Security Standard - Software Development (SS-003)

Chief Security Office

Date: 14/06/2023

Department for Work & Pensions

This Software Development 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 Department 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. Contents

2. Revision History

Version	Author	Description	Date
1.0		First published version	26/06/2017
1.1		Multiple revisions in all sections following feedback from SMEs.	07/10/2017
2.0		Full update in line with current best practices and standards; • Updated Intro, purpose, audience, scope; added reference to CIS v8 security controls • Added NIST CSF references 11.1.4 'must'; vulnerability assessments 11.1.5 Added ref to open source and ESRM supplier assurance 11.1.8 Integrity checking 11.1.11 Privilege escalation must be logged 11.1.13 Sanitise user data 11.1.14 Remove access 11.1.19 Automated testing 11.1.22 Do not use 'referrer' header 11.1.23 Secrets management 11.1.24 Code signing 11.2.2 Removed threat modelling; trusted/untrusted 11.2.3 UTF8 by default, EBCDIC by exception 11.2.7 Do not pass user supplied data for processing directly 11.2.9 Allowlist 11.2.10 Moved redirects from previous entry 11.2.11 Added RFC3986 11.3.1 Performed 11.4.12 NCSC password guidance 11.4.13 Change password reset limits 11.5.2 OAuth 11.5.3 standardised and federated 11.5.6 Added ref to Privileged User Access standard 11.5.14 Authenticate files before uploading 11.6 Sessions must be encrypted 11.6.10 Geo-locations 11.6.11 Secure Cookie Attribute; HttpOnly attribute; added ref to Use of Crypto standard 11.6.12 Separation of duties 11.6.14 Added ref to Use of Crypto standard 11.7.13 Re-apply permissions on re-enabled accounts 11.7.14 Non-human, automated accounts 11.7.15 ACP templates may be used 11.8.1 Examples added	14/06/2023

		11.8.5 FIPS140 validation 11.8.6 Added ref to PKI standard 11.9 STDOUT and STDERR definition 11.9.1 SIEM tooling compatibility – JSON, XML, CSV. 11.9.2 Success and failure logging; must include 11.9.7 Engineering users 11.10 Added ref to Secure Coding Guidelines 11.11.3 Allowlist 11.11.5 Transformation / Conversion 11.12 Added ref to Supply Chain Levels for Software Artifacts framework 11.12.3 All environments	

3. Approval History

Version	Name	Role	Date
1.0		Chief Security Officer	26/06/2017
1.1		Chief Security Officer	07/10/2018
2.0		Chief Security Officer	14/06/2023

This document will be reviewed for continued completeness, relevancy and accuracy within 1 year of being granted "final" status, and at yearly intervals thereafter.

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. K].
- 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, engineers, developers, security teams, project teams, including suppliers engaged in the design, development, implementation and operation of systems, services and applications.

7. Accessibility Requirements

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 Software Development Security Standard defines the minimum technical security measures that **must** be implemented when developing and maintaining software for use within the Authority. These requirements **must** also be considered against code that may not have been developed by the Authority (e.g. taken on from third party suppliers, not open source third party libraries or repositories). This document should be read in conjunction with the DWP Information Security Policy, Acceptable Use Policy and DWP Open Source Policy.

This set of requirements will support the iterative agile development lifecycle and will be matured and refined as projects and development efforts feedback information to mature it.

There is a risk from threat actors using malicious code to exploit vulnerabilities in Authority systems, applications and business practices. This presents a significant risk to the Confidentiality, Integrity and Availability of Authority Systems and the information they process, store and exchange.

All areas of the Authority should assume that, where there is a business requirement to exchange data with business partners or import content from other information sources including open source, information technology systems and information assets are highly likely to be affected by attacks via malicious code of a direct and also indirect nature.

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 Authority or our third party providers, such as the CIS Critical Security Controls v8 controls set. [see Appendix C 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 software development are implemented consistently across the Authority and by third party providers where applicable.
- mitigate risks from common threats and vulnerabilities associated with software development, 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, and provide an objective, measurable statement of the Authority's existing security posture in a number of important areas. The outcomes are based on the official NIST sub-categories where possible to ensure close alignment with the NIST Cyber Security Framework (CSF), and are enabled by the implementation of controls from the CIS Critical Security Controls v8 controls set. [see Appendix C External References]. Those relevant to the subject of each standard can be found in Appendix A of every technical security standard.

9. Purpose

The purpose of this standard is to ensure that Authority systems and services 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 software development within the Authority and supplier base (contracted third party providers), for the purposes of delivering applications and services that handle Authority data.

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 General Coding Practices

Reference	Minimum Technical Security Measures	NIST ID
11.1.1	The Software Development Security Standard (i.e. this document) **must** be in place, made available to the development team and shared in accessible area.	ID.GV-1 PR.AT-5
11.1.2	Rules for the development of software and systems **must** be established, documented and applied to developments within the organisation. This will be made available to projects and other parties with a need to know.	ID.GV-1
11.1.3	Projects delivered through an agile project delivery **must** have Security Stories. To ensure that security considerations are addressed throughout the development lifecycle, Security stories and relevant acceptance criteria **must** be added to development teams' backlogs upon project initiation. These may be identified during Discovery and Alpha, but **must** be implemented by the time the development moves into Beta.	PR.IP-7
11.1.4	Tested and approved managed code **must** be used rather than creating new unmanaged code for common tasks. Code analysis and vulnerability scanning tools are available for this purpose. Vulnerability assessments **must** be completed on a regular basis in line with the Technical Vulnerability Management Policy [Ref. L].	PR.IP-2 PR.IP-12 DE.CM-4
11.1.5	Supply Chain Agreements **must** be in place that third parties have been assessed via the ESRM Supplier Security Assurance process where applicable. Please note this may not be possible for open source products, but a commercial wrapper **must** be in place if available.	ID.SC-2 ID.SC-3
11.1.6	Outsourced development (including suppliers and external third parties) efforts **must** follow the Authority Software Development security standard (i.e. this document).	ID.GV-1 PR.AT-5

11.1.7	Utilise task specific built-in APIs to conduct operating system tasks. Do not allow the application to issue commands directly to the Operating System, especially through the use of application initiated command shells.	PR.IP-2
11.1.8	The integrity of interpreted code, libraries, executable files, and configuration files **must** be checked, using automated tooling where available.	PR.DS-6 DE.CM-4
11.1.9	Ensure that applications are written to be thread-safe.	PR.DS-6
11.1.10	Where appropriate, explicitly initialize all your variables and other data stores, either during declaration or just before the first usage.	PR.DS-6
11.1.11	In cases where the applications or functions **must** run with elevated privileges, raise privileges as late as possible, and drop them as soon as possible. These **must** be logged.	PR.AC-4
11.1.12	Avoid calculation errors by understanding your programming language's underlying representation and how it interacts with numeric calculation. Pay close attention to byte size discrepancies, precision, signed/unsigned distinctions, truncation, conversion and casting between types, "not-a-number" calculations, and how your language handles numbers that are too large or too small for its underlying representation.	PR.DS-6
11.1.13	All user supplied data **must** be sanitised before use.	PR.DS-6 DE.CM-4
11.1.14	Restrictions **must** be in place, and access removed, preventing unauthorised users from generating new code or altering existing code in Authority code repositories. Staff leaving the Authority (including contractors) must have their permissions removed in a timely manner.	PR.AC-1 PR.DS-1
11.1.15	Review all secondary applications, third party code and libraries to determine business necessity and validate safe functionality, as these can introduce new vulnerabilities and potential back doors. This activity should also be performed periodically after the initial review.	ID.RA-1 ID.RA-4 DE.CM-4
11.1.16	Safe updating **must** be implemented, if the production application will utilise automatic updates, using cryptography with verification of signatures.	PR.IP-3
11.1.17	Systems updates **must** use encrypted channels to transfer the code from the host server, where appropriate (mandatory for write operations).	PR.DS-2
11.1.18	Threat Modelling **must** be carried out as part of the design phase to include all application functionality and API's.	ID.RA-3
11.1.19	**There **must** be a planned Security Testing prior to go live – this **must** include automated testing at minimum, plus additional testing where necessary.	RS.AN-5

11.1.20	Enforce application logic flows to comply with business rules.	ID.AM-3
11.1.21	Limit the number of transactions a single user or device can perform in a given period of time. The transactions/time **must** be above the actual business requirement, but low enough to deter automated attacks.	PR.DS-4 PR.PT-4 PR.PT-5
11.1.22	The "referrer header" **must** not be used as an authorisation check, as it can be spoofed.	PR.AC-7
11.1.23	Secrets such as keys, certificates, or credentials **must** be managed in accordance with SS-007 Use of Cryptography Security Standard [Ref. A] as applicable.	PR.DS-1
11.1.24	All Authority code **must** be digitally signed, prior to committing to the code repository, via a centrally approved tool. Only an Authority approved enterprise code signing tool can be used for this purpose.	PR.AC-1

* For the full security control set on cloud related issues please refer to SS-023 Cloud Computing Security Standard [Ref. C].

** For full details of the required security testing please refer to SS-027 Security Testing Standard [Ref. D].

11.2 Input Validation

Reference	Minimum Technical Security Measures	NIST ID
11.2.1	All data inputs **must** be validated on a trusted system (e.g., the server, not the client).	PR.DS-6
11.2.2	Validate data from all data sources, whether trusted or untrusted.	ID.AM-5
11.2.3	Character sets **must** use UTF-8 by default, for all sources of input by default. Other character sets (such as EBCDIC) may be used by exception only, and **must** be recorded.	PR.DS-1
11.2.4	Data **must** be encoded to a common character set before being validated.	PR.DS-1
11.2.5	All validation failures **must** result in input rejection.	PR.DS-6
11.2.6	Where UTF-8 extended character sets is used, validate after UTF-8 decoding is completed.	PR.DS-6

11.2.7	Do not pass user supplied data directly to any dynamic include function i.e. prevent malicious code from affecting the logic of the function. Validation for all client provided data before processing, including all parameters, URLs and HTTP header content (e.g. Cookie names and values) **must** be in place. This is to include automated post backs from embedded code.	PR.DS-6
11.2.8	Header values in both requests and responses containing only ASCII characters **must** be verified.	PR.DS-6
11.2.9	All input MUST be validated against an "allow list" of allowed characters.	PR.DS-6
11.2.10	Validation **must** be in place for; - expected data types - data range - data length - redirects	PR.DS-6
11.2.11	If any potentially hazardous characters must be allowed as input, you **must** ensure that you implement additional controls like URI encoding, secure task specific APIs and accounting for the utilisation of that data throughout the application. If your standard validation routine cannot address the following inputs for example, then they **must** be checked where appropriate - Check for null bytes (%00) - Check for new line characters (%0d, %0a, \r, \n) - Check for “dot-dot-slash" (../ or ..\) path alterations characters. In cases where UTF-8 extended character set encoding is supported, address alternate representation like: %c0%ae%c0%ae/ (Utilise canonicalization to address double encoding or other forms of obfuscation attacks) Please refer to RFC3986 [see External References] for further information.	PR.DS-1 PR.DS-6

11.3 Output Encoding

Reference	Minimum Technical Security Measures	NIST ID
11.3.1	All encoding **must** be performed on a trusted system (e.g., the server, not the client).	PR.DS-1
11.3.2	A standard, tested routine for each type of outbound encoding **must** be used by the delivery team	PR.DS-1
11.3.3	Contextual output encoding: encode output data returned to the client that originated outside the application's trust boundary. HTML entity encoding is one example, but there are other use cases like SQL queries, XML, LDAP.	PR.DS-2

11.4 Password Management

(Please refer to NCSC guidance on password management, and also SS-001 pt.1 Access and Authentication Security Standard [Ref. E]).

Reference	Minimum Technical Security Measures	NIST ID
11.4.1	Password hashing **must** be implemented on a trusted system (e.g., the server, not the client) and be configured in the correct mode.	PR.AC-6 PR.DS-1
11.4.2	Passwords **must** be sent over an encrypted connection or as encrypted data, such as in an encrypted email, in line with SS-007 Use of Cryptography Security Standard [Ref. A].	PR.DS-2
11.4.3	Enforcement of password complexity requirements established by policy or regulation **must** be adhered to, in line with SS-001 pt.1 Access and Authentication Security Standard [Ref. E]. Authentication credentials **must** be sufficient to withstand attacks that are typical of the threats in the deployed environment. (e.g., requiring the use of alphabetic as well as numeric and/or special characters).	PR.AC-1 PR.AC-4 PR.AC-6 PR.AC-7
11.4.4	Enforce password length requirements established by policy or regulation in line with SS-001 pt.1 Access and Authentication Security Standard [Ref. E]. Eight characters is commonly used, but 16 is better or consider the use of multi-word pass phrases.	PR.AC-1 PR.AC-4 PR.AC-6 PR.AC-7
11.4.5	Password entry **must** be obscured on the user's screen. (e.g., on web forms use the input type "password") .	PR.AC-1 PR.AC-6 PR.AC-7
11.4.6	Enforce account disabling after 3 invalid login attempts. The account **must** be disabled for an escalating time interval, minimum time of 15 minutes, sufficient to discourage brute force guessing of credentials, but not so long as to allow for a denial-of-service attack to be performed.	PR.AC-1 PR.AC-4

11.4.7	Password reset and changing operations require the same level of controls as account creation and authentication.	PR.AC-1 PR.AC-4
11.4.8	If using email based resets, only send email to a pre-registered address with a temporary link/password.	PR.AC-1 PR.AC-4
11.4.9	Temporary passwords and links **must** have a short expiration time e.g. 10 minutes.	PR.AC-1 PR.AC-4
11.4.10	Enforce the changing of temporary passwords on the next use.	PR.AC-1 PR.AC-4
11.4.11	Notify users when a password reset occurs.	PR.AC-1 PR.AC-4
11.4.12	Password changes and prevention of password re-use **must** follow NCSC guidance on password management.	PR.AC-1 PR.AC-4
11.4.13	Passwords may only be reset up to 3 times a day before the account is locked, to prevent attacks on password re-use.	PR.AC-1 PR.AC-4
11.4.14	Implement monitoring to identify attacks against multiple user accounts, utilizing the same password. This attack pattern is used to bypass standard lockouts, when user IDs can be harvested or guessed.	DE.CM-1

11.5 Authentication

(SS-007 Use of Cryptography Security Standard can be referred to for specifics on encryption [Ref. A])

Reference	Minimum Technical Security Measures	NIST ID
11.5.1	All authentication controls **must** be enforced on a trusted system (e.g., the server, not the client).	PR.AC-1 PR.AC-7
11.5.2	Applications **must** establish and utilise standard, tested, authentication protocols or mechanisms e.g. API gateway, OAuth.	PR.AC-7
11.5.3	A standardised and federated approach for all authentication controls, including libraries that call external authentication services **must** be used where possible and available. Authentication logic **must** be segregated from the resource being requested and use redirection to and from the centralised authentication control.	PR.AC-1 PR.AC-5 PR.AC-7
11.5.4	All authentication controls **must** fail securely.	PR.AC-5 PR.AC-7
11.5.6	All administrative and account management functions **must** be in line with SS-001 pt.2 Privileged User Access Security Standard [Ref. F].	PR.AC-4

11.5.7	If the application manages a credential store, it **must** be managed in accordance with the SS-007 Use of Cryptography Security Standard [Ref. A] and SS-001 pt.1 Access and Authentication Security Standard [Ref. E].	PR.DS-1
11.5.8	Authentication data **must** be validated only on completion of all data input, especially for sequential authentication implementations.	PR.AC-1
11.5.9	Authentication failure responses **must** not indicate which part of the authentication data was incorrect. (For example, instead of "Invalid username" or "Invalid password", just use "Invalid username and/or password" for both).	PR.AC-1 PR.AC-5
11.5.10	Authentication **must** be used for connections to external systems that involve sensitive information or functions.	PR.AC-3
11.5.11	Authentication credentials for accessing services external to application/s **must** be encrypted and stored in a protected location on a trusted system (e.g., the server, not the client). The source code is NOT a secure location	PR.DS-1
11.5.12	HTTP POST (or PUT) **must** be used for requests to transmit authentication credentials.	PR.DS-2
11.5.13	Where appropriate, the last use (successful or unsuccessful) of a user account **must** be reported to the user at their next successful login, for both internal and external facing web applications.	PR.AC-6
11.5.14	Require authentication before allowing a file to be uploaded.	PR.AC-1 PR.AC-7

11.6 Session Management

(Sessions must be encrypted in line with SS-007 Use of Cryptography Security Standard [Ref. A]).

Reference	Minimum Technical Security Measures	NIST ID
11.6.1	Use the server or framework's session management controls. The application **must** only recognize these session identifiers as valid.	PR.DS-5
11.6.2	Session identifier creation **must** always be done on a trusted system (e.g., the server, not the client), and configured in the correct mode.	PR.DS-5
11.6.3	Set the domain and path for cookies containing authenticated session identifiers to an appropriately restricted value for the site.	PR.DS-5
11.6.4	Logout functionality **must** fully terminate the associated session or connection	PR.DS-5

11.6.5	Logout functionality **must** be available from all pages protected by authorization.	PR.DS-5
11.6.6	Establish a session inactivity timeout that is as short as possible, based on balancing risk and business functional requirements. It **must** be no more than two hours for external services, and 6 hours for internal services.	PR.DS-5
11.6.7	Disallow persistent logins and enforce periodic session terminations, even when the session is active. Especially for applications supporting rich network connections or connecting to critical systems. Termination times **must** support business requirements and the user **must** receive sufficient notification to mitigate negative impacts.	PR.DS-5
11.6.8	If a session was established before login, close that session and establish a new session after a successful login.	PR.DS-5
11.6.9	Generate a new session identifier on any re-authentication.	PR.DS-5
11.6.10	Do not allow concurrent logins with different geo-locations. If a session is still active, and the user is trying to login, logout the existing userid and allow the user to login.	PR.DS-5
11.6.11	Do not expose session identifiers in URLs, error messages or logs. Session identifiers **must** only be located in the HTTP cookie header. Cookies **must** set a secure attribute (see External References https://owasp.org/www.community/controls/SecureCookieAttribute) and **must** be transmitted over a secure connection in line with SS-007 Use of Cryptography Security Standard [Ref. A]. Set cookies with the HttpOnly attribute, unless you specifically require client-side scripts within your application to read or set a cookie's value (see External References https://owasp.org/www-community/HttpOnly).	PR.DS-2 PR.DS-5
11.6.12	Protect server side session data from unauthorized access, by other users of the server, by implementing appropriate access controls on the server, maintaining separation of duties.	PR.DS-1
11.6.13	Generate a new session identifier and deactivate the old one periodically. (This can mitigate certain session hijacking scenarios where the original identifier was compromised).	PR.DS-5
11.6.14	Generate and transmit new session identifiers in line with SS-007 Use of Cryptography Security Standard [Ref. A].	PR.DS-2

11.6.15	Supplement standard session management for sensitive server-side operations, like account management, by utilizing per-session strong random tokens or parameters. This method can be used to prevent Cross Site Request Forgery attacks.	PR.DS-2
11.6.16	Supplement standard session management for highly sensitive or critical operations by utilizing per-request, as opposed to per-session, strong random tokens or parameters.	PR.DS-5

11.7 Access Control

(SS-001 pt.1 Access and Authentication Security Standard [Ref. E] can be referred to for specifics)

Reference	Minimum Technical Security Measures	NIST ID
11.7.1	Use only trusted system objects, e.g. server side session objects, for making access authorization decisions.	PR.AC-1 PR.AC-4
11.7.2	Use a single site-wide component to check access authorization. This includes libraries that call external authorization services	PR.AC-4
11.7.3	Access controls **must** fail securely	PR.AC-4
11.7.4	Deny all access if the application cannot access its security configuration information	PR.AC-3 PR.AC-5
11.7.5	Enforce authorization controls on every request, including those made by server side scripts, "includes" and requests from rich client-side technologies.	PR.AC-1 PR.AC-4 PR.AC-6
11.7.6	Segregate privileged logic from other application code to ensure that privileges granted to one cannot be used for the other.	PR.AC-4
11.7.7	Restrict access to files or other resources, including those outside the application's direct control, to only authorized users	PR.AC-4
11.7.8	Restrict to authorised users only; • access to protected URLs • access to protected functions • direct object references • access to services • access to application data • access to user and data attributes and policy information used by access controls • access security-relevant configuration information	PR.AC-2 PR.AC-4
11.7.9	If state data **must** be stored on the client, use encryption and integrity checking on the server side to catch state tampering.	PR.DS-1

11.7.10	Limit the number of transactions a single user or device can perform in a given period of time. The transactions/time **must** be above the actual business requirement, but low enough to deter automated attacks	PR.DS-5 PR.DS-6
11.7.11	If user permissions are changed, force a log out and re-authentication.	PR.AC-1 PR.AC-4
11.7.12	Implement account auditing and enforce the disabling of unused accounts in line with SS-001 pt.1 Access and Authentication Security Standard [Ref. E].	PR.AC-1 PR.AC-2
11.7.13	The application **must** support disabling of accounts and terminating sessions when authorization ceases (e.g., Changes to role, employment status, business process, etc.). If a disabled account is re-enabled, all permissions **must** be removed and re-applied in line with the role before re-using.	PR.AC-1 PR.AC-2 PR.AC-4
11.7.14	Service accounts (i.e. non-human, for automated processes) or accounts supporting connections to or from external systems **must** have the least privilege possible	PR.AC-4
11.7.15	Create an Access Control Policy to document an application's business rules, data types and access authorization criteria and/or processes so that access can be properly provisioned and controlled. This includes identifying access requirements for both the data and system resources. Templates may be used for this purpose.	ID.GV-1

11.8 Cryptographic Practices

(SS-007 Use of Cryptography Security Standard [Ref. A] can be referred to for specifics on encryption)

Reference	Minimum Technical Security Measures	NIST ID
11.8.1	All cryptographic functions used to protect secrets (e.g. API tokens, KMS keys, private certificates, key material etc.) from the application user **must** be implemented on a trusted system (e.g., the server, not the client).	PR.DS-1
11.8.2	Protect master secrets from unauthorized access.	PR.DS-1
11.8.3	Cryptographic modules **must** fail securely.	PR.DS-1
11.8.4	All random numbers, random file names, random GUIDs, and random strings **must** be generated using the cryptographic module's approved random number generator when these random values are intended to be un-guessable.	PR.DS-1

11.8.5	Cryptographic modules used by the application must have an active validation to FIPS 140. (See NIST Cryptographic Module Validation Program http://csrc.nist.gov/groups/STM/cmvp/validation.html (See External References]).	PR.DS-1
11.8.6	Cryptographic keys **must** be managed as per SS-007 Use of Cryptography Security Standard [Ref. A] and SS-002 PKI & Key Management Security Standard [Ref. G]. Any exceptions **must** be notified.	PR.DS-1

11.9 Error Handling and Logging

For clarity, standard output (stdout) is the stream to which a program writes its output data; standard error (stderr) is another output stream typically used by programs to output error messages or diagnostic data.
(For relevant logging details pertaining to the wider Authority requirements for logging please refer to SS-012 Protective Monitoring Security Standard [Ref. H]).

Reference	Minimum Technical Security Measures	NIST ID
11.9.1	Logging Output Integration **must** be compatible with existing Departmental aggregation and SIEM tooling e.g. JSON, XML, or CSV formats.	ID.DE-3 DE.DP-2
11.9.2	All logging controls **must** be implemented on a trusted system (e.g., the server, not the client), and **must** support both success and failure of specified security events.	ID.DE-3 DE.DP-2
11.9.3	Ensure logs contain important log event data, which **must** include input and output validation failures, authentication successes and failures, authorisation (access control) failures, session management failures, application errors and system events, use of higher-risk functionality.	ID.DE-3 DE.DP-2
11.9.4	Where possible, ensure log entries that include un-trusted data will not execute as code in the intended log viewing interface or software. Sensitive data such as PII **must** not be written to log files.	DE.AE-4
11.9.5	Restrict access to logs to only authorised individuals.	PR.AC-4
11.9.6	Do not disclose sensitive information in error responses, including system details, session identifiers or account information.	PR.DS-5
11.9.7	Use error handlers that do not display debugging or stack trace information to the end user – engineering users should use appropriate outputs (std err and std out and also message output dialogues).	PR.DS-5
11.9.8	Implement generic error messages and use custom error pages .	PR.DS-5

11.10 Data Protection

For more detailed information on secure coding practices, please refer to the Engineering Secure Coding Guidelines (Ref. J].

Reference	Minimum Technical Security Measures	NIST ID
11.10.1	Protect security-specific server-side source-code from being downloaded by a user, for example; • security enforcing functions, • internal security or anti-fraud rules, • the existence, configuration and/or status of sensitive components within our estate such as configuration files, IP addresses and Domain names, • anything that identifies vulnerabilities in existing versions of Open Source code, owned by either Authority or external code, • any legal or commercial issues/or any system/service vulnerabilities or remediation plans, • code which incorporates components of code from a third party (whether open source or proprietary), where the relevant license terms preclude or restrict such sharing, • any code, standing data, sensitive data, staff records or other components that could help a threat actor to access customer records.	PR.DS-5

11.11 File Management

Reference	Minimum Technical Security Measures	NIST ID
11.11.1	Do not pass user supplied data directly to any dynamic include function i.e. prevent malicious code from affecting the logic of the function.	PR.DS-5
11.11.2	Limit the type of files that can be uploaded to only those types that are needed for business purposes. Where possible, validate that uploaded files are of the expected type by checking file headers. Checking for file type by extension alone is not sufficient.	PR.DS-5
11.11.3	When referencing existing files, use an allowlist of allowed file names and types. Validate the value of the parameter being passed and if it does not match one of the expected values, either reject it or use a hard coded default file value for the content instead.	PR.DS-5

11.11.4	Scan user uploaded files for viruses and malware using up to date dictionaries, rejecting those that may be suspect, in line with SS-015 Malware Protection Security Standard [Ref. I].	DE.CM-4
11.11.5	Data from external or less trusted sources may contain malicious content. If such data cannot be validated, it **must** be transformed or converted to another format to reduce this risk before being passed on to its destination. [See external references NCSC Secure Design Principles 2.1]	ID.AM-5

11.12 Application Lifecycle

Further information on application lifecycles can be found in the Supply Chain Levels for Software Artifacts framework (See External References).

Reference	Minimum Technical Security Measures	NIST ID
11.12.1	Code promotion **must** be controlled and managed at identified significant points within the development lifecycle	PR.IP-2 PR.IP-3
11.12.2	Where appropriate, when decommissioning hosts and environments that were used in the development effort, they **must** be securely wiped or deleted after use.	PR.IP-2 PR.DS-5
11.12.3	All environments **must** be logically segregated from each other.	PR.DS-7
11.12.4	Consideration **must** be given to any integration needs early in the project. New software products and systems **must** integrate with applicable existing solutions and capabilities where appropriate.	PR.IP-2

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.

Table 1 – List of Security Outcomes Mapping

NIST Ref	Security Outcome (sub-category)	Related Security measure
ID.AM-3	Organizational communication and data flows are mapped	11.1.20
ID.AM-5	Resources (e.g., hardware, devices, data, time, personnel, and software) are prioritized based on their classification, criticality, and business value	11.2.2, 11.11.5
ID.GV-1	Organizational cybersecurity policy is established and communicated	11.1.1, 11.1.2, 11.1.6, 11.7.15
ID.RA-1	Asset vulnerabilities are identified and documented	11.1.15
ID.RA-3	Threats, both internal and external, are identified and documented	11.1.18
ID.RA-4	Potential business impacts and likelihoods are identified	11.1.15
ID.SC-2	Suppliers and third party partners of information systems, components, and services are identified, prioritized, and assessed using a cyber supply chain risk assessment process	11.1.5
ID.SC-3	Contracts with suppliers and third-party partners are used to implement appropriate measures designed to meet the objectives of an organization's cybersecurity program and Cyber Supply Chain Risk Management Plan.	11.1.5

PR.AC-1	Identities and credentials are issued, managed, verified, revoked, and audited for authorized devices, users and processes	11.1.14, 11.1.24, 11.4.3, 11.4.4, 11.4.5, 11.4.6, 11.4.7, 11.4.8, 11.4.9, 11.4.10, 11.4.11, 11.4.12, 11.4.13, 11.5.1, 11.5.3, 11.5.8, 11.5.9, 11.5.14, 11.7.1, 11.7.5, 11.7.11. 11.7.12, 11.7.13
PR.AC-2	Physical access to assets is managed and protected	11.7.8, 11.7.12, 11.7.13
PR.AC-3	Remote access is managed	11.5.10, 11.7.4
PR.AC-4	Access permissions and authorizations are managed, incorporating the principles of least privilege and separation of duties	11.1.11, 11.4.3, 11.4.4, 11.4.6, 11.4.7, 11.4.8, 11.4.9, 11.4.10, 11.4.11, 11.4.12, 11.4.13, 11.5.6, 11.7.1, 11.7.2, 11.7.3, 11.7.5, 11.7.6, 11.7.7, 11.7.8, 11.7.11, 11.7.13, 11.7.14, 11.9.4
PR.AC-5	Network integrity is protected (e.g., network segregation, network segmentation)	11.5.3, 11.5.4, 11.5.9, 11.7.4
PR.AC-6	Identities are proofed and bound to credentials and asserted in interactions	11.4.1, 11.4.3, 11.4.4, 11.4.5, 11.5.13, 11.7.5
PR.AC-7	Users, devices, and other assets are authenticated (e.g., single-factor, multi-factor) commensurate with the risk of the transaction (e.g., individuals' security and privacy risks and other organizational risks)	11.1.22, 11.4.3, 11.4.4, 11.4.5, 11.5.1, 11.5.2, 11.5.3, 11.5.4, 11.5.14
PR.AT-5	Physical and cybersecurity personnel understand their roles and responsibilities	11.1.1, 11.1.6

PR.DS-1	Data-at-rest is protected	11.1.14, 11.1.23, 11.2.3, 11.2.4, 11.2.11, 11.3.1, 11.3.2, 11.4.1, 11.5.7, 11.5.11, 11.6.12, 11.7.9
PR.DS-2	Data-in-transit is protected	11.1.17, 11.3.3, 11.4.2, 11.5.12, 11.6.11, 11.6.14, 11.6.15, 11.8.1, 11.8.2, 11.8.3, 11.8.4, 11.8.5, 11.8.6
PR.DS-4	Adequate capacity to ensure availability is maintained	11.1.21
PR.DS-5	Protections against data leaks are implemented	11.6.1, 11.6.2, 11.6.3, 11.6.4, 11.6.5, 11.6.6, 11.6.7, 11.6.8, 11.6.9, 11.6.10, 11.6.11, 11.6.13, 11.6.16, 11.7.10, 11.9.6, 11.9.7, 11.9.8, 11.10.1, 11.11.1, 11.11.2, 11.11.3, 11.12.2
PR.DS-6	Integrity checking mechanisms are used to verify software, firmware, and information integrity	11.1.8, 11.1.9, 11.1.10, 11.1.12, 11.1.13, 11.2.1, 11.2.5, 11.2.6, 11.2.7, 11.2.8, 11.2.9, 11.2.10, 11.2.11, 11.7.10
PR.DS-7	The development and testing environment(s) are separate from the production environment	11.12.3
PR.IP-2	A System Development Life Cycle to manage systems is implemented	11.1.4, 11.1.7, 11.12.1, 11.12.2, 11.12.4

PR.IP-3	Configuration change control processes are in place	11.1.16, 11.12.2
PR.IP-12	A vulnerability management plan is developed and implemented	11.1.4
PR.IP-7	Protection processes are improved	11.1.3
PR.PT-4	Communications and control networks are protected	11.1.21
PR.PT-5	Mechanisms (e.g., failsafe, load balancing, hot swap) are implemented to achieve resilience requirements in normal and adverse situations	11.1.21
DE.AE-3	Event data are collected and correlated from multiple sources and sensors	11.9.1, 11.9.2, 11.9.3
DE.AE-4	Impact of events is determined	11.9.4
DE.CM-1	The network is monitored to detect potential cybersecurity events	11.4.14
DE.CM-4	Malicious code is detected	11.1.4, 11.1.8, 11.1.13, 11.1.15, 11.11.4
DE.DP-2	Detection activities comply with all applicable requirements	11.9.1, 11.9.2, 11.9.3
RS.AN-5	Processes are established to receive, analyze and respond to vulnerabilities disclosed to the organization from internal and external sources (e.g. internal testing, security bulletins, or security researchers)	11.1.19

Appendix B Internal References

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

Table 2 – Internal References

Ref	Document	Publicly Available*
A	SS-007 Use of Cryptography Security Standard	Yes
B	Information Management Policy	Yes
C	SS-023 Cloud Computing Security Standard.	Yes
D	SS-027 Security Testing Security Standard	No
E	SS-001 pt.1 Access and Authentication Security Standard	Yes
F	SS-001 pt.2 Privileged User Access Security Standard	Yes
G	SS-002 PKI & Key Management Security Standard	Yes
H	SS-012 Protective Monitoring Security Standard	Yes
I	SS-015 Malware Protection Security Standard	Yes
J	Engineering Secure Coding Guidelines	No
K	Security Assurance Strategy	No
L	Technical Vulnerability Management Policy	Yes

Requests to access non-publicly available documents **should be made to an Authority Contracts/Supplier Manager.*

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 3 – External References

External Documents List
CIS Critical Security Controls v8 controls set
UK General Data Protection Regulation (UK GDPR)
SEI CERT Secure Coding Standards - www.securecoding.cert.org/
NIST Cryptographic Module Validation Program http://csrc.nist.gov/groups/STM/cmvp/validation.html
RFC 3986: Uniform Resource Identifier (URI): Generic Syntax
https://owasp.org/www.community/controls/SecureCookieAttribute
https://owasp.org/www-community/HttpOnly
Supply Chain Levels for Software Artifacts framework - https://slsa.dev/spec/v0.1/levels
DWP Open Source Policy
NCSC Secure Design Principles - https://www.ncsc.gov.uk/collection/cyber-security-design-principles/making-compromise-difficult

Appendix D Abbreviations

Table 4 – Abbreviations

Abbreviation	Definition
AES	Advanced Encryption Standard – defined in FIPS 197 in NIST. Different modes of operation are covered in different documents.
CIS	Centre for Internet Security
DA	Design Authority (DA)
DAST	Dynamic Application Security Testing
DDoS	Distributed Denial of Service
DWP	Department of Work and Pensions.
FIPS 140-2	Federal Information Processing Standard 140-2 is a U.S. government computer security standard used to approve cryptographic modules
GCSP	Government Security Classification Policy
IDE	Integrated Development Environment
ITHC	IT Security Health Check
LDAP	Lightweight Directory Access Protocol
NIST	National Institute of Standards and Technology
NIST – CSF	National Institute of Standards and Technology – Cyber Security Framework
OWASP	Open Web Application Security Project
SAST	Static Application Security Testing
SQL	Structured Query Language
TLS	Transport Layer Security
XFS	'X' File System
XML	Extensible Markup Language

Appendix E Definition of Terms

Table 5 – Glossary

Term	Definition
Security Wrapper	A concept that contains security controls and architectures that are complimentary and which can include some or all of the following; Identity and Access management, security event and incident management, anti-malware, cryptographic key management, trust management, resilience management, threat and vulnerability management, fraud management (anti-fraud)
Cryptographic Key Material	Any parameter passed to an encryption cipher which influences the output of the algorithm (with the exception of the message itself).
Security Story	Similar to a user story, a definition of expectation with respect to security; developed from an initial security review, examining functionality, policies, legal, regulatory and contractual requirements, etc. These may then be better defined with the development team and prioritised for handling. When the development cycle/sprint begins, the development team can pick up the security stories as well as the functional stories to begin the process of agile security development.

Appendix F Accessibility artefacts

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

https://accessibility-manual.dwp.gov.uk/

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

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