PLATFORM ONE (P1)

IRON BANK VALUE STREAM

Iron Bank Acceptance Baseline Criteria

Document Version: 1.0

April 2022

Prepared by John Bermudez, Seed Innovations

[DISCLAIMER NOTICE]

Change Record

Date	Version	Page/Paragraph	Description of Change	Made by:
21 April 2022	DRAFT	Entire Document	Document Draft Update	Christoper Vernooy, Phillip Record, Blake Burkhart
05 May 2022	1.0	Entire Document	Initial Release - MVP	Danny Holtzman
07 Sept 2022	1.1		Minor updates to match what is currently implemented	

Table of Contents

1. Introduction

1.1 Background

Platform One's Iron Bank serves as a registry of hardened containers for the Department of Defense (DoD) - vendor software containers and open-source software (OSS) containers may be contributed to their component forms to Iron Bank, then built to be made available to customers and downstream programs. Section 4 of Executive Order (EO) 14028 (Improving the Nation’s Cybersecurity) tasks the National Institute of Standards and Technology (NIST) to initiate pilot programs for cybersecurity labeling in coordination with the Federal Trade Commission (FTC) and other agencies. This directs NIST to “. . . identify secure software development practices or criteria for a consumer software labeling program . . . [whose criteria] must reflect a baseline level of security practices . . . "

This document is the finalization of the acceptance baseline criteria (ABCs) to guide contributors on the requirements for acceptance into the Iron Bank. A major change is that only Big Bang (core + addon) containers will be labelled as "approved". Approval often entails the container under assessment meeting or exceeding Iron Bank Hardening Standards, though in some cases a container may be approved if it does not meet the hardening standards in this document. Other containers may be labeled ‘***verified***’ or ‘***unverified***’ based on the level of manual review by Iron Bank for the findings and justifications.

All containers will have a scorecard broken down based on where they are passing and failing on the acceptance standards. Containers did not get an ATO or CTF in the previous process, *and they will not in this revision of the process either*. The major change is that containers will not have to be inspected and every finding approved prior to the container and associated standards compliance being made available in Iron Bank, if the container is not part of Big Bang (Core + Addons).

2. Document Scope and Goals

Iron Bank's current process for hardening, remediation, justification, and certification is time consuming and often produces disparate outcomes, creating confusion internally and for our partners, customers, and contributors. When contributors are unable to complete the process, their application cannot be added to Iron Bank and is unavailable as a hardened image for Iron Bank consumers.

To provide a clear, consistent experience for all, while bringing the process into greater alignment with DISA's *DevSecOps Enterprise Container Hardening Guide v1.1*, more prescriptive standards for Iron Bank's minimum baseline criteria are needed.

Acceptance Baseline Criteria (ABC) will provide clarity and context to all parties operating with or within Iron Bank. In addition, a process for calculating a risk score for each container regardless of compliance with Iron Bank standards will present a synthesis of risk factors to downstream consumers, called the *Overall Risk Assessment (ORA)*. This is designed to be a living policy document and will be updated as needed. History of changes will be added as they occur and any major revisions that impact our partners, vendors, or customers will result in re-socialization of the document.

3. Terminology & Acronyms

Term or Acronym	Meaning
Approved/Hardened Base Image	Base images that exist in Iron Bank that meet or exceed the hardening standards in this guide or have been approved based on the mitigation or justification of any issues and accepted by Iron Bank Container Approvers
hardening_manifest.yml	Structured data file using yaml that requires specific information such as repository, tags, labels, all resources that need to be added to the container from any upstream source, maintainers, etc.
Development Tools	development packages (*-devel.rpm/deb, *-dev.rpm/deb, etc), compilers, or other tools used to build code or support that
Justifications	explanation of why the vulnerability exists and if/when it can be resolved OR why it is a false positive or unable to exploited
Secrets	passwords, private keys, or other files that contain sensitive authentication information or placeholders for those files
Mitigation	degrading or removing the potential for a vulnerability to be exploited or reducing the severity of impact if it is exploited.
Remediation	removing a vulnerability or fixing a compliance issue (e.g., file permissions issue)

UBI	Universal Base Image created by RedHat based off of RHEL (RedHat Enterprise Linux) operating system
Distroless	blank base image that has a minimal operating system and set of tools to leverage the host kernel. all other software needs to be installed. Creates a much smaller image but doesn't include some basic tools.
STDOUT	Standard output on Linux, this is the default file descriptor that a process can write to
STIG	Security Technical Implementation Guides for various systems published by the DoD. They include hardening recommendations for operating systems (OS's) and applications
TLS	Transport Layer Security. This is frequently used to protect data in transit using an encrypted tunnel
CVE	Common Vulnerabilities and Exposures, a list of publicly known and tracked security flaws in various technologies including hardware, software, appliances, routers, etc.
OSSF Scorecard	Open-Source Security Foundation Scorecard. "Scorecards is an automated tool that assesses a number of important heuristics ("checks") associated with software security and assigns each check a score of 0-10.”
Justification	A justification of any kind must be provided in this timeframe.
Remediation	A finding must be fully resolved within this time frame as defined in *Section 4.3, Table B*
Mitigation	Mitigation of a vulnerable component entails reducing the risk of exploit, reducing the footprint of the vulnerability and/or reducing the severity (impact) of a successful exploit of the vulnerable component. In addition, it can include steps to monitor for Indicators of Compromise (IoC).
Max Count	Maximum quantity of vulnerabilities that will be accepted until the specified CVE Age Tolerance, as defined in Section 2.4, Table B
CVE Age (Published Date) Tolerance	Findings with an original security advisory publish date older than this period are not allowed without full resolution of the issue

4. Acceptance baseline criteria for Approved Containers (ABC's)

4.1 Methodology

To ensure the best possible outcome, Iron Bank will adhere to the following process:

1. Ensure the current process is documented, and all requirements are enumerated.
2. Align with DISA standards in *Final_DevSecOps_Enterprise_Container_Hardening_Guide_1.1.pdf*, sections 2.2-2.4 specifically.
3. Centralize and define all minimum standard criteria for containers for internal and external consumption.
4. Ensure containers meet a minimum-security standard as outlined above and define the process for any deviation.

⚠ **Iron Bank may refuse to harden or accept onboarding of any container for any reason.**

4.2 Minimum Baseline Criteria for Compliance with Iron Bank Hardening Standards

4.2.1 Fundamental Requirements

1. Containers must include any dependencies or file to be downloaded from the Internet in the resources section of the hardening_manifest.yaml file, and a checksum of the file for verification in each repository.
    1. All container `***entrypoint***` scripts must be included in the Repo1 container repository in a folder named `***scripts***` and copied into the container.
    2. Container build instructions/scripts will not be allowed to download any file from the Internet.
2. Software/services that are not necessary to run in production must be removed.
    1. No development tools, unless specifically required and approved
    2. All binaries used to install any application must be deleted or otherwise removed
        1. i.e., using a multi-stage build, deleting after installation, etc. from the final container (e.g., installers, .rpm, etc.)

   3. Disable unused features and modify default configurations to reduce the possible security breach footprint.

3. No unnecessary software. Examples of unnecessary software include but are not limited to:
    1. SSH clients
    2. Audio or video libraries (unless required)
    3. Development packages
    4. Third-party repository managers
    5. Different shells (zsh, csh, etc.) or text formatters
    6. Additional software examples can be found in the Unified Compliance Framework: 00880 Series (Hardening Procedures) document.
4. Repository and container build instructions must be free of sensitive data and should be idempotent.
    1. Images and Repo1 repositories must not include embedded Secrets.
    2. Ensure that Kubernetes Secrets and/or Secrets Vault are used to store Secrets.
    3. Use arguments, environment variables, and configmaps for configurations as much as possible.
        1. Configuration files intended to be modified in the container file system should not be used.

4.2.2 Scanning Requirements

1. Containers must not have its content encrypted prior to hardening, Iron Bank pipeline, or for the purpose of circumventing scanners.
2. All built containers and downloaded files must undergo a virus scan and results review.
3. Containers must adhere to all Iron Bank configuration policies for container scanning or justify why it is not required.

4.2.3 Related Documents and Labels Requirements

1. Container documentation must be copied into the repository under a “documentation” folder in Iron Bank Repo1.
2. Contributor or Hardener must provide a README file that describes basic usage and function of the application.
    1. File should be in .md format
3. Each repository must contain a LICENSE file that describes the End User License Agreement (EULA)
    1. Examples include GNU V3, APACHE, etc.
4. Containers must have appropriate Open Container Initiative (OCI) compatible labels defined in the hardening_manifest.yaml structured data file.

4.2.4 Iron Bank Compliance Requirements

1. Containers must be based on an approved, hardened base image, and hosted in Iron Bank.
    1. Approved images and versions as of Q42021 are detailed in *Section 2.4, Table A*
2. If the base image contains critical security flaws, attempts must be made to mitigate by applying security hardening, configuration changes, etc. per *DISA DevSecOps Enterprise Container Hardening Guide v1.1* using the timelines in *Section 4.3, Table B*
    1. If the flaw is not corrected, the impacted library/binary must be removed or documented as 'risk accepted' by Iron Bank approvers.
    2. Alternatively, vendors can choose to base their containers on a different base image, all of those that are acceptable are listed in *Section 2.4, Table A*
3. Findings justifications must be submitted by the deadline.
    1. Deadlines are outlined in *Section 4.3, Table B*
4. Containers must comply with all applicable OS DISA STIGs and there is no reasonable way to comply without breaking functionality of the container or application
    1. This is typically achieved using an approved base image.
    2. Approved base images can be found in *Section 2.4, Table A*.
5. Compliance with any applicable application STIG is recommended.
6. All logs generated by the container must be sent to STDOUT
7. When applicable and in accordance with the *DISA DevSecOps Enterprise Container Hardening Guide v1.1*, containers must support TLS 1.2 or 1.3 for all SSL/TLS encryption.
    1. Exceptions may be approved or conditionally approved while the TLS version is upgraded to a compliant version
8. All vulnerabilities must be remediated or mitigated in all supported versions of the container, as detailed **in table in** *Section 4.3, Table B* UNLESS
    1. It's a false positive, and/or the finding is publicly disputed with the official CVE tracking organization.
    2. It's officially withdrawn from the official CVE tracking organization, and the finding was not disputed by the container contributor organization.
    3. The installation/configuration in the Iron Bank container is proven to not be vulnerable and documentation is provided.
    4. The finding does not have a remediation or mitigation available (provided the software is not EoL and actively being developed and/or maintained).
9. A contributor must not grant or set overly permissive file permissions. Examples include:
    1. Setting a SUID or GUID bit on any file.
    2. Allowing "other" write on system directories or files or normally restricted locations or files for which they normally would not have this access.

        1. Examples include:
            1. /etc/*
            2. /var/*
            3. /root/
            4. etc.
    3. Changing any read-only file (0400, 0440, etc) to a writeable file
    4. Using "777" as a file permission
    5. Granting the group "other" execute permissions on any file outside of /bin or /usr/bin
    6. Etc.

4.2.5 Support Requirements

1. Containers submit to Iron Bank should have current upstream support (e.g., commercial, or open source).
2. If an organization has two license options, both containers must be provided to Repo One. The organization cannot only provide the paid version. The DoD Hardening Team will not accept enterprise containers if the open-source container is not also provided.
    1. Example: Both an open source and an enterprise managed version of the same tool.

4.2.6 Future Requirements

The following requirements are expected to be enforced in a future version of the Iron Bank Acceptance Baseline Criteria. A tentative implementation target is early 2023.

- All containers are required to use FIPS compliant encryption. This applies to all containers that require, provide, utilize, or otherwise need encryption
    - All approved base container images have FIPS enabled and configured
    - Contributor and vendor containers must not remove or disable FIPS or compliant encryption algorithms. All additional software added to the base image must also be FIPS compliant.

⚠ **WARNING**:
Failure to submit adequate justification, mitigation (if applicable), or remediation within SLAs will result in delays or the container not being added to Iron Bank and/or will be considered out of compliance with Iron Bank standards.

NOTE:
A container may be not added to Iron Bank even if one of the above items is true. Iron Bank must not enumerate every possible situation. For example, if an upstream package has a CRITICAL finding and the contributor/developer determines they will not fix the issue

4.3 Additional Information

Table A - Approved Base Images in Iron Bank

The following Base images are currently approved by Iron Bank. Additional base images may be approved in the future

Name	Image
UBI8	registry1.dso.mil/ironbank/redhat/ubi/ubi8
UBI8 - Minimal	registry1.dso.mil/ironbank/redhat/ubi/ubi8-minimal
Distroless (base)	registry1.dso.mil/ironbank/google/distroless/base
Distroless (CC)	registry1.dso.mil/ironbank/google/distroless
Distroless (Java 8)	registry1.dso.mil/ironbank/google/distroless/java-8
Distroless (Java 11)	registry1.dso.mil/ironbank/google/distroless/java-11
Distroless (static)	registry1.dso.mil/ironbank/google/distroless/static

Table B - Vulnerability Lifecycle Timelines and SLAs

Severity	CVSS Score	Justification	Max Count	Mitigation/Remediation	Remediation CVE Age Tolerance (Published Date)
Critical	9.0 - 10.0★	Within 5 calendar days of detection	1 finding	**Mitigate** or **remediate** within **15 calendar days** from the date of detection	***Remediate*** in less than 90 calendar days ***of CVE discovery***
High	7.0 - 8.9	Within 10 calendar days of detection	4 findings	**Mitigate** or **remediate** within **35 calendar days** from the date of detection	***Remediate*** in less than 180 calendar days ***of CVE discovery***
Medium	4.0 - 6.9	Within 30 days of detection	-	**Mitigate** or **remediate** within **180 calendar days** from the date of detection	-
Low	0.1 - 3.9	Within 60 calendar days of detection	-	**Mitigate** or **remediate** within **360 calendar days** from the date of detection	-

⚠ ★ On a case-by-case basis for truly severe and impacted containers, Iron Bank will work with vendors to get remediations pushed more quickly than the Critical line above. An example of this was Log4J. In this case the remediation time frame will be much shorter, but this will be an "all hands-on deck" situation.

↑ In this case if Iron Bank has not reached out, please let us know immediately when you are aware your product is affected or not affected and why. We will work with each vendor to ensure the patches or mitigations make it into the container artifacts ASAP.

NOTE:
A conditional acceptance for specific vulnerabilities may be granted on a case-by-case basis beyond the above limits, when applicable. However, additional security requirements may be included until the vulnerabilities are resolved.

NOTE:
False positives and policy exceptions are not covered in the **resolution** table.

- - -BLANK- - -

Table C - Justifications

Justification: Upstream Contributor/Package Manager	Finding Justification Guidelines	Additional Information
False Positive	No mitigation or remediation required	False positives include items that a scanner incorrectly identifies such as a wrong package or version. This does **NOT** include findings that are mitigated or "not exploitable".
Disputed	No mitigation or remediation required	Issues marked as DISPUTED within the NVD. This does **NOT** include issues a contributor is disputing. It must be marked as such within the NVD.
Won't Fix	Must be mitigated; must be remediated	Upstream (not OS distribution such as Redhat, Debian, Ubuntu, etc.) states they will not fix the security flaw.
Distro - Won't Fix	No mitigation or remediation required	Issues marked as WONT_FIX by the vendor. Reserved for OS distributions packages.
No Fix Available	Must be mitigated; must be remediated	There is no patch available. This **ONLY** considers the vulnerable library itself, not downstream products.
Distro - Pending Resolution	Must be mitigated; must be remediated	Vulnerability is for a library provided by the Operating System (OS) distribution. Only applicable when using the latest version of a distribution and library.
Mitigated	Mitigation is complete; must be remediated	Issue has a mitigation that reduces severity or risk.
Not Vulnerable	Mitigation is complete; must be remediated	Issue is not exploitable within application.

Unreleased	Must be mitigated; must be remediated	Fix is available in a branch for the next release but is not yet available.
Pending Resolution	Must be mitigated; must be remediated	Upstream project is aware of vulnerability and is tracking an issue ticket to fix.
True Positive	Must be mitigated; must be remediated	Image is vulnerable to this finding. Default state of a new finding.
Policy N/A	No mitigation or remediation required	Product functionality requires security policy exception. (Only applies to policy findings, not CVEs.)

All finding resolutions and justifications will be approved for Big Bang ***or*** on a case-by-case basis for other containers and the context of the finding and complexity of the fix will be taken into account.

4.3.1 General Guidelines

Approvers will reach a decision regarding each container via a process that depicts the number and severity of findings from the reporting performed in *Section 4.3, Table B*. For example, if the quantity of findings is high, but the quality of those findings is of medium and/or low severity, Approvers may reject the container altogether due to the magnitude of findings. Similarly, if the Approver determines a finding is critical and the container has a high chance of ending up on a public-facing system the container may be rejected. Conversely, reports that demonstrate a container has only a few low/medium findings may mean the container is approved entirely. The level of acceptable risk for cybersecurity findings that cannot be remediated at time of processing will be decided on a case-by-case basis.

Because some containers must have configurations that bypass security requirements outlined above, they may be approved where other containers may be rejected for the same finding. For example, Istio and Container Network Interface (CNI) pods must run as root- but other containers must not. All containers will be assessed on a case-by-case basis; the above criteria are the minimum baseline standard.

In addition, containers are not intended to protect against DDoS or EM side channel attacks – it is expected the *deployment specifications* will address these concerns. (IE: Kubernetes rate limiting + pod restarts for denial-of-service attacks, secured location for side channel, etc.)

4.3.2 Remediation Guidelines

A finding is considered remediated when it no longer appears on routine scans. This is accomplished using any of the below methods:

- The vulnerable component/dependency is updated to a non-vulnerable version
- The vulnerable component is removed or replaced with one that doesn't have the vulnerability
- The vulnerable function or component has been removed
    - Example: Use Apache instead of a built-in version of Tomcat to serve Web content

4.3.3 Mitigation Guidelines

Findings are considered "mitigated" when any of the below are true:

- The vulnerable component cannot be accessed or exploited by a normal user, unauthenticated, via any remote method, locally by any user other than `root`, and is not a vulnerability that allows privilege escalation or extends a user’s privileges, does not allow the reading of memory contents outside of the buffer, etc.
- The vulnerable component has been disabled
- The access to or the ability of the component to be exploited has been disabled, or specific filetypes an exploit relies on, and cannot be restored.

In addition, a timeline for full remediation and a simple explanation of how the vulnerability was mitigated must be provided in the justifications.

4.3.4 Compliance Guidelines

Iron Bank uses NIST-standard, open-source tool *OpenSCAP*, which may be freely downloaded-it uses standardized baseline files to complete automated compliance checking.

All compliance findings must be remediated. Some examples of common findings are:

- File/Folder permissions - file permissions should only be changed from the standard if the application requires it.
- Secrets - remove all Secrets

A finding is considered **delinquent** if:

- A mitigation or remediation has not been applied within the general/mitigated guidelines
- A justification has not been submitted within the justification guidelines
- A finding is older than the maximum grace period

4.3.5 General Justification Guidelines/Templates

When providing justification for vulnerabilities, it's important to provide consistent information that can be noted for future reference and provide context. Below some guidelines have been provided, however contributors should think critically about what should be included- this list is not exhaustive.

- **MEDIUM** or **LOW** Vulnerability:
    - Vulnerability will be fixed in release x.y.z by MM/YYYY or Quarter # of 202#
    - Vulnerability cannot be exploited because ____________________
    - There is no update for vulnerable component as of MM/YYYY which is maintained and/or released by vendor/organization ABC
    - Vulnerable component cannot be updated and is version locked because of another component. This will be updated in release x.y.z by MM/YYYY or Quarter # of 202# and the vulnerable component will be upgraded as well

- **HIGH** or **CRITICAL** Vulnerability:
    - Why the vulnerability is mitigated or not exploitable or why the scan is incorrect
    - There is no update for vulnerable component as of MM/YYYY which is maintained and/or released by vendor/organization ABC
    - Vulnerability will be fixed in release x.y.z by MM/YYYY or Quarter # of 202# (must be within the timeframe as discussed above)

- Findings can be justified as ***not vulnerable*** if:
    - The vulnerable component is not used, or requires an unused integration
    - A scanner is mistaken or affects a different OS or CPU/Instruction architecture

4.4 Iron Bank Container Status in Iron Bank

Containers may have different statuses based on why the container is being hosted by Iron Bank. Iron Bank will only approve containers for use when they are part of Big Bang (core and addons). Vendor and open source containers will only be rated against whether they are compliant or not with the hardening standards in this document.

In addition, there are different approval levels based on the whether the container has been reviewed by the container approvers team and if the container is part of the Big Bang Core/Addons packages. These statuses are listed below and the analysis and meaning for each status.

1. **Big Bang (Core + Addon) Status:**
    1. **Approved**:
        - The container image is compliant with the hardening standards in this document and has been manually reviewed by a member of the Container Approver team
    2. **Conditionally Approved:**
        - The container image is pending review due to non-compliance with the hardening standards in this document
        - The container image temporarily approved while the contributor works towards getting the container compliant with the hardening standards in this document and the approval expiration date will be displayed on Iron Bank
        - This container image is not compliant with the hardening standards in this document **and** the container approver team has approved justifications as to why it is not reasonably achievable for the container image and/or mitigations were approved for applicable findings.

        Deployment Mitigation Suggestions

        In this case, any mitigations a customer or program should take will be listed next to the conditional approval in Iron Bank (eg. Must be behind a firewall/internal use only, etc.

    3. **Rejected / Approval Pulled**
        1. This status is not expected to occur often and only when it is necessary to pull a container that has a clear and present danger associated with its continued use.
    4. **EOL (End of Life)**
        1. A container repository is EoL upstream or EoL'd by a vendor and will be replaced by a new product to fill the functionality, if necessary for the Big Bang product.
2. **Open source/Vendor Containers that are NOT part of Big Bang (Core + addons)**
    1. **Compliant**
        1. The container image is compliant with the hardening standards in this document

2. **Non-Compliant**
    1. This container image is not compliant with the hardening standards in this document
3. **Rejected/Pulled**
    1. This status is not expected to occur often and only when it is necessary to pull a container that has a clear and present danger associated with its continued use.
4. **EOL**
    1. A container repository is EoL upstream or EoL'd by a vendor
    2. A container that has not been maintained in 6-9 months is in jeopardy of being marked as EoL as it is no longer receiving updates and may be pulled

5. Summary

This document aims to clarify specific requirements and standards that most containers should be compliant in order to reduce the attack surface of running a specific container and how containers will be assessed in a standardized manner.

While there will always be a handful of exceptions and edge cases, these are rare, and must be well-documented. All published standards follow best practices for the relevant technology or service, and all findings must be remediated as stated, or justified. Not all findings are eligible for sign-off, and/or may require a fix within a certain timeframe or by a fixed release-furthermore, containers may still be rejected even if all guidelines are followed based on the discretion of Iron bank

Conditional acceptance of specific vulnerabilities *may* be granted in some select cases. However, any hosted containers are subject to withdrawal, as deemed appropriate and containers expire automatically when the software is no longer supported by the contributor or vendor, if these entities are different. Finding denials are final - they cannot be challenged, appealed, or otherwise re-adjudicated.

Enrichment of risk data is available to AOs via the *Overall Risk Assessment (ORA) process*. Note that this process also cannot be appealed, as it enhances the visibility of potential risk to the availability, integrity, or confidentiality of information systems (IS) or their contained data based on similar metrics as the **Open-Source Security Foundation** (OSSF).

> It is always the responsibility of the program or organization using each container and/or software product to evaluate and accept or reject the risks of running it based on their criteria.

6. References

Container Hardening Guide Version 1.1

Container Image Creation and Deployment Guide Version 2, Release 0.6

Unified Compliance Framework: 00880 Series (Hardening Procedures)

DoD DevSecOps Reference Design

Open Source Security Foundation (OSSF)

OpenScap

DCCSCR Readme

Overall Risk Assessment (ORA)

Iron Bank documentation