Security Standard – Containerisation (SS-011)

Chief Security Office

Date: 30/05/2024

Department for Work & Pensions

This Containerisation Security Standard is part of a suite of standards, designed to promote consistency across the Department for Work and Pensions (DWP), and supplier base with regards to the implementation and management of security controls. For the purposes of this standard, the term DWP and Authority are used interchangeably.

Technical security standards form part of the DWP Digital Blueprint which is a living body of security principles, architectural patterns, code of practice, practices and radars, that aim to support Product Delivery Units (PDUs) and suppliers in delivering the DWP and HMG Digital Strategy. Security standards and policies considered appropriate for public viewing are published here:

https://www.gov.uk/government/publications/dwp-procurement-security-policies-and-standards.

Technical security standards cross-refer to each other where needed, so can be confidently used together. They contain both mandatory and advisory elements, described in consistent language (see table below).

Table 1 – Terms

Term	Intention
must	denotes a requirement: a mandatory element.
should	should denotes a recommendation: an advisory element.
may	denotes approval.
might	denotes a possibility.
can	denotes both capability and possibility.
is/are	is/are denotes a description.

1. Table of Contents

2. Revision history

Version	Author	Description	Date
1.0		First published version	18/09/2017
2.0		Full update in line with current best practices and standards; • Updated Intro, purpose, audience, scope • Shortened overview of containerisation • Written to be vendor and technology agnostic as far as possible to increase applicability • Replaced use of technical control requirements to minimum security measures • Re-formatted document to categorise security measures under five headings to correspond with each core containerisation component • Added new requirements (security measures) under each category • Added NIST sub-category references against each security measure • Added new table in Appendix A which list security outcomes the measures support the achievement of • Updated references and included links to external publications etc. 9.3.4 Added reference to build team. 9.4.4 Replaced CMDB with code repository or container registry.	22/08/2022

2.1		All NIST references reviewed and updated to reflect NIST 2.0 All security measures reviewed in line with risk and threat assessments References added to other standards; - Server Operating System - Network Security Design standard - Privileged User Access - User Access Controls - Software Development - Security Patching - Security Incident Management 11.1.5, 11.1.8 & 11.1.10 ‘should’ changed to ‘must’ 11.1.11 Communications monitoring 11.2.1 Orchestrator browse up 11.2.2 ‘considered’ changed to ‘utilised’ 11.2.10 Misconfiguration 11.3.9 Selecting library or image names 11.3.11 Security controls on base images 11.5.5 Orchestrator platform; Disallowing container creation 11.5.12 Container scanning 11.5.21 ‘should’ changed to ‘must’ 11.5.23 Exposed APIs	30/05/2024

3. Approval history

Version	Name	Role	Date
1.0		Chief Security Officer	18/09/2017
2.0		Chief Security Officer	22/08/2022
2.1		Chief Security Officer	30/05/2024

This document is continually reviewed to ensure it is updated in line with risk, business requirements, and technology changes, and will be updated at least every 2 years - the current published version remains valid until superseded.

4. Compliance

Compliance with this standard will be verified through various methods, including but not limited to;

- controls tests performed by first-line teams and by 2nd line activities (e.g. security testing teams)
- security assurance activities to ensure that Architectural Design and delivery are appropriate and aligned to applicable Authority Security Standards. [See Security Assurance Strategy – Ref. L].
- independent external audit

Results of these will be fed back to the appropriate Authority Risk and System Owners.

5. Exceptions Process

In this document the term **“must”** is used in bold letters to indicate a mandatory security measure. Any exceptions to the application of this standard, or where specific security measures cannot be adhered to, **must** be presented to the Authority. This **must** be carried out prior to deployment and managed through the design caveats or exception process.

Such exception requests will invoke the Risk Management process to clarify the potential impact of any deviation to the configuration detailed in this standard.

Exceptions to the standard **must** be maintained on a risk register for accountability, traceability, and security governance reporting to senior management.

6. Audience

This document is intended for, but not necessarily limited to, technical architects, technical engineers, developers, security teams, project teams, including suppliers engaged in the design, development, implementation and operation of systems, services and applications that utilise containerisation technology.

7. Accessibility statement

Users of this standard **must** consider accessibility design requirements as appropriate. Further information on accessibility standards can be found in Appendix F.

8. Introduction

This standard defines the minimum technical security measures that **must** be implemented to secure Authority systems and data utilising application containerisation technology.

For the purposes of this standard, containerisation can be described as an operating system (OS) level virtualisation method where applications run in isolated user spaces, called containers, while using the same shared host. A container is essentially a stand-alone, all-in-one package for a software application. They contain everything that an application needs, such as its libraries, binaries, configuration files and software dependencies, all encapsulated into an independent, self-contained unit. The container itself is abstracted from the host OS, with only limited access to underlying resources (if configured securely).

As this standard only provides minimum measures, they **should** be exceeded as appropriate depending on the threats and risks that need to be addressed, the sensitivity of the data, and in keeping with latest security enhancements.

The security measures are derived from industry best practice i.e. guidance published by NIST, CIS and OWASP (see Appendix C for full list external references) and support the implementation of appropriate security controls as selected by the Authority or our third party providers, such as the CIS Critical Security Controls v8 controls set. [see External References]

Every effort has been made to ensure the security measures are vendor and technology agnostic as far as possible; this is to ensure greater applicability of the standard regardless of the technologies used. The security measures **may** be implemented in different ways, depending on the technology choices and business requirements in question.

The aim of this standard is to:

- ensure security controls that are applicable to core containerisation components are implemented consistently across the Authority and by third party providers where applicable.
- mitigate risks from common threats and vulnerabilities associated with containerisation technology, to an acceptable level for operation.
- support the achievement of security outcomes described in Appendix A.

Technical security standards ultimately support the achievement of security outcomes sought by the Authority. They set the expectations for what needs to be done to achieve them, and why. The outcomes are based on the official NIST sub-categories where possible to ensure close alignment with the NIST Cyber Security Framework (CSF) and can be found in Appendix A of every technical security standard.

9. Purpose

The purpose of this standard is to ensure systems and services utilising application containerisation technology to process Authority data are designed, configured, deployed, and managed consistently to protect against typical threats at the OFFICIAL tier.

This standard also serves to provide a baseline in which assurance and compliance activities can be carried out, so that the Authority can be assured that security obligations are being met or exceeded.

10. Scope

This standard applies to all use of containerisation technology within the Authority and supplier base (contracted third party providers), for the purposes of delivering applications and services that handle Authority data.

All forms of virtualisation other than technologies that support containerisation are outside the scope of this document.

Also, this standard only addresses the core components of containerisation technology - platform (host OS), orchestrators, images, registries (repositories) and containers. Because this standard only looks at the core components, the measures should be applicable to most container deployments regardless of the container technology or vendor, host OS platform, or location i.e. public or private cloud.

Any queries regarding the security measures laid out in this standard **should** be sent to the Authority.

11. Minimum Technical Security Measures

The following section defines the minimum security measures that **must** be implemented to achieve the security outcomes described in Appendix A. For ease of reference, the official NIST sub-category ID is provided against each security measure e.g. PR.PT-3, to indicate which outcome(s) it contributes towards. Refer to Appendix A for full description of outcomes.

11.1 Platform Hardening (host)

Reference	Minimum Technical Security Measures	NIST ID
11.1.1	The OS's supporting the containers **must** be in vendor support and hardened in accordance with SS-008 - Server Operating System Security Standard [Ref. A] or an approved 'Gold Build' (where applicable) to reduce the attack surface of the host as much as possible. Industry benchmarks **must** be used where available e.g. CIS Docker Benchmarks.	PR.PS-01
11.1.2	Container-specific OS's **should** be used whenever possible instead of general-purpose ones to reduce the attack surface. These are specifically designed to host containers and have other services and functionality disabled by default, providing mitigation against typical risks and hardening activities associated with general purpose operating systems (OS). See SS-008 Server Operating System Security Standard [Ref. A] for further information on securing OSs.	PR.PS-01
11.1.3	Containers **must** not be used as a method to separate data or services that have different security profiles.	ID.AM-05
11.1.4	Hosts **must** be set up such that, by default, network stacks within the containers on the host cannot inter-communicate unless an approved[1] pattern is used. When containers are run, they **must** obtain their own individual network stack. See SS-018 Network Security Design Security Standard [Ref. N] for more information.	PR.IR-01

[1] Approved patterns are those formally approved by the DWP Digital Design Authority and published as part of the DWP Digital Blueprint.

11.1.5	Hosts that run containers **must** only run containers and not run other applications, like web servers or databases outside of containers. The host OS **must** not run unnecessary services, such as a print spooler, that increase its attack and patching surface. For the avoidance of doubt, this measure does not apply to services/agents deployed on hosts as part of the OS build approved by the Authority. See SS-008 Server Operating System Security Standard [Ref. A] for further information on securing OSs.	PR.IR-01 PR.IR-03
11.1.6	Hosts **must** be continuously scanned for vulnerabilities and updates applied in accordance with the DWP Technical Vulnerability Management Policy [Ref. B] and SS-033 – Security Patching Standard [Ref. C], not just to the container runtime but also to lower level components such as the kernel that containers rely upon for secure compartmentalised operation.	ID.RA-01 RS.MI-01 ID.RA-06
11.1.7	The OS and any associated Gold Build images **must** be kept up to date in accordance with SS-033 - Security Patching Standard [Ref. C], not only with security updates, but also the latest component updates recommended by the vendor. This is particularly important for the kernel and container runtime components as newer releases of these components often add additional security protections and capabilities beyond simply correcting vulnerabilities.	ID.RA-01 PR.PS-02 ID.RA-06
11.1.8	Host OS's **must** be used solely for hosting containers. There **must** also be no application level dependencies provided by the host, instead all components and dependencies **must** be packaged and deployed in containers.	PR.PS-01 DE.CM-03

11.1.9	All authentication to the hosts **must** be audited, and any login anomalies monitored, and any escalations to perform privileged operations **must** be logged in accordance with SS-012 - Protective Monitoring Standard [Ref. I]. This will make it possible to identify anomalous access patterns such as an individual logging on to a host directly and running privileged commands to manipulate containers.	DE.CM-01 DE.CM-03 DE.CM-06 DE.CM-09
11.1.10	Access to the host OS **must** be based on the need-to-have and least privilege principle. See SS-001-2 Privileged User Access Security Standard [Ref. G] for further information on privileged users.	PR.AA-05
11.1.11	A baseline of normal communication activities **must** be created and thresholds on monitoring and logging tools **must** be configured to trigger when events such as traffic spikes, or unusual traffic flows are detected.	PR.PS-01 ID.AM-03

11.2 Orchestrator

Reference	Minimum Technical Security Measures	NIST ID
11.2.1	The principle of least privilege **must** be implemented for Orchestrators, in which users are only granted the ability to perform specific actions on the specific hosts, containers, and images their role requires. Orchestrators must not be permitted to 'Browse Up' from a lower security zone to a higher security zone, without going through the appropriate elevation process. See SS-001-2 Privileged User Access Security Standard [Ref. G] for further information on privileged users.	PR.AA-05 PR.PS-01

11.2.2	Access to cluster-wide administrative accounts **must** be tightly controlled and monitored in line with SS-012 Protective Monitoring Standard [Ref. I], as these accounts provide the ability to affect all resources in the environment. Strong authentication methods **must** be utilised as appropriate, such as requiring multi-factor authentication instead of just a password. See SS-001 Access & Authentication [Ref. F] and SS-001-2 Privileged User Access [Ref. G] Security Standards for further information.	PR.AA-05
11.2.3	Single sign-on **must** be implemented where possible, as this will simplify the orchestrator authentication experience, make it easier for users to use strong authentication credentials, and centralise auditing of access, making anomaly detection more effective. Any elevated access e.g. for administrative purposes, must force a re-authentication of user credentials, which must include a valid Multi Factor Authentication interaction utilising a hard-token. See SS-001 Access & Authentication Security Standard [Ref. F]	PR.AA-03
11.2.4	Orchestrators **must** be configured to separate network traffic into discrete virtual networks based on security profiles where possible. For example, public-facing apps with increased threat exposure could share a virtual network. See SS-018 Network Security Design Security Standard [Ref. N] for more information.	PR.IR-01

11.2.5	Orchestrators **must** be configured to isolate deployments to specific sets of hosts by sensitivity levels in accordance with the DWP Security Classification Policy [Ref. H]. This particular approach will vary depending on the Orchestrator in use, but the general model is to define rules that prevent high sensitivity workloads from being placed on the same host as those running lower sensitivity workloads. See SS-003 Software Development [Ref. M] and SS-018 Network Security Design [Ref. N] Security Standards for more information.	PR.IR-01
11.2.6	Orchestrators **must** ensure that nodes are securely introduced to a cluster, have a persistent identity throughout their lifecycle, and can provide an accurate inventory of nodes and their connectivity states. See SS-003 Software Development [Ref. M] Security Standard for more information.	ID.AM-02
11.2.7	Orchestrators used for managing the build, distribution and run phases of the application container lifecycle **must** be supported by a CMDB or asset inventory. See SS-003 Software Development [Ref. M] Security Standard for more information.	ID.RA-07 PR.PS-01
11.2.8	Clusters **must** be configured to monitor resource consumption patterns of individual containers to aid detection of unanticipated spikes in resource usage that could lead to non-availability of critical resources.	PR.IR-04 DE.CM-01 ID.AM-03

11.2.9	Containers **must** be grouped according to their purpose, sensitivity, and threat posture on a single host OS kernel to allow for additional defence in depth. See SS-003 Software Development [Ref. M] and SS-018 Network Security Design [Ref. N] Security Standards for more information.	PR.IR-01
11.2.10	Care **must** be taken when configuring containers, as misconfiguration is the most commonly exploited vulnerability. All default settings for dashboards, clusters and endpoints **must** be reviewed and appropriately hardened using available benchmarks i.e., CIS, STIG etc., to minimise vulnerabilities due to misconfigurations. Under no circumstance **must** nodes and clusters be deployed using default configurations without assessing the risk implications. See also SS-033 Security Patching Standard [Ref. C].	PR.IR-01 PR.DS-01 PR.PS-01 ID.AM-08

11.3 Images

Reference	Minimum Technical Security Measures	NIST ID
11.3.1	Image configurations **must** be reviewed against secure configuration best practices where available e.g., CIS Benchmarks, to reduce the attack surface.	PR.PS-01
11.3.2	Images **must** be configured to run as non-privileged users where technically possible. Where this cannot be achieved, functionalities such as user namespace remapping **must** be used to map the container user to a non-privileged user on the host OS.	PR.PS-01
11.3.3	Secrets **must** be stored outside of images and provided dynamically at runtime as needed.	PR.AA-01 PR.AA-05
11.3.4	All images regardless of where they are sourced, **must** be vetted, tested, and validated. All images **must** also be digitally signed in accordance with SS-002 - PKI and Key Management Standard [Ref. D] before being added to the image registries. Separation of duties **must** be maintained between the build team and those approving the criteria for acceptance of an image.	PR.DS-01 PR.DS-02
11.3.5	Images **must** be scanned for embedded malware and vulnerabilities, when acquired, before deployment and following significant changes.	ID.RA-01 PR.DS-01 PR.DS-10 PR.PS-05

11.3.6	Offline (stored) images **must** be kept up to date, and all runtime images re-created using the latest images. Furthermore, images **must** be regularly assessed or tested for compatibility with the wider ICT estate as to minimise the risk of ‘breaking’ applications that are still dependent on older software versions. Where updating runtime images with the latest images causes incompatibility issues with a given application, use of older images **must** be subject to a risk assessment and formal risk owner approval.	ID.AM-02 PR.DS-01 DE.CM-09 ID.AM-08 PR.PS-01 ID.RA-05 ID.RA-07
11.3.7	When any changes are made to the base image or dependent image (e.g., patching a vulnerability), the corresponding image **must** be recreated, and the container re-launched using the modified image. This ensure a single master, or gold image is maintained for any service.	PR.PS-01
11.3.8	Routine checks **must** be carried out (at least every 2 weeks as a minimum) to ensure the latest images available are being used. This process **should** be automated where possible.	PR.PS-01
11.3.9	Only approved container images **must** be used as a source (see 9.3.4). Care **must** be taken in selecting base images or libraries, to ensure the correct, valid names are being used.	PR.PS-01
11.3.10	Code base (ideally within source code management) used for image builds **must** be backed up in accordance with SS-035 – Secure Backup and Restore Security Standard [Ref. K].	PR.DS-11 RC.RP-01

11.3.11	Security controls **must** be applied to all base images, whether they are used directly or transitively.	PR.PS-01 PR.PS-06

11.4 Registry (Repository)

Reference	Minimum Technical Security Measures	NIST ID
11.4.1	Connection to Production registries from tools, orchestrators and container runtimes **must** be over encrypted channels in accordance with SS-007 - Use of Cryptography [Ref. E].	PR.DS-02
11.4.2	To mitigate against inadvertently using out of date and potentially vulnerable images, container registries **must** be regularly pruned of images that are no longer required (at least once a quarter). This process **should** be automated where possible.	ID.RA-01 PR.PS-02
11.4.3	Operational teams **must** access images using immutable names that specify the discrete version of images to be used. As such, deployment tasks **must** specify the exact versions to be used.	PR.PS-01 ID.RA-07
11.4.4	Identification of all images and versions **must** be maintained at all times in a code repository or container registry.	PR.PS-01 ID.RA-07
11.4.5	All access to registries **must** be authenticated and authorised in accordance with SS-001 (part 1) - Access and Authentication Controls Standard [Ref. F] and SS-001 (part 2) - Privileged User Access Controls Standard [Ref. G].	PR.AA-03 PR.AA-05 PR.PS-01

11.4.6	Any write access to registries **must** be authenticated to ensure that only images from trusted entities can be added to it.	PR.AA-03 PR.PS-01
11.4.7	Images **must** be approved by authorised personnel and only pushed to a registry after they have passed a security assessment process i.e. passed a vulnerability scan. This process **should** be automated where possible. See SS-003 Software Development Security Standard [Ref. M] for more information.	ID.RA-01 PR.DS-01 PR.IR-01 PR.PS-01 ID.AM-08 ID.RA-07
11.4.8	The number of accounts accessing the registry **must** be limited to mitigate against the threat of account hijacking.	PR.AA-05
11.4.9	Image Registries **must** support signed images.	PR.DS-01 PR.DS-02
11.4.10	Image Registries **must** not allow unrestricted network access.	PR.IR-01
11.4.11	Logging and Alerting **must** be enabled on Image Registries where supported to detect anomalous activity. See SS-014 Security Incident Management [Ref. O] for more information.	PR.PS-04
11.4.12	Threat detection capability **must** be utilised for Image Registries where supported. See SS-014 Security Incident Management [Ref. O] for more information.	DE.CM-01 DE.CM-09

11.4.13	Logs **must** be forwarded from Image Registries to security monitoring tools to support threat detection in accordance with SS-012 - Protective Monitoring Standard [Ref. I]. See SS-014 Security Incident Management [Ref. O] for more information.	PR.PS-04 DE.CM-01 DE.CM-09
11.4.14	Artefacts maintained within image registries and associated meta data **must** be backed up in accordance with SS-035 – Secure Backup and Restore Security Standard [Ref. K].	PR.DS-11 RC.RP-01

11.5 Containers

Reference	Minimum Technical Security Measures	
11.5.1	Mandatory access control (MAC) technologies **must** be considered as appropriate, to provide enhanced control and isolation for containers.	PR.AA-05
11.5.2	Separate environments **must** be used for development, test, production, and other scenarios, each with specific controls to provide role-based access control for container deployment and management activities. See SS-003 Software Development Security Standard [Ref. M] for more information.	PR.AA-05
11.5.3	Container runtimes **must** be monitored for vulnerabilities, and when problems are detected, they **must** be remediated (in accordance with SS-033 – Security Patching Standard [Ref. C]). A vulnerable runtime exposes all containers it supports, as well as the host itself, to potentially significant risk. Security tools **should** be used to look for CVE vulnerabilities in runtimes deployed, to ensure that Orchestrators only allow deployments of properly maintained runtimes.	ID.RA-01 DE.CM-09

11.5.4	Systems administrations **must** apply the default deny rule to all container capabilities, and only allow those capabilities needed through an explicit ‘Allow List’.	PR.PS-01
11.5.5	All container creation **must** be associated with individual user identities and logged to provide a clear audit trail of activity. Only the Orchestrator Platform is permitted to create containers; all other containers **must** be set up to disallow further container creation.	PR.AA-01 PR.AA-05 PR.PS-04 DE.CM-03
11.5.6	SSH / RDP and other administration tools designed to provide remote shells to host **must** be disabled within containers.	PR.IR-01 PR.AA-03 PR.AA-05
11.5.7	Although created inside a container, all logs **must** be managed by a process executing outside the container and **must** not be managed by a process running inside the container.	PR.PS-04
11.5.8	As part of the container configuration, commands and capabilities not required to support the service provided by the container **must** be removed or disabled.	PR.PS-01
11.5.9	Containers **must** externally present only the necessary ports and services required by the consuming business or administrative services.	PR.PS-01 PR.AA-06 PR.IR-01
11.5.10	Network specific operations **must** be disabled inside containers. Network configuration **must** be applied to the container at start-up and not be dynamically modified.	PR.PS-01 PR.AA-06 PR.IR-01

11.5.11	Under no circumstance **must** containers be able to mount sensitive directories on a host's file system, especially those containing configuration settings for the operating system.	ID.AM-03 PR.IR-01 PR.DS-02 PR.DS-01 PR.DS-10 PR.PS-01
11.5.12	To mitigate malicious network activity related to packet spoofing, access to raw sockets **must** not be allowed within the container. Additionally, containers **must** be regularly scanned to ensure that network scanning tools such as pnscan, masscan and zgrab are not present.	ID.AM-03 PR.IR-01 PR.DS-02 DE.CM-01
11.5.13	Run File systems in containers **must** be read only to prevent malicious scripts being saved or files being overwritten.	PR.AA-05
11.5.14	Containers **must** not be allowed to load modules dynamically. All code that is required to execute within the container **must** be within the container image.	PR.PS-01
11.5.15	During the build process, the identity of all dependencies **must** be verified and authenticated using code signing and signatures.	PR.DS-01 PR.DS-02 DE.CM-09
11.5.16	Build processes **must** enforce the use of the most up to date image dependencies, where appropriate.	PR.IR-01 ID.AM-03
11.5.17	Members of the test team **must** only be given access to images in a test environment and the hosts used for running them and **should** only be able to manipulate the containers they created.	PR.AA-05

11.5.18	If an application has multiple components that need to run distinctly from one another, then each component **should** be deployed in its own container.	PR.PS-01
11.5.19	Services between containers or groups of containers, **must** be exposed only via port binding, with ports explicitly opened in a container configuration file, specifying that the only permitted connection to a given application is from another container.	ID.AM-03 PR.IR-01 PR.AA-03 PR.AA-05 PR.DS-01 PR.DS-02 PR.DS-10 PR.PS-01 DE.CM-01
11.5.20	Containers **must** be configured in accordance with the requirements set out in SS-012 - Protective Monitoring Standard [Ref I].	PR.PS-04 PR.PS-06 DE.CM-09
11.5.21	All diagnostics in production **must** be done via log files or other approved tooling which negates direct access to running containers.	PR.PS-04
11.5.22	If supported, the container runtime **must** be configured to enforce running signed images only, this will prevent images from external, un-vetted sources from being used.	PR.DS-01 PR.DS-02
11.5.23	Exposed APIs **must** be minimised or removed altogether, to reduce the likelihood of exploitation of external container remote services.	PR.PS-02

12. Appendices

Appendix A. Security Outcomes

The minimum security measures defined in this standard contribute to the achievement of security outcomes described in the table below. For consistency, the official NIST Sub-category IDs have been carried through to the standards which can also be cross referenced against the CIS Critical Security Controls.

Table 2 – List of Security Outcomes Mapping

Ref	Security Outcome (Sub-category)	Related Security Measure
ID.AM-02	Inventories of software, services, and systems managed by the organization are maintained	11.3.6
ID.AM-03	Representations of the organization's authorized network communication and internal and external network data flows are maintained Inventories of services provided by suppliers are maintained	11.1.11, 11.2.8, 11.5.11, 11.5.12, 11.5.16, 11.5.19
ID.AM-05	Assets are prioritized based on classification, criticality, resources, and impact on the mission	11.1.3
ID.AM-08	Systems, hardware, software, services, and data are managed throughout their life cycles	11.2.10, 11.3.6, 11.4.7
ID.RA-01	Vulnerabilities in assets are identified, validated, and recorded	11.1.6, 11.1.7, 11.3.5, 11.4.2, 11.4.7, 11.5.3
ID.RA-05	Threats, vulnerabilities, likelihoods, and impacts are used to understand inherent risk and inform risk response prioritization	11.3.6

ID.RA-06	Risk responses are chosen, prioritized, planned, tracked, and communicated	11.1.6, 11.1.7
ID.RA-07	Changes and exceptions are managed, assessed for risk impact, recorded, and tracked	11.2.7, 11.3.6, 11.4.3, 11.4.4, 11.4.7
ID.RA-08	Processes for receiving, analyzing, and responding to vulnerability disclosures are established	11.1.6
PR.AA-01	Identities and credentials for authorized users, services, and hardware are managed by the organization	11.3.3, 11.5.5
PR.AA-03	Users, services, and hardware are authenticated	11.2.3, 11.4.5, 11.4.6, 11.5.6, 11.5.19
PR.AA-05	Access permissions, entitlements, and authorizations are defined in a policy, managed, enforced, and reviewed, and incorporate the principles of least privilege and separation of duties	11.1.10, 11.2.1, 11.2.2, 11.3.3, 11.4.5, 11.4.8, 11.5.1, 11.5.2, 11.5.5, 11.5.6, 11.5.13, 11.5.17, 11.5.19
PR.AA-06	Physical access to assets is managed, monitored, and enforced commensurate with risk	11.5.9, 11.5.10
PR.DS-01	The confidentiality, integrity, and availability of data-at-rest are protected	11.2.10, 11.3.4, 11.3.5, 11.3.6, 11.4.7, 11.4.9, 11.5.11, 11.5.15, 11.5.19, 11.5.22
PR.DS-02	The confidentiality, integrity, and availability of data-in-transit are protected	11.3.4, 11.4.1, 11.4.9, 11.5.11, 11.5.12, 11.5.15, 11.5.19, 11.5.22
PR.DS-10	The confidentiality, integrity, and availability of data-in-use are protected	11.3.5, 11.5.11, 11.5.19
PR.DS-11	Backups of data are created, protected, maintained, and tested	11.3.10, 11.4.14

PR.PS-01	Configuration management practices are established and applied	11.1.1, 11.1.2, 11.1.8, 11.1.11, 11.2.1, 11.2.7, 11.2.10, 11.3.1, 11.3.2, 11.3.6, 11.3.7, 11.3.8, 11.3.9, 11.3.11, 11.4.3, 11.4.4, 11.4.5, 11.4.6, 11.4.7, 11.5.4, 11.15.8, 11.5.9, 11.5.10, 11.5.11, 11.5.12, 11.5.14, 11.5.18, 11.5.19
PR.PS-02	Software is maintained, replaced, and removed commensurate with risk	11.1.7, 11.4.2, .5.23
PR.PS-04	Log records are generated and made available for continuous monitoring	11.4.11, 11.4.13, 11.5.5, 11.5.7, 11.5.20, 11.5.21
PR.PS-05	Installation and execution of unauthorized software are prevented	11.3.5
PR.PS-06	Secure software development practices are integrated, and their performance is monitored throughout the software development life cycle	11.3.11, 11.5.20
PR.IR-01	Networks and environments are protected from unauthorized logical access and usage	11.1.4, 11.2.4, 11.2.5, 11.2.9, 11.2.10, 11.4.7, 11.4.10, 11.5.6, 11.5.9, 11.5.10, 11.5.11, 11.5.12, 11.5.16, 11.5.19
PR.IR-04	Adequate resource capacity to ensure availability is maintained	11.2.8
DE.CM-01	Networks and network services are monitored to find potentially adverse events	11.1.9, 11.2.8, 11.4.12, 11.4.13, 11.5.12, 11.5.19
DE.CM-03	Personnel activity and technology usage are monitored to find potentially adverse events	11.1.8, 1.1.9, 11.5.5
DE.CM-06	External service provider activities and services are	11.1.9

	monitored to find potentially adverse events	
DE.CM-09	Computing hardware and software, runtime environments, and their data are monitored to find potentially adverse events	11.1.9, 11.3.6, 11.4.12, 11.4.13, 11.5.3, 11.5.15, 11.5.20
RS.MA-01	The incident response plan is executed in coordination with relevant third parties once an incident is declared	11.4.14
RC.RP-01	The recovery portion of the incident response plan is executed once initiated from the incident response process	11.3.10, 11.4.14

Appendix B. Internal references

Below, is a list of internal documents that **should** be read in conjunction with this standard.

Table 3 – Internal References

Ref	Document	Publicly Available*
A	SS-008 – Server Operating System	Yes
B	DWP Technical Vulnerability Management Policy	Yes
C	SS-033 – Security Patching	Yes
D	SS-002 – PKI and Key Management	Yes
E	SS-007 – Use of Cryptography	Yes
F	SS-001 (part 1) – Access and Authentication Controls	Yes
G	SS-001 (part 2) – Privileged User Access Controls	Yes
H	DWP Security Classification Policy	Yes
I	SS-012 - Protective Monitoring Standard	Yes
J	SS-015 – Malware Protection	Yes
K	SS-035 – Secure Backup and Restore	tbc
L	DWP Security Assurance Strategy	No
M	SS-003 Software Development	Yes
N	SS-018 Network Security Design	Yes
O	SS-014 Security Incident Management	Yes

Requests to access non-publicly available documents* ***should* *be made to the Authority.*

Appendix C. External references

The following publications and guidance were considered in the development of this standard and **should** be referred to for further guidance.

Table 4 – External References

External Documents List
Amazon Elastic Container Service Best Practices Guide
NIST – Cyber security Framework – 2018-04-16
NIST – 800-53 – Rev 5 – Security and Privacy Controls for Information
NIST Special Publication 800-190 – Application Container Security Guide (September 2017)
NIST 8176 – Security Assurance Requirements for Linux Application Container Deployments
CIS v8 Critical Security Controls
CIS Docker Benchmark (Version 1.3.0)
ISO/IEC 27002:2013
Cloud Security Alliance Cloud Controls Matrix Version 4
OWASP Application Security Verification Standard (ASVS)
OWASP Container Security Verification Standard (Version 1.0)
UK Government Technical Standard
National Security Agency – Kubernetes Hardening Guidance (Version 1.0)

Appendix D. Abbreviations

Table 5 – Abbreviations

Abbreviation	Definition	Owner
CIS	Centre for Internet Security	Industry body
CMDB	Configuration Management Database	Industry term
CVE	Common Vulnerabilities and Exposures	Industry term
DWP	Department of Work and Pensions.	UK Government
GSCP	Government Security Classification Policy	UK Government
ISO	International Organization for Standardization	Industry term
MAC	Mandatory Access Control	Industry term
NIST	National Institute of Standards and Technology	US Government
NIST – CSF	National Institute of Standards and Technology – Cyber Security Framework	US Government
OS	Operating System	Industry term
OWASP	Open Web Application Security Project	Open source
OWASP ASVS	(OWASP) Application Security Verification Standard	Open source
RDP	Remote Desktop Protocol	Industry term
SSH	Secure Shell	Industry term

Appendix E. Glossary

Table 6 – Glossary

Term	Definition
Image	A package that contains all files required to run a container.
OFFICIAL	Information classification mark, identified in the Government Security Classification Policy.
Container	A method for packaging and securely running an application within an application virtualisation environment. Also known as an application container or a server application container.
Container runtime	The environment for each container; comprised of binaries coordinating multiple operating systems components that isolate resources and resource usage for running containers.
Container specific OS	A minimalistic host operating system explicitly designed to only run containers.
Gold Build	A detailed build document and associated master image that has been evaluated for security issues, which then forms a template used to deploy replicated instances across the network.
Namespace isolation	A form of isolation that limits which resources a container may interact with.
Orchestrator	A tool that enables DevOps personas or automation working on their behalf to pull images from registries, deploy those images into containers, and manage the running of containers.

Registry	A service that allows developers to easily store images as they are created, tag and catalogue images for identification and version control to aid in the discovery and reuse and find and download images that others have created.
Virtualisation	The simulation of the software and/or hardware upon which other software runs.

Appendix F. Accessibility artefacts

A variety of accessibility guidance is available from the below URL, that includes:

https://www.gov.uk/guidance/guidance-and-tools-for-digital-accessibility

https://www.gov.uk/guidance/accessibility-requirements-for-public-sector-websites-and-apps