



National Aeronautics and
Space Administration
Mary W. Jackson NASA Headquarters
Washington, DC 20546-0001

MEMORANDUM FOR NASA CONTRACTOR AND GRANT/COOPERATIVE
AGREEMENT SOFTWARE PUBLISHER AND SUPPLIER COMMUNITY

FROM: Karla Smith Jackson
Assistant Administrator for Procurement
Senior Procurement Executive
Office of Procurement (OP)

Jeffrey Seaton
Chief Information Officer
Office of Chief Information Officer (OCIO)

SUBJECT: Update to January 2023 Supplier Documentation Requirements for Software
Producers Offering Third-Party Software to NASA for Purchase and/or Use

References:

- (a) Office of Management and Budget (OMB) Memorandum M-23-16 “Update to Memorandum M-22-18, Enhancing the Security of the Software Supply Chain through Secure Software Development Practices” dated June 9, 2023, [M-23-16 \(whitehouse.gov\)](#)
- (b) OMB Memorandum M-22-18 “Enhancing the Security of the Software Supply Chain through Software Development Practices” dated September 14, 2022, [M-22-18 \(whitehouse.gov\)](#)
- (c) Cybersecurity & Infrastructure Security Agency (CISA), Repository for Software Attestations and Artifacts (RSAA) [User Guide](#)

On January 23, 2023, NASA issued a memorandum (See Enclosure 1) to communicate requirements documented in OMB Memorandum M-22-18 directing Federal agencies to comply with the National Institute of Standards and Technology (NIST) Guidance when using third-party software on the agency’s information systems or otherwise affecting the agency’s information. M-23-16 reinforced the requirements established in M-22-18 and extended the timelines for agencies to collect attestations after the release of the CISA Secure Software Development Attestation Common Form (hereafter referred to as the “Common Form”) and approval by OMB under the Paperwork Reduction Act (PRA).

On March 8, 2024, OMB approved the release and use of the Common Form. On March 18, 2024, CISA's Repository for Software Attestations and Artifacts (RSAA) was released for use. OMB's approval of the Common Form established June 8, 2024 (critical software¹) and September 8, 2024 (all software) as the attestation collection due dates consistent with M-23-16.

On April 12, 2024, NASA received approval from OMB to use the Common Form (see Enclosure 2). This memorandum communicates NASA's process for ensuring software complies with OMB policies and provides collaboration opportunities for NASA vendor/suppliers to learn more on the process and what is required.

Process for Collecting & Using the CISA Repository

NASA's OCIO will ensure software requests comply with the OMB mandated secure software development practices prior to acquiring and/or utilizing the software in NASA's environment. Specifically, the following actions will be conducted to ensure compliance:

1. NASA will leverage RSAA to check if the software being requested or utilized has an existing attestation.
 - a. If an attestation, and, associated artifacts have already been posted in RSAA, NASA will not obtain a separate attestation and no further software publisher action is needed.
2. If there is no attestation in RSAA, NASA offerors, contractors, and vendors will need to collaborate with the software publisher to ensure the attestation and associated artifacts are uploaded in RSAA.
 - a. As outlined in NASA's January 2023 memorandum, Plans of Action & Milestones (POA&Ms) will be required to document the secure software development practice(s) that cannot be attested to **at this time**. At a minimum, the POA&M must document:
 - i. Common Form practice(s) that the software publisher cannot attest to.
 - ii. Compensating or mitigating practice(s).
 - iii. Date the Common Form practice(s) will be implemented.
 - b. If a software publisher will **never** be able to attest to Common Form practice(s), a waiver request must be submitted. At a minimum, the waiver request must document:
 - i. Common Form practice(s) that the software publisher will **never** be able to attest to.
 - ii. Compensating or mitigating practice(s).
 - iii. Risk Mitigation Strategy(ies)

¹ <https://www.nist.gov/itl/executive-order-improving-nations-cybersecurity/critical-software-definition-explanatory>

Timeline for Collection of Attestation Forms

As stated in the OMB Memorandum M-22-18, the requirements of the memorandum apply to agencies' use of software developed after the effective date of M-18-22 memorandum, as well as agencies' use of existing software that is modified by major version changes.

New Contracts. Effective immediately, for new contracts that include the use of software: NASA OCIO will require attestations to be uploaded to RSAA for new contracts (including micro-purchases), that include the use of software, regardless of whether such software use is considered critical.

Existing Contracts. Effective immediately, for existing software that is not free, open source, or government developed, software publishers should upload the required attestations to RSAA, regardless of whether such software use is considered critical.

Communication/Outreach - Software Publisher/Vendor/Supplier Collaboration Opportunities

In addition to the enclosed Frequently Asked Questions (FAQs), beginning Wednesday, July 17, 2024, NASA OCIO will host weekly, 11:30am – 1pm Eastern Standard Time (EST), collaboration opportunities to NASA software publishers, vendors, and/or suppliers for guidance and FAQ clarification. The following Microsoft Teams meeting information will be leveraged for the weekly collaborations.

[Join the meeting now](#)

Meeting ID: 239 463 088 16

Passcode: o69jDY

Dial in by phone

+1 256-715-9946

Phone conference ID: 960 804 580#

If there are questions, please contact Agency-DL-SoftwareAttestation@mail.nasa.gov
We appreciate your cooperation and continued commitment to the NASA mission.

For Karla Smith Jackson
Assistant Administrator for Procurement

Jeffrey M. Seaton
NASA Chief Information Office

cc:

Marvin Horne, Deputy Assistant Administrator for Procurement

Julia Wise, Director, Procurement Management and Policy
Division

Andre Sheppard, Acting Director, Procurement Strategic Operations
Division
Geoffrey Sage, Director, Enterprise Services and Analysis
Division

OP Procurement Officers (PO)/Deputy POs

Sean Gallagher, Deputy CIO for Operations

Patrick Newbold, Deputy CIO for Strategy

Mike Witt, Senior Agency Information Security Officer (SAISO)

Rob Binkley, Deputy SAISO

Marion Meissner, Chief, Privacy & Cybersecurity Risk Management

Gib Winter, Chief, Cybersecurity Service (CyS) Line

Dave Murnan, Cyber Protection (CP) Service Manager

Kanitra Tyler, CyS | CP Supply Chain Risk Management

Leigh Anne Giraldi, Chief, Enterprise Business Management Office (EBMO)

Louise Moroney, Deputy Chief, EBMO

Fran Beavers, Associate Chief, EBMO

Jaime Payne, Agency Software Manager, EBMO

Kay Twitchell, Deputy, Software License and Asset Management Office (SLAM), EBMO

Enclosures:

1. NASA January 2023 Supplier Documentation Requirements for Software Producers Offering Third-Party Software to NASA for Purchase and/or Use Memorandum
2. NASA-approved Self-Attestation Common Form
3. NASA Software Attestation FAQs

Enclosure 1

National Aeronautics and
Space Administration
Mary W. Jackson NASA Headquarters
Washington, DC 20546-0001



MEMORANDUM FOR NASA CONTRACTOR AND GRANT/COOPERATIVE AGREEMENT SOFTWARE SUPPLIER COMMUNITY

SUBJECT: Supplier Documentation Requirements for Software Producers Offering Third-Party Software to NASA for Purchase and/or Use

References:

- (a) Office of Management and Budget (OMB) Memorandum M-22-18 “Enhancing the Security of the Software Supply Chain through Software Development Practices” dated September 14, 2022, [M-22-18 \(whitehouse.gov\)](#)
- (b) OMB Memorandum M-21-30 “Protecting Critical Software Through Enhanced Software Measures” dated August 10, 2021, [M-21-30 \(whitehouse.gov\)](#)

The following is notification and communication of NASA’s process as required in references (a) and (b). Evidence of documentation is not required to be provided to NASA at this time. To ensure the integrity of its systems, and in compliance with requirements defined in Executive Order (EO) 14028, Improving the Nation’s Cybersecurity, and references (a) and (b), NASA requires all software producers to comply with the above requirements in order to offer software to NASA for purchase and/or use. Producers must implement and attest to conformity with secure software development practices as defined by the National Institute of Standards and Technology (NIST). As NASA further defines its processes for collecting these documents from its vendors, additional requirements may be implemented and communicated.

SECURITY OF SOFTWARE DEVELOPMENT REQUIREMENTS

1. **SCOPE.** As defined in the [Software Security Guidance Under Executive Order \(EO\) 14028 Section 4e \(nist.gov\)](#), these requirements apply to all software acquired and/or used by NASA, which includes firmware, operating systems, applications, and application services (e.g., cloud-based software, open-source software, as well as products containing software). This also includes software renewals and major version changes.
2. **ENSURE ALL SOFTWARE COMPLIES WITH SECURE SOFTWARE DEVELOPMENT PRACTICES RECOMMENDED BY NIST.** Federal agencies, including NASA, “must only use software provided by software producers who can attest to complying with the Government-specified secure software development practices, as described in the NIST Guidance.” This guidance references two documents, i.e., [The](#)

[NIST Secure Software Development Framework \(SSDF\), SP-800-218](#) and [the NIST Software Supply Chain Security Guidance](#).

3. **COMPLETE THE SELF-ATTESTATION CHECKLIST.** Software producers must implement and attest to conformity with secure software development practices by completing and submitting the self-attestation checklist. This required checklist, derived from NIST SP 800-218, Table 1: The Secure Software Development Framework (SSDF) Version 1.1, is attached to this memo (see Enclosure) and must be accompanied by a statement of conformity on company/entity letterhead. Submissions of non-public information shall be sent to Agency-ICT-SCRM@nasa.onmicrosoft.com with a copy to the cognizant Contracting Officer (CO) in accordance with the applicable deadline.
4. **A THIRD PARTY ASSESSMENT MAY BE ACCEPTED IN LIEU OF THE SELF-ATTESTATION CHECKLIST.** A third-party assessment from a certified FedRAMP Third-Party Assessor Organization (3PAO) or other NASA-approved third party may be accepted if an entity is unable to complete the self-attestation checklist. This information is posted on the [FedRAMP Marketplace](#).
5. **INABILITY TO ATTEST.** If a software producer cannot attest to one or more practices from NIST Guidance and the self-attestation form, all risks must be documented as well as the practices the producer has in place to mitigate those risks. Those documents, in addition to a Plan of Action & Milestones (POA&M) must be submitted to the cognizant NASA Contracting Officer (CO). The CO will coordinate with NASA's Office of the Chief Information Officer (OCIO) to make a risk-based determination on NASA's use of the software.
6. **NASA MAY REQUIRE ADDITIONAL DOCUMENTATION.** Based on the criticality of the software, NASA may require a Software Bill of Materials (SBOM). SBOMs must be generated in one of the data formats defined in [the National Telecommunications and Information \(NTIA\) report "The Minimum Elements for a Software Bill of Materials \(SBOM\)"](#) or successor guidance as published by the Cybersecurity and Infrastructure Security Agency (CISA).

The FAR Council has opened a proposed rule (FAR Case 2023-002, Supply Chain Software Security) to implement section 4(n) of Executive Order EO 14028. This rule will also focus on requirements outlined in OMB M-22-18. Status of FAR Cases are located at [rpt_OpenFARCasesWebReports \(osd.mil\)](#). Once the rule is finalized, relevant NASA acquisition policy may be updated to further implement the FAR rule and OMB guidance.

I appreciate your cooperation and continued commitment to the NASA mission. Please direct any further questions regarding this letter to your cognizant NASA CO.

Karla
Jackson

Digitally signed by Karla
Jackson
Date: 2023.01.18
14:20:18 -05'00'

Karla Smith Jackson
Assistant Administrator for Procurement

cc:

Jeffrey Seaton, Chief Information Officer

Marvin Horne, Deputy Assistant Administrator for Procurement

Julia Wise, Director, Procurement Management and Policy Division

Andre Sheppard, Acting Director, Procurement Strategic Operations Division

Geoffrey Sage, Director, Enterprise Services and Analysis Division

OP Procurement Officers (PO)/Deputy POs

Enclosure:

1. Checklist for Documenting Secure Software Development Activities in Support of EO 14028 Section 4e

Enclosure

Checklist for Documenting Secure Software Development Activities in Support of EO 14028 Section 4e

Practices	Tasks	Summary of Activities (including risk-based and mitigation actions in implementing the secure software development practices and tasks)	EO 14028 Subsections
Prepare the Organization (PO)			
Define Security Requirements for Software Development (PO.1): Ensure that security requirements for software development are known at all times so that they can be taken into account throughout the SDLC, and duplication of effort can be minimized because the requirements information can be collected once and shared. This includes requirements from internal sources (e.g., the organization’s policies, business objectives, and risk management strategy) and external sources (e.g., applicable laws and regulations).	PO.1.1: Identify and document all security requirements for the organization’s software development infrastructures and processes and maintain the requirements over time.		4e(ix)
	PO.1.2: Identify and document all security requirements for organization-developed software to meet and maintain the requirements over time.		4e(ix)
	PO.1.3: Communicate requirements to all third parties who will provide commercial software components to the organization for reuse by the organization’s own software. [Formerly PW.3.1]		4e(vi) 4e(ix)
Implement Roles and Responsibilities (PO.2): Ensure that everyone inside and outside of the organization involved in the SDLC is prepared to perform their SDLC-related roles and responsibilities throughout the SDLC.	PO.2.1: Create new roles and alter responsibilities for existing roles as needed to encompass all parts of the SDLC. Periodically review and maintain the defined roles and responsibilities, updating them as needed.		4e(ix)
	PO.2.2: Provide role-based training for all personnel with responsibilities that contribute to secure development. Periodically review personnel proficiency and role-based training, and update the training as needed.		4e(ix)
	PO.2.3: Obtain upper management or authorizing official commitment to secure development and convey that commitment to all with development-related roles and responsibilities.		4e(ix)
Implement Supporting Toolchains (PO.3): Use automation to reduce human effort and improve the accuracy, reproducibility, usability, and comprehensiveness of security practices throughout the SDLC, as well as provide a way to document and demonstrate the use of these practices. Toolchains and tools may be used at different levels of the organization, such as organization-wide or project-specific, and may address a particular part of the	PO.3.1: Specify which tools or tool types must or should be included in each toolchain to mitigate identified risks, as well as how the toolchain components are to be integrated with each other.		4e(iii) 4e(ix)
	PO.3.2: Follow recommended security practices to deploy, operate, and maintain tools and toolchains.		4e(i)(F) 4e(ii) 4e(iii) 4e(v)

Practices	Tasks	Summary of Activities (including risk-based and mitigation actions in implementing the secure software development practices and tasks)	EO 14028 Subsections
SDLC, like a build pipeline.			4e(vi) 4e(ix)
	PO.3.3: Configure tools to generate artifacts of their support of secure software development practices as defined by the organization.		4e(i)(F) 4e(ii) 4e(v) 4e(ix)
Define and Use Criteria for Software Security Checks (PO.4): Help ensure that the software resulting from the SDLC meets the organization’s expectations by defining and using criteria for checking the software’s security during development.	PO.4.1: Define criteria for software security checks and track throughout the SDLC.		4e(iv) 4e(v) 4e(ix)
	PO.4.2: Implement processes, mechanisms, etc. to gather and safeguard the necessary information in support of the criteria.		4e(iv) 4e(v) 4e(ix)
Implement and Maintain Secure Environments for Software Development (PO.5): Ensure that all components of the environments for software development are strongly protected from internal and external threats to prevent compromises of the environments or the software being developed or maintained within them. Examples of environments for software development include development, build, test, and distribution environments.	PO.5.1: Separate and protect each environment involved in software development.		4e(i)(A) 4e(i)(B) 4e(i)(C) 4e(i)(D) 4e(i)(F) 4e(ii) 4e(iii) 4e(v) 4e(vi) 4e(ix)
	PO.5.2: Secure and harden development endpoints (i.e., endpoints for software designers, developers, testers, builders, etc.) to perform development-related tasks using a risk-based approach.		4e(i)(C) 4e(i)(E) 4e(i)(F) 4e(ii) 4e(iii) 4e(v) 4e(vi) 4e(ix)

Practices	Tasks	Summary of Activities (including risk-based and mitigation actions in implementing the secure software development practices and tasks)	EO 14028 Subsections
Protect Software (PS)			
Protect All Forms of Code from Unauthorized Access and Tampering (PS.1): Help prevent unauthorized changes to code, both inadvertent and intentional, which could circumvent or negate the intended security characteristics of the software. For code that is not intended to be publicly accessible, this helps prevent theft of the software and may make it more difficult or time-consuming for attackers to find vulnerabilities in the software.	PS.1.1: Store all forms of code – including source code, executable code, and configuration-as-code – based on the principle of least privilege so that only authorized personnel, tools, services, etc. have access.		4e(iii) 4e(iv) 4e(ix)
Provide a Mechanism for Verifying Software Release Integrity (PS.2): Help software acquirers ensure that the software they acquire is legitimate and has not been tampered with.	PS.2.1: Make software integrity verification information available to software acquirers.		4e(iii) 4e(ix) 4e(x)
Archive and Protect Each Software Release (PS.3): Preserve software releases in order to help identify, analyze, and eliminate vulnerabilities discovered in the software after release.	PS.3.1: Securely archive the necessary files and supporting data (e.g., integrity verification information, provenance data) to be retained for each software release.		4e(iii) 4e(vi) 4e(ix) 4e(x)
	PS.3.2: Collect, safeguard, maintain, and share provenance data for all components of each software release (e.g., in a software bill of materials [SBOM]).		4e(vi) 4e(vii) 4e(ix) 4e(x)
Produce Well-Secured Software (PW)			
Design Software to Meet Security Requirements and Mitigate Security Risks (PW.1): Identify and evaluate the security requirements for the software; determine what security risks the software is likely to face during operation and how the software’s design and architecture should mitigate those risks; and justify any cases where risk-based analysis indicates that security requirements should be relaxed or waived. Addressing security requirements and risks during software design (secure by design) is key for improving software security and also helps improve development efficiency.	PW.1.1: Use forms of risk modeling – such as threat modeling, attack modeling, or attack surface mapping – to help assess the security risk for the software.		4e(ix)
	PW.1.2: Track and maintain the software’s security requirements, risks, and design decisions.		4e(v) 4e(ix)
	PW.1.3: Where appropriate, build in support for using standardized security features and services (e.g., enabling software to integrate with existing log management, identity management, access control, and vulnerability management systems) instead of creating proprietary implementations of security features and services. [Formerly PW.4.3]		4e(ix)

Practices	Tasks	Summary of Activities (including risk-based and mitigation actions in implementing the secure software development practices and tasks)	EO 14028 Subsections
<p>Review the Software Design to Verify Compliance with Security Requirements and Risk Information (PW.2): Help ensure that the software will meet the security requirements and satisfactorily address the identified risk information.</p>	<p>PW.2.1: Have 1) a qualified person (or people) who were not involved with the design and/or 2) 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.</p>		<p>4e(iv) 4e(v) 4e(ix)</p>
<p><i>Verify Third-Party Software Complies with Security Requirements (PW.3): Moved to PW.4</i></p>	<p><i>PW.3.1: Moved to PO.1.3</i> <i>PW.3.2: Moved to PW.4.4</i></p>		
<p>Reuse Existing, Well-Secured Software When Feasible Instead of Duplicating Functionality (PW.4): Lower the costs of software development, expedite software development, and decrease the likelihood of introducing additional security vulnerabilities into the software by reusing software modules and services that have already had their security posture checked. This is particularly important for software that implements security functionality, such as cryptographic modules and protocols.</p>	<p>PW.4.1: Acquire and maintain well-secured software components (e.g., software libraries, modules, middleware, frameworks) from commercial, open-source, and other third-party developers for use by the organization's software.</p>		<p>4e(iii) 4e(vi) 4e(ix) 4e(x)</p>
	<p>PW.4.2: Create and maintain well-secured software components in-house following SDLC processes to meet common internal software development needs that cannot be better met by third-party software components.</p>		<p>4e(ix)</p>
	<p><i>PW.4.3: Moved to PW.1.3</i></p>		
	<p>PW.4.4: Verify that acquired commercial, open-source, and all other third-party software components comply with the requirements, as defined by the organization, throughout their life cycles.</p>		<p>4e(iii) 4e(iv) 4e(vi) 4e(ix) 4e(x)</p>
	<p><i>PW.4.5: Moved to PW.4.1 and PW.4.4</i></p>		
<p>Create Source Code by Adhering to Secure Coding Practices (PW.5): Decrease the number of security vulnerabilities in the software and reduce costs by minimizing vulnerabilities introduced during source code creation that meet or exceed organization-defined vulnerability severity criteria.</p>	<p>PW.5.1: Follow all secure coding practices that are appropriate to the development languages and environment to meet the organization's requirements.</p>		<p>4e(iv) 4e(ix)</p>
	<p><i>PW.5.2: Moved to PW.5.1 as example</i></p>		
<p>Configure the Compilation, Interpreter, and Build Processes to Improve Executable Security (PW.6): Decrease the number of security vulnerabilities in the software and reduce costs by eliminating vulnerabilities before testing occurs.</p>	<p>PW.6.1: Use compiler, interpreter, and build tools that offer features to improve executable security.</p>		<p>4e(iv) 4e(ix)</p>
	<p>PW.6.2: Determine which compiler, interpreter, and build tool features should be used and how each should be configured, then implement and use the approved</p>		<p>4e(iv) 4e(ix)</p>

Practices	Tasks	Summary of Activities (including risk-based and mitigation actions in implementing the secure software development practices and tasks)	EO 14028 Subsections
	configurations.		
<p>Review and/or Analyze Human-Readable Code to Identify Vulnerabilities and Verify Compliance with Security Requirements (PW.7): Help identify vulnerabilities so that they can be corrected before the software is released to prevent exploitation. Using automated methods lowers the effort and resources needed to detect vulnerabilities. Human-readable code includes source code, scripts, and any other form of code that an organization deems human-readable.</p>	<p>PW.7.1: Determine whether code <i>review</i> (a person looks directly at the code to find issues) and/or code <i>analysis</i> (tools are used to find issues in code, either in a fully automated way or in conjunction with a person) should be used, as defined by the organization.</p>		4e(iv) 4e(ix)
	<p>PW.7.2: Perform the code review and/or code analysis based on the organization’s secure coding standards, and record and triage all discovered issues and recommended remediations in the development team’s workflow or issue tracking system.</p>		4e(iv) 4e(v) 4e(ix)
<p>Test Executable Code to Identify Vulnerabilities and Verify Compliance with Security Requirements (PW.8): Help identify vulnerabilities so that they can be corrected before the software is released in order to prevent exploitation. Using automated methods lowers the effort and resources needed to detect vulnerabilities and improves traceability and repeatability. Executable code includes binaries, directly executed bytecode and source code, and any other form of code that an organization deems executable.</p>	<p>PW.8.1: Determine whether executable code testing should be performed to find vulnerabilities not identified by previous reviews, analysis, or testing and, if so, which types of testing should be used.</p>		4e(ix)
	<p>PW.8.2: Scope the testing, design the tests, perform the testing, and document the results, including recording and triaging all discovered issues and recommended remediations in the development team’s workflow or issue tracking system.</p>		4e(iv) 4e(v) 4e(ix)
<p>Configure Software to Have Secure Settings by Default (PW.9): Help improve the security of the software at the time of installation to reduce the likelihood of the software being deployed with weak security settings, putting it at greater risk of compromise.</p>	<p>PW.9.1: Define a secure baseline by determining how to configure each setting that has an effect on security or a security-related setting so that the default settings are secure and do not weaken the security functions provided by the platform, network infrastructure, or services.</p>		4e(iv) 4e(ix)
	<p>PW.9.2: Implement the default settings (or groups of default settings, if applicable), and document each setting for software administrators.</p>		4e(iv) 4e(ix)
Respond to Vulnerabilities (RV)			
<p>Identify and Confirm Vulnerabilities on an Ongoing Basis (RV.1): Help ensure that vulnerabilities are identified more quickly so that they can be remediated more quickly in accordance with risk, reducing the window of opportunity for attackers.</p>	<p>RV.1.1: Gather information from software acquirers, users, and public sources on potential vulnerabilities in the software and third-party components that the software uses and investigate all credible reports.</p>		4e(iv) 4e(vi) 4e(viii) 4e(ix)
	<p>RV.1.2: Review, analyze, and/or test the software’s code to identify or confirm the presence of previously undetected</p>		4e(iv) 4e(vi)

Practices	Tasks	Summary of Activities (including risk-based and mitigation actions in implementing the secure software development practices and tasks)	EO 14028 Subsections
	vulnerabilities.		4e(viii) 4e(ix)
	RV.1.3: Have a policy that addresses vulnerability disclosure and remediation, and implement the roles, responsibilities, and processes needed to support that policy.		4e(viii) 4e(ix)
Assess, Prioritize, and Remediate Vulnerabilities (RV.2): Help ensure that vulnerabilities are remediated in accordance with risk to reduce the window of opportunity for attackers.	RV.2.1: Analyze each vulnerability to gather sufficient information about risk to plan its remediation or other risk response.		4e(iv) 4e(viii) 4e(ix)
	RV.2.2: Plan and implement risk responses for vulnerabilities.		4e(iv) 4e(v) 4e(viii) 4e(ix)
Analyze Vulnerabilities to Identify Their Root Causes (RV.3): Help reduce the frequency of vulnerabilities in the future.	RV.3.1: Analyze identified vulnerabilities to determine their root causes.		4e(ix)
	RV.3.2: Analyze the root causes over time to identify patterns, such as a particular secure coding practice not being followed consistently.		4e(ix)
	RV.3.3: Review the software for similar vulnerabilities to eradicate a class of vulnerabilities, and proactively fix them rather than waiting for external reports.		4e(iv) 4e(viii) 4e(ix)
	RV.3.4: Review the SDLC process and update it if appropriate to prevent (or reduce the likelihood of) the root cause recurring in updates to the software or in new software that is created.		4e(ix)

Enclosure 2

Department of Homeland Security

Cybersecurity and Infrastructure Security Agency (CISA)

Secure Software Development Attestation Form Instructions

Read all instructions before completing this form

Privacy Act Statement

Background: This information may be disclosed as generally permitted under Executive Order 14028, Improving the Nation’s Cybersecurity (E.O. 14028) and Memorandum M-22-18, “Enhancing the Security of the Software Supply Chain through Secure Software Development Practices” (M-22-18), as amended. This form collects contact information from vendor employees who make the attestation.

Authority: 44 U.S.C. § 3554, Executive Order (E.O.) 14028, “Improving the Nation’s Cybersecurity,” and OMB Memorandum M-22-18, “Enhancing the Security of the Software Supply Chain through Secure Software Development Practices,” as amended by OMB Memorandum M-23-16, “Update to Memorandum M-22-18, Enhancing the Security of the Software Supply Chain through Secure Software Development Practices,” authorize the collection of this information.

Purpose: The purpose of this form is to provide the Federal Government assurances that software used by agencies is securely developed.

Routine Uses: Use and disclosure of your records within and outside of NASA may occur in accordance with the NASA Security Records Systems (NASA 10SECR) system of records notice published at <https://www.federalregister.gov/documents/2023/08/08/2023-16900/privacy-act-of-1974-system-of-records> and as permitted by the Privacy Act of 1974, as amended (5 U.S.C. 552a(b)).

Disclosure: Failure to provide any of the information requested may result in the agency no longer utilizing the software at issue. Willfully providing false or misleading information may constitute a violation of 18 U.S.C. § 1001, a criminal statute.

What is the Purpose of Filling out this Form?

The Federal Information Security Modernization Act of 2014 (FISMA) requires each Federal agency to provide security protections for both “information collected or maintained by or on behalf of an agency” and for “information systems used or operated by an agency or by a contractor of an agency or other organization on behalf of an agency.” FISMA and other provisions of Federal law authorize the Director of the Office of Management and Budget (OMB) to promulgate information security standards for information security systems, including to ensure compliance with standards promulgated by the National Institute of Standards and Technology (NIST).

Executive Order 14028, “Improving the Nation’s Cybersecurity” (E.O. 14028), emphasizes the importance of securing software used by the Federal Government to perform its critical functions. To further this objective, E.O. 14028 required NIST to issue guidance “identifying practices that enhance the security of the software supply chain.”¹ The NIST Secure Software Development Framework (SSDF) (SP 800-218),² and the NIST Software Supply Chain Security Guidance³ (these two documents, taken together, are hereinafter referred to as “NIST Guidance”) include a set of practices that create the foundation for developing secure software.

E.O. 14028 further requires that the Director of OMB take appropriate steps to ensure that Federal agencies comply with NIST Guidance. To that end, OMB issued Memorandum M-22-18, “Enhancing the Security of the Software Supply Chain through Secure Software Development Practices” (M-22-18), on September 14, 2022. That memorandum was updated on June 9, 2023, through OMB Memorandum M-23-16, “Update to Memorandum M-22-18, Enhancing the Security of the Software Supply Chain through Secure Software Development Practices” (M-23-16). M-22-18, as amended by M-23-16, provides that a Federal agency may use software subject to M-22-18’s requirements only if the producer of that software has first attested to compliance with Federal Government-specified secure software development practices drawn from the SSDF.

This self-attestation form identifies the minimum secure software development requirements a software producer must meet, and attest to meeting, before software subject to the requirements of M-22-18 and M-23-16 may be used by Federal agencies. This form is used by software producers to attest that the software they produce is developed in conformity with specified secure software development practices.

Software requires self-attestation if any of the conditions is met:

1. The software was developed after September 14, 2022;
2. The software was developed prior to September 14, 2022, but was modified by major version changes (e.g., using a semantic versioning schema of Major.Minor.Patch, the software version number goes from 2.5 to 3.0) after September 14, 2022; or

¹ [Executive Order on Improving the Nation’s Cybersecurity \(E.O. 14028\), Section 4\(e\)](#).

² Available at: <https://csrc.nist.gov/Projects/ssdf>

³ Available at: <https://www.nist.gov/system/files/documents/2022/02/04/software-supply-chain-security-guidanceunder-EO-14028-section-4e.pdf>

3. The producer delivers continuous changes to the software code (as is the case for software-as-a-service products or other products using continuous delivery/continuous deployment).

Software products and components in the following categories are not in scope for M-22-18, as amended by M-23-16, and do not require a self-attestation:

1. Software developed by Federal agencies;
2. Open-source software that is freely and directly obtained by a Federal agency;
3. Third-party open source and proprietary components that are incorporated into the software end product used by the agency; or
4. Software that is freely obtained and publicly available.

Software producers who utilize third party components in their software are required to attest that they have taken specific steps, detailed in “Section III – Attestation and Signature” of the common form, to minimize the risks of relying on such components in their products.

Agency-specific instructions may be provided to the software producer outside of this common form. Conformance to agency-specific requirements may be included with this form as an addendum; agencies are responsible for fulfilling any Paperwork Reduction Act requirements applicable to agency-specific additions.

If a software producer is unable to submit via the online form, they may email a pdf version of the form to the respective agency:

Online Form Instructions:

- Selecting the provided URL: <https://softwaresecurity.cisa.gov>

OR

Local PDF Instructions:

- Saving the completed form as a PDF using the following naming convention:
Software Producer: Software Producers name which manufactured/compiled the software product
Product name: Complete name of software product
Version: Version number of software product
Attestation date: Date the software product was attested:
e.g. [Software Producer]_[Product]_[Version]_[Attestation Date]
→Acme_SecuritySuite_4.6.2.1_20230124
Individual agencies will provide their respective email addresses.

Filling Out the Form

Software Producer Information

Please provide a description of the software and information about the software producer. All fields in the attestation form are required to be appropriately completed by the software producer. Incomplete forms will not be accepted.

The form must be signed by the Chief Executive Officer (CEO) of the software producer or their designee, who must be an employee of the software producer and have the authority to bind the corporation. By signing, that individual attests that the software in question is developed in conformity with the secure software development practices delineated within this form. The software may be used by a federal agency, consistent with the requirements of M-22-18, as amended by M-23-16, once the agency has received an appropriately signed copy of the attestation form.

The software producer may choose to demonstrate conformance with the minimum requirements by submitting a third-party assessment documenting that conformance. A third-party assessment must be performed by a Third Party Assessor Organization (3PAO) that has either been FedRAMP certified or approved in writing by an appropriate agency official. The 3PAO must use relevant NIST Guidance that includes all elements outlined in this form as part of the assessment baseline. To rely upon a third-party assessment, the software producer must check the appropriate box in Section III and attach the assessment to the form. The producer need not sign the form in this instance. The agency shall take appropriate steps to ensure that the assessment is not posted publicly, either by the vendor or by the agency itself.

Additional Information:

In the event that an agency cannot obtain a completed self-attestation from the software producer, an agency may still decide to use the producer's software if the producer identifies the practices to which they cannot attest, documents practices they have in place to mitigate associated risks, and submits a plan of actions and milestones (POA&M) to the agency. When an attestation is not provided, per OMB guidance, agencies are responsible for requesting from OMB an extension or waiver for the continued use.

This common self-attestation form fulfills the minimum requirements set forth by OMB in M-22-18, as amended by M-23-16.

The attestation form, background, and instructions are subject to change and may be modified.

Secure Software Development Attestation Form

Version 1.0

Section I

New Attestation Attestation Following Extension or Waiver Revised Attestation

Type of Attestation: Company-wide Individual Product Multiple Products or Specific Product Version(s) (please provide complete list)

If this attestation is for an individual product or multiple products, provide the software name, version number, and release/publish date to which this attestation applies. Additional pages can be attached to this attestation if more lines are needed:

Product(s) Name	Version Number ⁴ (if applicable)	Release/Publish Date (if applicable)
		YYYY-MM-DD

For the above specified software, this form does not cover software or any components of that software that fall into the following categories:

1. Software developed by Federal agencies;
2. Open source software that is freely and directly obtained directly by a Federal agency;
3. Third-party open source and proprietary components that are incorporated into the software end product used by the agency; or
4. Software that is freely obtained and publicly available.

Note: In signing this attestation, software producers are attesting to adhering to the secure software development practices outlined in Section III for code developed by the producer.

Section II

1. Software Producer Information

Company Name:

Address:

City:

⁴ Attestations are binding for future versions of the named software product unless and until the software producer notifies the agencies to which it previously submitted the form that its development practices no longer conform to the required elements specified in the attestation.

State or Province:
Postal Code:
Country:
Company Website:

2. Primary Contact for this Document and Related Information (may be an individual, role, or group):

Name:
Title:
Address:
Phone Number:
Email Address (may be an alias/distribution list):

Section III

Attestation and Signature

On behalf of the above-specified company, I attest that, to the best of my knowledge, [software producer] presently makes consistent use of the following practices, derived from the secure software development framework (SSDF),⁵ in developing the software identified in Section I:

- 1) The software is developed and built in secure environments. Those environments are secured by the following actions, at a minimum:
 - a) Separating and protecting each environment involved in developing and building software;
 - b) Regularly logging, monitoring, and auditing trust relationships used for authorization and access:
 - i) to any software development and build environments; and
 - ii) among components within each environment;
 - c) Enforcing multi-factor authentication and conditional access across the environments relevant to developing and building software in a manner that minimizes security risk;
 - d) Taking consistent and reasonable steps to document, as well as minimize use or inclusion of software products that create undue risk within the environments used to develop and

⁵ The SSDF are standards and best practices established by the National Institute of Standards and Technology (NIST) in NIST Special Publication (SP) 800-218.

build software;

- e) Encrypting sensitive data, such as credentials, to the extent practicable and based on risk;
 - f) Implementing defensive cybersecurity practices, including continuous monitoring of operations and alerts and, as necessary, responding to suspected and confirmed cyber incidents;
- 2) The software producer makes a good-faith effort to maintain trusted source code supply chains by employing automated tools or comparable processes to address the security of internal code and third-party components and manage related vulnerabilities;
 - 3) The software producer maintains provenance for internal code and third-party components incorporated into the software to the greatest extent feasible;
 - 4) The software producer employs automated tools or comparable processes that check for security vulnerabilities. In addition:
 - a) The software producer operates these processes on an ongoing basis and prior to product, version, or update releases;
 - b) The software producer has a policy or process to address discovered security vulnerabilities prior to product release; and
 - c) The software producer operates a vulnerability disclosure program and accepts, reviews, and addresses disclosed software vulnerabilities in a timely fashion and according to any timelines specified in the vulnerability disclosure program or applicable policies.
- I further attest that the software producer will notify any agency to which it has submitted this form if and when the producer ceases to make consistent use of the practices identified above in developing the software.

Signature of CEO or Designee with authority to bind the corporation

Date (YYYY-MM-DD): _____

Name:

Title:

OR

- A certified FedRAMP Third Party Assessor Organization (3PAO) or other 3PAO approved in writing by an appropriate agency official has evaluated our conformance to all elements in this form. The 3PAO used relevant NIST Guidance that includes all elements outlined in this form as the assessment baseline. The assessment is attached.

ATTACHMENT(S):

- **[Artifact/Addendum Title]:** [Artifact/Addendum Description]

Burden Statement

The public reporting burden to complete this information collection is estimated at **3 hours and 20 minutes** per response, including time for reviewing instructions, searching data sources, gathering, and maintaining the data needed, and completing and reviewing the collection of information. An agency may not conduct or sponsor, and a person is not required to respond to, a collection of information unless it displays a currently valid OMB control number and expiration date. Send comments regarding this burden estimate or any other aspect of this collection information, including suggestions for reducing this burden, to DHS/CISA [**CSCRM@cisa.dhs.gov**](mailto:CSCRM@cisa.dhs.gov).

APPENDIX REFERENCES

Minimum Attestation References:

The minimum requirements within the Secure Software Attestation Form address requirements put forth in E.O. 14028 subsection (4)(e). A mapping to specific SSDF practices and tasks is provided for reference purposes.

Attestation Requirements	Related E.O. 14028 Subsection		Related SSDF Practices and Tasks
1) The software is developed and built in secure environments. Those environments are secured by the following actions, at a minimum:	4e(i)		[See rows below]
a) Separating and protecting each environment involved in developing and building software;	4e(i)(A)		PO.5.1
b) Regularly logging, monitoring, and auditing trust relationships used for authorization and access: i) to any software development and build environments; and ii) among components within each environment;	4e(i)(B)		PO.5.1
c) Enforcing multi-factor authentication and conditional access across the environments relevant to developing and building software in a manner that minimizes security risk;	4e(i)(C)		PO.5.1, PO.5.2
d) Taking consistent and reasonable steps to document, as well as minimize use or inclusion of software products that create undue risk within the environments used to develop and build software;	4e(i)(D)		PO.5.1
e) Encrypting sensitive data, such as credentials, to the extent practicable and based on risk;	4e(i)(E)		PO.5.2
f) Implementing defensive cybersecurity practices, including continuous monitoring of operations and alerts and, as necessary,	4e(i)(F)		PO.3.2, PO.3.3, PO.5.1, PO.5.2

responding to suspected and confirmed cyber incidents;			
2) The software producer makes a good-faith effort to maintain trusted source code supply chains by employing automated tools or comparable processes to address the security of internal code and third-party components and manage related vulnerabilities;	4e(iii)		PO 1.1, PO.3.1, PO.3.2, PO.5.1, PO.5.2, PS.1.1, PS.2.1, PS.3.1, PW.4.1, PW.4.4, PW 7.1, PW 8.1, RV 1.1
3) The software producer maintains provenance for internal code and third-party components incorporated into the software to the greatest extent feasible;	4e(vi)		PO.1.3, PO.3.2, PO.5.1, PO.5.2, PS.3.1, PS.3.2, PW.4.1, PW.4.4, RV.1.1, RV.1.2
4) The software producer employed automated tools or comparable processes that check for security vulnerabilities. In addition: a) The software producer operates these processes on an ongoing basis and prior to product, version, or update releases; b) The software producer has a policy or process to address discovered security vulnerabilities prior to product release; and c) The software producer operates a vulnerability disclosure program and accepts, reviews, and addresses disclosed software vulnerabilities in a timely fashion and according to any timelines specified in the vulnerability disclosure program or applicable policies.	4e(iv)		PO.4.1, PO.4.2, PS.1.1, PW.2.1, PW.4.4, PW.5.1, PW.6.1, PW.6.2, PW.7.1, PW.7.2, PW.8.2, PW.9.1, PW.9.2, RV.1.1, RV.1.2, RV.1.3, RV.2.1, RV.2.2, RV.3.3

Enclosure 3

NASA Software Attestation Frequently Asked Questions (FAQs)

- **Who is ultimately accountable, at NASA, including NASA SEWP, for ensuring these requirements are met?**
 - ✓ Shared accountability with the Chief Information Officer (CIO) and Chief Acquisition Officer (CAO)
- **How will NASA collect attestations?**
 - ✓ NASA will leverage CISAs [Repository for Software Attestations and Artifacts \(RSAA\)](#). Software publishers should upload attestations and artifacts, as needed, into RSAA.
- **What is the definition of “software?” [Ref. M-22-18 & M-23-16]**
 - ✓ Firmware, operating systems, applications, and application services (e.g., cloud-based software), as well as products containing software.
 - ✓ Please note this does include Operational Technology (OT) and Internet of Things (IoT) devices and software that enables them.
- **What software is in-scope of the M-22-18/M-23-16 requirements?**
 - ✓ Software developed, or released, after September 14, 2022.
 - ✓ Software developed prior to September 14, 2022, but modified by a major version change after September 14, 2022.
 - ✓ Software code where the producer delivers continuous changes (such as software-as-a-service products or other products using continuous delivery/continuous deployment).
- **What software is out-of-scope of the M-22-18/M-23-16 requirements?**
 - ✓ Software developed by Federal agencies.
 - ✓ Open-source software that is freely and directly obtained by a Federal Agency
 - ✓ 3rd party components
 - ✓ Freely obtained and publicly available proprietary software
- **What is open-source software? [Ref. M-23-16]**
 - ✓ Freely obtained software that can be accessed, used, modified, and shared by anyone.

- ✓ Though freely obtained, demonstrations or pilots of software products that are otherwise unavailable on a no-cost basis remain subject to M-22-18 attestation requirements, as amended.
- **What is a 3rd party component? [Ref. M-22-18 & NIST SP 800-95]**
 - ✓ A software object, meant to interact with other components, encapsulating certain functionality or a set of functionalities. A component has a clearly defined interface and conforms to a prescribed behavior common to all components within an architecture.
- **What is an example of freely obtained and publicly available proprietary software? [Ref. M-23-16]**
 - ✓ Web Browsers
- **What is the definition of critical software?**
 - ✓ Please refer to <https://www.nist.gov/itl/executive-order-improving-nations-cybersecurity/critical-software-definition-explanatory>
- **As a software publisher, what am I attesting to?**
 - ✓ Software publishers are attesting to utilizing secure software development practices in the development and distribution of software listed on the “Software Records” tab in RSAA.
 - ✓ Software publishers are attesting to the security of the software “as built.” If you include open-source software within your commercial product, you are attesting that you have a secure build process, know where you source it from, and that you've tested it, etc. (Ref. common form requirement #3)
- **How often must the software publisher provide an attestation?**
 - ✓ Software producers should provide an attestation each time implementation of the secure software development practice(s) changes.
 - ✓ Attestations that included a Plan of Action & Milestones (POA&M) should be updated when the POA&M closure evidence has been uploaded as an “Artifact” in RSAA.
- **What can partners expect from NASAs Solutions for Enterprise-wide Procurement (SEWP)?**
 - ✓ Please visit https://www.sewp.nasa.gov/software_attestation.shtml
- **Who are NASAs POCs and how do I reach them?**
 - ✓ Kanitra Tyler, Information and Communications Technology (ICT)/Cyber Supply Chain Risk Management (C-SCRM) Service Element Lead, Brittney Chappell,

Senior Procurement Analyst, Kay Twitchell, Deputy Software License and Asset Manager and Theresa Kinney, NASA SEWP Deputy Program Manager can be reach via Agency-DL-SoftwareAttestation@mail.nasa.gov