Definition of Critical Software Under Executive Order (EO) 14028

October 13. 2021

Note:

NIST is updating its characterization of critical software to reflect conversations with the National Security Council (NSC) and the Office of Management and Budget (OMB). The definition of critical software applies only to Government *management* of software (Sections 4i and 4j). The requirements in 4e and 4k related to *acquisition* apply to all software, not just to critical software. This does not alter the definition of critical software, although it changes NIST's initial guidance about how the definition should be used. NIST has modified several FAQs accordingly.

Introduction

Executive Order (EO) 14028 on Improving the Nation's Cybersecurity, issued on May 12, 2021, directs the National Institute of Standards and Technology (NIST) to publish a definition of the term *critical software*.

> *(g) Within 45 days of the date of this order, the Secretary of Commerce, acting through the Director of NIST, in consultation with the Secretary of Defense acting through the Director of the NSA, the Secretary of Homeland Security acting through the Director of CISA, the Director of OMB, and the Director of National Intelligence, shall publish a definition of the term "critical software" for inclusion in the guidance issued pursuant to subsection (e) of this section. That definition shall reflect the level of privilege or access required to function, integration and dependencies with other software, direct access to networking and computing resources, performance of a function critical to trust, and potential for harm if compromised.*

The EO directs the Cybersecurity & Infrastructure Security Agency (CISA) to use this published definition of *critical software* to develop a list of software categories and products that are in scope for that definition and thus subject to the further requirements of the EO.

> *(h) Within 30 days of the publication of the definition required by subsection (g) of this section, the Secretary of Homeland Security acting through the Director of CISA, in consultation with the Secretary of Commerce acting through the Director of NIST, shall identify and make available to agencies a list of categories of software and software products in use or in the acquisition process meeting the definition of critical software issued pursuant to subsection (g) of this section.*

To coordinate the definition with its eventual application, NIST solicited position papers from the community, hosted a virtual workshop to gather input, and consulted with CISA, the Office of Management and Budget (OMB), the Office of the Director of National Intelligence (ODNI), and the National Security Agency (NSA) to develop the definition, the concept of a phased implementation, and a preliminary list of common categories of software that would fall within the scope for the initial phase. Additional guidance on applying this definition for implementing the EO will be forthcoming from CISA and OMB. NIST worked closely with CISA and OMB to ensure that the definition and recommendations are consistent with their plans.

This paper starts with background information and context for the term *critical* and introduces the concept of a phased approach. It defines the term *critical software* in the context of the EO and provides a preliminary list of software that meets the definition of EO-critical and is recommended to be included in the initial phase of implementation. The paper concludes with frequently asked questions (FAQs). CISA will provide the final set of software categories for the initial and future implementation phases.

Background

Recent incidents have demonstrated the need for the Federal Government to improve its efforts to identify, deter, protect against, detect, and respond to malicious cyber actions and actors. In particular, threat actors are exploiting the pervasive use of software and the complexity of the underlying code and software development and distribution practices. One of the goals of the EO is to assist in developing a security baseline for critical software products used across the Federal Government. The designation of software as *EO-critical* will then drive additional activities, including how the Federal Government purchases and manages deployed critical software. In particular, the EO seeks to limit acquisition to software that has met security measures such as use of a secure development process and integrity checks that are defined in Section 4(e) of the EO.

Given the broad scope of the EO and its potential impact on both government operations and the software marketplace, NIST set the following goals for the definition of critical software:

- Clarity – The implementation of the program will drive activity across the entire Federal Government, with impacts on the software industry. Having a clear definition that can be used by the software industry and the Government is vital to the successful implementation of the EO.
- Viability – For the EO to be viable, its implementation must take into consideration how the software industry functions, including product development, procurement, and deployment. The software marketplace is dynamic and evolves continuously. How software is developed, brought into an organization, and used by an organization is changing rapidly. Software is purchased as a product, as part of a product, and as a service. Software is often modular, consisting of many components.

There are many existing definitions and uses of the term *critical*. Most are based on how technology supports various tasks or processes, such as *safety critical* or *critical infrastructure*. The use of the term in the EO is slightly different because it is based not on the context of use, but on the properties of a given piece of software that make it likely to be critical in most use cases. That is, it focuses on critical functions that address underlying infrastructure for cyber operations and security. This is similar to the concept of *Federal Civilian Enterprise Essential IT under the High Value Assets program.*

In order to separate the common usage of *critical* with the definition under the EO, we will use the term *EO-critical* when it is unclear which usage is being discussed.

Approach

Given the size, scope, and complexity of the software marketplace and the infrastructure needed within the government to implement the EO, NIST has consulted with key agencies regarding the concept of a phased approach for securing the supply chain of EO-critical software. This will allow both the Federal Government and the software industry to implement the EO in an incremental manner, thus providing the opportunity for feedback and improvements to its processes with each additional phase.

Definition and Explanatory Material

This section provides the definition of EO-critical software. Following that is a table with a preliminary list of software categories recommended for the initial phase along with some explanatory material. At a later date, CISA will provide the authoritative list of software categories that are within the scope of the definition and to be included in the initial phase of implementation. A pointer to that information will be provided here when available.

Finally, there is a set of FAQs at the bottom of the paper that provides answers to questions that may arise about the interpretation of the definition, the phased approach, and other related topics.

***EO-critical software* is defined as any software that has, or has direct software dependencies upon, one or more components with at least one of these attributes:**

- **is designed to run with elevated privilege or manage privileges;**
- **has direct or privileged access to networking or computing resources;**
- **is designed to control access to data or operational technology;**
- **performs a function critical to trust; or,**
- **operates outside of normal trust boundaries with privileged access.**

The definition applies to software of all forms (e.g., standalone software, software integral to specific devices or hardware components, cloud-based software) purchased for, or deployed in, production systems and used for operational purposes. Other use cases, such as software solely used for research or testing that is not deployed in production systems, are outside of the scope of this definition. (See FAQ #10 and FAQ #11.)

NIST recommends that the initial EO implementation phase focus on standalone, on-premises software that has security-critical functions or poses similar significant potential for harm if compromised. Subsequent phases may address other categories of software such as:

- software that controls access to data;

- cloud-based and hybrid software;
- software development tools such as code repository systems, development tools, testing software, integration software, packaging software, and deployment software;
- software components in boot-level firmware; or
- software components in operational technology (OT).

The table below provides a preliminary list of software categories considered to be EO-critical. This table is provided to illustrate the application of the definition of EO-critical software to the scope of the recommended initial implementation phase described above. As noted previously, CISA will provide the authoritative list of software categories at a later date.

Category of Software	Description	Types of Products	Rationale for Inclusion
Identity, credential, and access management (ICAM)	Software that centrally identifies, authenticates, manages access rights for, or enforces access decisions for organizational users, systems, and devices	• Identity management systems • Identity provider and federation services • Certificate issuers • Access brokers • Privileged access management software • Public key infrastructure	• Foundational for ensuring that only authorized users, systems, and devices can obtain access to sensitive information and functions
Operating systems, hypervisors, container environments	Software that establishes or manages access and control of hardware resources (bare metal or virtualized/ containerized) and provides common services such as access control, memory management, and runtime execution environments to software applications and/or interactive users	• Operating systems for servers, desktops, and mobile devices • Hypervisors and container runtime systems that support virtualized execution of operating systems and similar environments	• Highly privileged software with direct access and control of underlying hardware resources and that provides the most basic and critical trust and security functions

Category of Software	Description	Types of Products	Rationale for Inclusion
Web browsers	Software that processes content delivered by web servers over a network, and is often used as the user interface to device and service configuration functions	• Standalone and embedded browsers	• Performs multiple access management functions • Supports browser plug-ins and extensions such as password managers for storing credentials for web server resources • Provides execution environments for code downloaded from remote sources • Provides access management for stored content, such as an access token which is provided to web servers upon request
Endpoint security	Software installed on an endpoint, usually with elevated privileges which enable or contribute to the secure operation of the endpoint or enable the detailed collection of information about the endpoint	• Full disk encryption • Password managers • Software that searches for, removes, or quarantines malicious software • Software that reports the security state of the endpoint (vulnerabilities and configurations) • Software that collects detailed information about the state of the firmware, operating system, applications, user and service accounts, and runtime environment	• Has privileged access to data, security information, and services to enable deep inspection of both user and system data • Provides functions critical to trust

Category of Software	Description	Types of Products	Rationale for Inclusion
Network control	Software that implements protocols, algorithms, and functions to configure, control, monitor, and secure the flow of data across a network	• Routing protocols • DNS resolvers and servers • Software-defined network control protocols • Virtual private network (VPN) software • Host configuration protocols	• Privileged access to critical network control functions • Often subverted by malware as the first step in more sophisticated attacks to exfiltrate data
Network protection	Products that prevent malicious network traffic from entering or leaving a network segment or system boundary	• Firewalls, intrusion detection/ avoidance systems • Network-based policy enforcement points • Application firewalls and inspection systems	• Provides a function critical to trust, often with elevated privileges
Network monitoring and configuration	Network-based monitoring and management software with the ability to change the state of—or with installed agents or special privileges on—a wide range of systems	• Network management systems • Network configuration management tools • Network traffic monitoring systems	• Capable of monitoring and/or configuring enterprise IT systems using elevated privileges and/or remote installed agents

Category of Software	Description	Types of Products	Rationale for Inclusion
Operational monitoring and analysis	Software deployed to report operational status and security information about remote systems and the software used to process, analyze, and respond to that information	• Security information and event management (SIEM) systems	• Software agents widely deployed with elevated privilege on remote systems • Analysis systems critical to incident detection and response and to forensic root cause analysis of security events • Often targeted by malware trying to deactivate or evade it
Remote scanning	Software that determines the state of endpoints on a network by performing network scanning of exposed services	• Vulnerability detection and management software	• Typically has privileged access to network services and collects sensitive information about the vulnerabilities of other systems
Remote access and configuration management	Software for remote system administration and configuration of endpoints or remote control of other systems	• Policy management • Update/patch management • Application configuration management systems • Remote access/ sharing software • Asset discovery and inventory systems • Mobile device management systems	• Operates with significant access and elevated privileges, usually with little visibility or control for the endpoint user

Category of Software	Description	Types of Products	Rationale for Inclusion
Backup/recovery and remote storage	Software deployed to create copies and transfer data stored on endpoints or other networked devices	• Backup service systems • Recovery managers • Network-attached storage (NAS) and storage area network (SAN) software	• Privileged access to user and system data • Essential for performing response and recovery functions after a cyber incident (e.g., ransomware)

FAQs

The following FAQs were compiled in consultation with OMB and CISA to provide additional context to the material.

1. **When will the next phase begin?**

   CISA and OMB will monitor the implementation of the program in the initial phase and decide when to include additional software categories.

2. **What do you mean by "direct software dependencies" in the definition?**

   For a given component or product, we mean other software components (e.g., libraries, packages, modules) that are directly integrated into, and necessary for operation of, the software instance in question. This is not a systems definition of dependencies and does not include the interfaces and services of what are otherwise independent products.

3. **What do you mean by "critical to trust" in the definition?**

   "Critical to trust" covers categories of software used for security functions such as network control, endpoint security, and network protection.

4. **Does it matter if the software product is in the cloud or in an on-premises or a hybrid environment?**

   No. If a product or service provides functions that are part of the definition of EO-critical, then the product or service itself is EO-critical, regardless of its deployment model. Having said that, NIST has recommended that the initial phase of the EO focus on on-premises software. Many on-premises products rely on cloud-based components and services that perform EO-critical functions (e.g., cloud-based access control). In such situations, the on-premises components are in scope if they directly perform EO-critical functions. It is suggested that cloud-based components and systems be addressed in later phases of implementation to allow time to coordinate with other Federal requirements for such systems (e.g., FedRAMP).

5. **Can open source software be EO-critical?**

   Yes. If open source software performs functions that are defined as EO-critical, then it is EO-critical. In practice, open source software is often incorporated into other products..

6. **Can Government-developed software be EO-critical?**

   Yes. The Federal Government develops software both in-house and through contracts. These products are often referred to as GOTS (government-off-the-shelf) software. If GOTS software performs functions included in the definition of EO-critical, then it is EO-critical.

7. **What if a product is partly EO-critical and partly not?**

   If a product contains functions that are part of the definition of EO-critical, then the product itself is EO-critical. However, some EO-critical software products may contain distinct components that do not have EO-critical attributes or do not directly support the EO-critical functions provided by the product.

8. **How will this work with FedRAMP?**

   Section 3 of the EO addresses modernization of FedRAMP. The recommended phased approach starts with on-premises software, with the understanding that some on-premises software which relies on cloud-hosted components may be in scope. CISA will coordinate with FedRAMP to define the scope and applicability of the EO to cloud-based software in later phases of the implementation.

9. **The definition excludes software that won't be deployed in production systems for operational purposes. Can you provide more explanation?**

   There are several use cases where software is owned but is not deployed in a manner that would pose a significant risk of harm if compromised. Examples include software used as the subject of research and software collected for archival purposes.

10. **What about software used in National Security Systems (NSS)? Are they covered?**

    Section 9 of the EO describes the applicability of the requirements of this EO to National Security Systems.

11. **Can embedded software or firmware be EO-critical?**

    Yes. If embedded software or firmware performs functions that are defined as EO-critical, then it is EO-critical. Due to the complexities of such products, we recommended that such software not be included in the initial phase of implementation.

12. **Shouldn't departments and agencies decide what is EO-critical based on how the software is used to support the agency's mission?**

    No. The definition of EO-critical is based on the functions of the software, not its use. The types of software defined by the table are likely to be EO-critical in most situations.

13. **What about safety-critical or other high-assurance systems?**

There are many types of safety-critical and other high-assurance systems. Many of them have regulatory or industry-based security requirements. If these systems make use of software that contains EO-critical functions, then that software is EO-critical. Safety-critical and high-assurance software and systems will have additional security requirements. For example, if a high-assurance system contains an operating system, the operating system is EO-critical and must meet the EO-critical requirements in addition to the safety-critical or other system requirements.

.