement, contractual requirements including requirements such as SBOM, provenance information, vulnerability disclosure, self-attestation, etc.

© 2024 KPMG Assurance and Consulting Services LLP, an Indian Limited Liability Partnership and a member firm of the KPMG global organization of independent member firms affiliated with KPMG International Limited, a private English company limited by guarantee. All rights reserved.

Practical guidance to demystify SSCS expectations

Self-Attestation areas	Description	Key expectations
4. Vulnerability scanning and response	**4(a) Automated checks:** Use automated tools to scan security vulnerabilities in the software on an ongoing basis and, at minimum, prior to product, version, or update release. **4(b) Vulnerability management policy:** Establish a robust vulnerability management process to address discovered vulnerabilities before releasing software. **4(c) Vulnerability disclosure program:** Establish and enforce a vulnerability disclosure program to promptly accept, review, and mitigate disclosed software vulnerabilities in a timely fashion.	1. Define and enforce criteria for software security checks and track throughout the SDLC including key performance indicators (KPIs), key risk indicators (KRIs), vulnerability severity scores, review of the artifacts generated, and analyze collected data in the context of the security successes and failures 2. Deploy skilled resource(s) persons who were not involved with the design and/or automated processes instantiated in the toolchain review the software design to confirm and enforce that it meets all of the security requirements and satisfactorily addresses the identified risk information 3. Establish a mechanism to perform regular security checks for the acquired commercial, open-source, and all other third-party software components throughout their life cycles 4. Conduct the code review and/or code analysis based on the organisation's secure coding standards, and record and triage all discovered issues and recommended remediations in the development team's workflow or issue tracking system 5. Establish and enforce software security testing approach including the scope of the testing, design the tests, perform the testing, and document the results, recording and triaging all discovered issues and recommended remediations in the development team's workflow or issue tracking system 6. Define and implement software secure baseline, and conduct testing to ensure that the settings, including the default settings, are working as expected 7. Implement and configure the toolchain to perform automated code analysis and testing on a regular or continuous basis for all supported releases to identify or confirm the presence of previously undetected vulnerabilities 8. Establish and enforce a vulnerability disclosure program including: a. Vulnerability identification and remediation b. Product Security Incident Response Team (PSIRT) and processes to handle the responses to vulnerability reports and incidents, including communications plans for all stakeholders 9. Establish and implement a robust vulnerability management program including: a. Review the software for similar vulnerabilities to eradicate a class of vulnerabilities, and proactively fix them rather than waiting for external reports b. Remediation management process and tool to log, track, monitor, and close the identified vulnerability with software product and gap(s) associated with supply chain environment(s) 10. Establish and enforce security incident management process including periodic review and testing.

© 2024 KPMG Assurance and Consulting Services LLP, an Indian Limited Liability Partnership and a member firm of the KPMG global organization of independent member firms affiliated with KPMG International Limited, a private English company limited by guarantee. All rights reserved.

Role of technology in SSCS

Most organisations have already automated their software development pipeline and associated infrastructure to streamline development processes. In the context of SSCS, organisations may want to consider the below functionalities as they evaluate options to automate and streamline compliance with self-attestation requirements:



Centralized data repository for version-level content:

Periodically collect, update, and maintain a centralized inventory for each software product and component release including software composition analysis results, vulnerability scan reports, provenance data (SBOMs, VEX, VDR, CSAF[6] reports), and other supporting SSCS artifacts.



Vulnerability management (aside from SAST/ DAST):

Regularly monitor, detect, assess, and mitigate newly discovered vulnerabilities in any internal, third-party, and open-source software components.



Automate workflow:

Automate remediation workflow to assign tasks to the developer team (including third-party/partner) along with deadlines to fix the identified vulnerabilities in any internal, third-party, and open-source software components.



Analyze and maintain provenance data:

Analyze, monitor, and maintain the origin of internal, third-party, and open-source software code and components along with associated supply chain.



Streamlined reporting:

Generate and provide an overall security compliance posture view of in-scope software products and demonstrate a clear alignment between organisational and industry security practices, and regulatory requirements.

6. CSAF: Common Security Advisory Framework

© 2024 KPMG Assurance and Consulting Services LLP, an Indian Limited Liability Partnership and a member firm of the KPMG global organization of independent member firms affiliated with KPMG International Limited, a private English company limited by guarantee. All rights reserved.

We can help design, automate and execute SSCS program for you!

Our team of in-house trained professionals specializes in design, execution, and automation of SSCS programs helping organizations effectively manage risks across the entire lifecycle of software products and associated infrastructure. Moreover, we've partnered with leading industry players to enhance our solutions with advanced technology platforms.

When it comes to self-attestation requirements, we help organisations prepare and align with industry-leading practices and regulatory requirements.

We leverage Lineaje (our alliance partner) technology platform to generate and maintain provenance & SBOM for selected software products and components. Lineaje platform helps verify and attest integrity and authenticity of the software code and components including direct and transitive dependencies at the Nth level and generates assurance level scores for each component. Further, it helps to scan and generate alerts to any new and/or existing identified vulnerabilities of the software components on a near real-time basis and allows for improved remediation time when new vulnerabilities are disclosed.

KPMG in India along with Lineaje, can help with the below areas in your Software Supply Chain Security Program journey:

1. **Readiness assessment:** Review existing SSCS practices against self-attestation requirements, identify areas of improvement, and develop a plan of action and milestones (PAO&M) to mitigate the identified gap(s).
2. **Design SSCS policy:** Design SSCS policy in alignment with industry-leading practices and regulatory requirements to secure software products and components (internal, third party, and open-source components) and associated infrastructure supported by automated tool workflow, appropriate governance and oversight, clear roles & responsibility (RACI), and communication channels.

Deployment and configuration of technology platforms: – Support in the deployment and implementation of technology platforms for:

a) **Software product and component provenance management:**

i. Generate and maintain provenance and SBOMs.

ii. Analyze SBOM: Assess each SBOM for potential risks including vulnerabilities, code quality, security posture, provenance, etc.; compliance violations, and tamper ability of the supply chain, based on the defined SSCS framework.

iii. Securely publish and share SBOM: Publish compliant & ATTESTED SBOMs for each product & SKU. Further, generate and share SBOM, VEX, or CSAF & other deployment mitigations to enable secure deployment of applications by customers.

b) **Software product vulnerability management:** Conduct vulnerability scan of in-scope software products and components leveraging Lineaje platform on a near real-time basis. Further, analyze the outcome of the vulnerability scan and support in developing and implementing the mitigation plan.

c) **Remediation management:** Log, track, monitor, and closure of the identified vulnerabilities and issues in software components (internal, third-party and open-source components) by working with identified stakeholders and partners.

Acknowledgements

- Pavitra Shah
- Priyal Khurana
- Nisha Fernandes
- Karthika Prabasankar



© 2024 KPMG Assurance and Consulting Services LLP, an Indian Limited Liability Partnership and a member firm of the KPMG global organization of independent member firms affiliated with KPMG International Limited, a private English company limited by guarantee. All rights reserved.

KPMG in India contacts:

Akhilesh Tuteja
Global Head
Cyber Security
E: atuteja@kpmg.com

Kunal Pande
National Co-Head - Digital Risk Security and Governance (DRSG)
E: kpande@kpmg.com

Srijit Menon
Partner
National Head for TPRM in India
E: srijitmenon@kpmg.com

Atul Gupta
Partner
Head – Digital Trust and Cyber
E: atulgupta@kpmg.com

Srinivas Potharaju
Partner and Head - Digital Risk and Cyber
E: srinivasbp@kpmg.com

Anil Singh
Associate Director
E: anilsingh1@kpmg.com

Lineaje contacts:

Javed Hasan
Co-founder and CEO
E: javed@lineaje.com

Anand Revashetti
Co-founder and CTO
E: anand@lineaje.com

kpmg.com/in

Follow us on:

kpmg.com/in/socialmedia

Access our latest insights on KPMG Insights Edge

The information contained herein is of a general nature and is not intended to address the circumstances of any particular individual or entity. Although we endeavor to provide accurate and timely information, there can be no guarantee that such information is accurate as of the date it is received or that it will continue to be accurate in the future. No one should act on such information without appropriate professional advice after a thorough examination of the particular situation.

KPMG Assurance and Consulting Services LLP, Lodha Excelus, Apollo Mills Compound, NM Joshi Marg, Mahalaxmi, Mumbai - 400 011 Phone: +91 22 3989 6000, Fax: +91 22 3983 6000.

© 2024 KPMG Assurance and Consulting Services LLP, an Indian Limited Liability Partnership and a member firm of the KPMG global organization of independent member firms affiliated with KPMG International Limited, a private English company limited by guarantee. All rights reserved.

The KPMG name and logo are trademarks used under license by the independent member firms of the KPMG global organization.

This document is for e-communication only. (017_FFLY0924_KP)