MEMORANDUM FOR SEE DISTRIBUTION

SUBJECT: Cybersecurity Policy for Container Use within the Army

1. References. See enclosure.

2. Purpose. The purpose of this memorandum is to amplify guidance on the roles and responsibilities of key stakeholders in ensuring the secure development, deployment, and management of container applications in accordance with established policies and guidelines within the Department of the Army and subordinate commands.

3. Applicability. This policy applies to container use within the Department of the Army. Per AR 25-2, para 2-7, the Army Chief Information Officer (CIO), on behalf of the Secretary of the Army, establishes policy, resourcing, and oversight of the Army Cybersecurity Program. This guidance memorandum meets provisions outlined in AR 25-2, para 1-8, where the Army CIO, when needed, will issue policy memoranda to amplify guidance for the policies in AR 25-2.

4. Roles and Responsibilities:

   a. The Army CIO: The Army CIO will review and approve updates to this policy, ensuring concurrence and overseeing adherence for containerized applications.

   b. The Authorizing Official (AO):

      (1) Ensures proper oversight of cybersecurity related functions, to include containerized applications.

      (2) Authorizes and accepts the risk for the container application authorization framework and the container application deployment platform.

      (3) Remains accountable for the continued operation of the container application authorization framework and the container application deployment platform.
SUBJECT: Cybersecurity Policy for Container Use within the Army

c. Program Information System Security Managers (P-ISSM) must:

(1) Functions as the primary cybersecurity technical advisor to the AO and managerial lead throughout the container application development lifecycle.

(2) Ensure the secure configuration and approval of the container application authorization framework in accordance with this policy and referenced guidance.

(3) Approve the overall container application authorization framework and provide the AO an overall assessment.

d. Information System Owner/Project Manager must:

(1) Ensure an Organizational Information System Security Manager (O-ISSM) is appointed.

(2) Ensure required assessments and secure configuration are completed.

(3) Collaborate with the O-ISSM to ensure adherence to every component of the container application authorization framework.

e. O-ISSMs must:

(1) Manage security issues related to container applications within their scope.

(2) Verify and approve each submitted Body of Evidence (BoE).

(3) Ensure the availability of container application manifest BoEs in eMASS.

f. Security Control Assessors (SCAs) must:

(1) Conduct security validation throughout the container application development lifecycle.

(2) Validate secure development by ensuring the artifacts of containerized application meet standards and mitigations to determine and make recommendations on residual risk.

(3) Ensure the O-ISSM is provided or given access to the body of evidence (BoE) and artifacts.
5. Policy.

a. All Components. This policy applies to all the components of container-based solutions, which includes the base containers, application containers, container application development platforms, container application authorization frameworks, container application manifests, and container application deployment platforms.

   (1) Must be secure by design. The responsibility for security resides with all parties including the developers. Security must be addressed from the design phase and ensured in subsequent phases of the container application development lifecycle.

   (2) Must adhere to the principle of open design. Every component covered by this policy must be fully transparent, auditable, and traceable. Artifacts will be available for review by internal parties.

   (3) Must adhere to the principle of modular design. Container applications are modular by design, but every element must also extend the principle to the security primitives to allow for the most current and secure versions to be utilized.

b. Base Containers. The containers from which application containers are derived.

   (1) Must utilize minimal supply chain by pulling binaries directly from the maintainers.

   (2) Must verify the integrity of the binaries on every external access.

   (3) Must be as current as possible to limit the impact of emerging vulnerabilities.

   (4) Must be as minimal as possible to reduce risk exposure.

   (5) Must be hosted on authorized container development platforms.

   (6) Must include inheritable and appropriate STIGs for container use-cases.

c. Application Containers. The container or containers that comprise the container application.

   (1) Must adhere to the principle of least privilege and separation of privilege.

      (a) The runtime user must be non-root or provide a detailed description of privileged execution is necessary.
(b) Exposed ports should be in the non-privileged range (1025-65536), where possible, to allow the deployment platform to also adhere to least privilege.

(c) Not include embedded credentials or other sensitive information.

(2) Must adhere to the principle of minimized and least common mechanism.

(a) Utilize an additive approach to dependencies and code, which shall only include necessary dependencies and segments.

(b) Utilize multi-stage builds to minimize containers when build dependencies are greater in number or size than runtime dependencies.

(c) Utilize container optimization tools to further minimize containers and/or use distroless as the final layer.

(d) Use minimized containers in production unless no other options are available for the use case.

(e) Run a single command unless a security reason requires a second security-focused process to be running.

(f) Limit containers to a single listening service at most except where a health or security service needs to be included in the container rather than pod.

(g) Implement applicable STIGs based on container utilization.

d. Container Application Development Platforms. These platforms contain the tools necessary for secure and efficient container application development. They may also be general application development platforms, but those details are not covered in this policy.

(1) Must provide a mechanism for secure and collaborative source code management, such as GitLab.

(2) Must provide executors to enable secure and unprivileged container builds, such as GitLab runners.

(3) Except for the executors, these platforms at minimum must implement impact level 4 controls and be isolated from container application deployment platforms.

(4) Must provide version-controlled registries for containers and security artifacts.
(5) Must provide a mechanism for secure and collaborative project management, documentation, and static content host.

(6) Should provide self-service for as many components of the container application development platform as possible.

(7) Must provide a mechanism that enables the assessment of supply chain security risks, which, at a minimum, enforces the scanning of all dependencies for malicious signatures.

(8) Must include a mechanism that enables the tracking and verification of cryptographic signature chains for containers and authorization statuses.

(9) Should provide a messaging and alerting service for real-time collaboration and bot integration.

e. Container Application Authorization Frameworks. These frameworks reside on the container application development platforms. They streamline the development process and automate as much of the security process as possible by limiting the non-developers involved to essential security checkpoints. They leverage continuous integration and continuous deployment automation and include additional tools to make these optimizations possible.

(1) Containers and their source code must be subject to the following assessments:

(a) Appropriate use of approved, secure base containers from authorized container application development platforms.

(b) Credential detection on the project.

(c) Unit testing with appropriate code coverage levels for the container application deployment platform environment (development, testing, production).

(d) Static application security testing (SAST) or source code analysis for custom source.

(e) Container linting and safety checks.

(f) Secure, non-privileged container build process.
(g) Scan for known malware signatures.

(h) Software composition analysis including vulnerability scan of all components and dependencies for the container application deployment platform environment. A software bill of materials (SBOM) must be generated and stored as an artifact.

(i) Dynamic Application Security Testing (DAST) for containers with listening services.

(2) The artifacts from the container assessments comprise the BoE, which must be available, auditable, and traceable to the specific version of the container for which the artifacts were generated.

(3) Authorized containers must be reassessed monthly. These frameworks should automate this process.

(4) Must provide a mechanism for container application authorization status tracking.

(5) Findings must be mitigated in the context of each container’s intended utilization.

(6) Cybersecurity personnel must either review the artifacts for each container or set criteria for each artifact and an associated pipeline implementation.

(7) Must provide a mechanism that coalesces the set of containers that comprise the container application manifest for holistic security review. These frameworks should automate this process.

f. Container Application Manifests. The container application manifests describe the deployment of the container application.

(1) Must include documentation that details the interactions between the container that comprise the container application and the container application deployment platform.

(2) Must align with container application deployment platform(s) and inherit available security controls.

(3) Must enforce least privilege by only adding capabilities to containers that are necessary for the application to function as described in this manifest. Persistent
containers with system-level privileges should be avoided.

(4) Must not include hardcoded credentials.

(5) Should specify resource limits for every container, including ephemeral, to limit the impacts of unforeseen issues.

(6) Should follow secure volume practices by using read-only file systems where possible and must not mount sensitive host directories.

(7) Should include mechanisms to enable readiness and liveness probes for container health.

g. Container Application Deployment Platforms. These platforms are where container applications run and the mechanism that provides access, authentication, and a security boundary for the container application. They provide the common, core services that enable the developers of container applications to operate in a truly agile fashion that benefits from the advantages of DevSecOps.

(1) Must implement security controls and common services that are inheritable by container applications.

(2) Must implement continuous monitoring at all levels from the host- or service-level through to the container application level.

(3) Should employ intrusion prevention systems that employ active countermeasures but must at least employ intrusion detection systems that provide alerts on anomalous events.

(4) Should implement separate environments for development, testing, and production that are isolated from each other and the container application development platforms but must at least isolate the development environment from the production-level container application deployment platform.

(5) Should have tiered security scrutiny to allow the greatest flexibility to the development environment and the strictest security for production.

(6) Should have mechanisms for promotion from one environment to the next given the security requirements have been met.

(7) Must minimize, harden, and secure hosts and services in accordance with applicable guidance.
6. Exception Requests. Requests for exceptions to the requirements in this policy should be documented and submitted to the Army CIO. Exception requests must include justification (including cost and mission impacts). Each request will be adjudicated based upon all factors including cybersecurity risk.

7. Policy Review. The OCIO Cybersecurity Directorate (SAIS-CSD) will ensure review of this policy annually for update, rescission, or inclusion in Army regulation as appropriate.

8. Points of contact.
   a. For policy questions: usarmy.pentagon.hqda-cio.mbx.policy-inbox@army.mil.
   b. SAIS-CS Director: MG Jan C. Norris, Army CISO, jan.c.norris.mil@army.mil.
   c. SAIS-CSP: Christopher Joseph, christopher.a.joseph4.civ@army.mil.
   d. ARCYBER: MAJ Daniel S. Hawthorne, daniel.s.hawthorne.mil@army.mil.

DISTRIBUTION:
Principal Officials of Headquarters, Department of the Army
Commander
   U.S. Army Forces Command
   U.S. Army Training and Doctrine Command
   U.S. Army Materiel Command
   U.S. Army Futures Command
   U.S. Army Pacific
   U.S. Army Europe and Africa
   U.S. Army Central
   U.S. Army North
   U.S. Army South
   U.S. Army Special Operations Command
(CONT)
SUBJECT: Cybersecurity Policy for Container Use within the Army

DISTRIBUTION: (CONT)
- Military Surface Deployment and Distribution Command
- U.S. Army Space and Missile Defense Command/Army Strategic Command
- U.S. Army Cyber Command
- U.S. Army Medical Command
- U.S. Army Intelligence and Security Command
- U.S. Army Corps of Engineers
- U.S. Army Military District of Washington
- U.S. Army Test and Evaluation Command
- U.S. Army Human Resources Command
- Superintendent, U.S. Military Academy
- Commandant, U.S. Army War College
- Director, U.S. Army Civilian Human Resources Agency
- Executive Director, Military Postal Service Agency
- Director, U.S. Army Criminal Investigation Division
- Superintendent, Arlington National Cemetery
- Director, U.S. Army Acquisition Support Center

CF:
- Principal Cyber Advisor
- Director of Business Transformation
- Commander, Eighth Army
References


b. Department of Defense Instruction (DoDI) 8510.01 (Risk Management Framework for DoD Information Technology).


g. NIST Special Publication (SP) 800-53 (Security and Privacy Controls for Federal Information Systems and Organizations, Revision 5), September 2020.

h. NIST SP 800-190 (Application Container Security Guide), September 2017.

i. White House Executive Order (EO) 13800 (Presidential Executive Order on Strengthening the Cybersecurity of Federal Networks and Critical Infrastructure).