AIRPORTS COMPANY
SOUTH AFRICA

Web Application Security Procedure
Public

Information Security

Web application Security Standards

AREA OF APPLICABILITY

Information Security

DIVISION

Information Technology

Next Revision Date

January 2027

Control Disclosure:

Public

Effective Date:

AIRPORTS COMPANY
SOUTH AFRICA

Web Application Security Procedure
Public

Table of Contents

Table of Contents 2
1. Scope 4
2. Objective 4
3 Definitions and Abbreviations 4
3.1 Definitions 4
3.2 Abbreviations 4
4. Procedure General 4
4.1 Classification of Controls 4
4.2 Architecture and Design Requirements 5
4.3. Authentication verification requirements 5
4.4 Session Management Requirements 8
4.5 Access control verification requirements 9
4.6 Malicious Input Handling Verification Requirements 10
4.7 Cryptography verification requirements 12
4.8 Error handling and logging verification requirements 13
4.9 Data Protection Verification Requirements 14
4.10 Communications security verification requirements 15
4.11 HTTP security configuration verification requirements 16
4.11 Files and resources verification requirements 17
4.12 Mobile application security verification requirements 18
4.13 Accountability and Responsibility 20
5. Process for Monitoring 20
6. Accountabilities and Responsibilities 21
6.1 Accountabilities 21
7. Reporting of Non-Conformance 22
8. Related Policy Documents 22
9. Related Legislation and Standard 22
10. Change Control and Verification 22
11. Records 22
12. Revision History 22

13. Endorsement (See Master in Corporate Policy Document Store) 23

AIRPORTS COMPANY
SOUTH AFRICA

Web Application Security Procedure
Public

1. Scope

This standard applies to all web applications used to conduct Airport Company South Africa's business. This extends to:

- All off-the-shelf, customised, and bespoke web applications including content
- management systems
- Airport Company South Africa web-based applications hosted by external providers
- All internal and public facing web applications
- Airports Company South Africa web applications developed to be accessed from mobile
- devices including tablets and smartphones.

2. Objective

These standard details the controls that shall be deployed on Airports Company South Africa's web applications to ensure their safe and continuing operation, and further outlines why these controls should be implemented, and lastly defines the means to measure compliance against the standard to ensure the effective and efficient functioning of web applications.

3 Definitions and Abbreviations

3.1 Definitions

ACSA
In the context of this procedure, the acronym ACSA refers to Airport Company South Africa SOC Limited

Company/Business/Organisation/Group
Airports Company South Africa SOC Limited

3.2 Abbreviations

Abbreviation	Description
ACSA	Airport Company South Africa

4. Procedure General

4.1 Classification of Controls

This web application security standard defines 3 levels of controls:

- **All Applications controls** are compulsory for all web applications on the network. These controls are opportunistic controls. They protect against application security vulnerabilities that are easy to discover or exploit.
- **Medium Criticality Applications controls** are compulsory for applications that contain sensitive data, which requires protection. This data includes internal information or information about employees that may be leveraged in social engineering, as well as nonessential, but important intellectual property and proprietary application sensitive data. These controls are standard controls that protect against common risks associated with modern day web applications.
- **High Criticality Applications controls** are compulsory for the most critical applications that perform high value transactions, contain sensitive data, or any application that requires the highest level of trust. These include applications that process or store valuable intellectual property, trade secrets, or any data that is critical to the survival or success of the organization, or it's competitive advantage. These controls are advanced controls and are reserved for the most critical web applications that require the highest level of security verification and assurance.

4.2 Architecture and Design Requirements

All verified web applications must satisfy the following high-level design requirements:

- Only those components that are needed by the application are identified and utilized.
- The application architecture has been defined and the application adheres to this architecture.

#	Key Performance Indicator	Level	KPI Type	Benchmark
1.1	Verify that all application components are identified. and are confirmed to be needed.	ALL	Boolean	True
1.2	Verify that a high-level architecture for the application has been defined.	ALL	Boolean	True
1.3	Verify that there is no sensitive business logic, secret keys, or other proprietary information in client side code.	Medium	Boolean	True

4.3. Authentication verification requirements

Authentication is the act of establishing, or confirming, something (or someone) as authentic, that is, that claims made by or about the thing are true. All verified applications must satisfy the following high-level requirements:

- Verifies the digital identity of the sender of a communication.

- Ensures that only those who are authorized can authenticate and their credentials are transported in a secure manner.

#	Key Performance Indicator	Level	KPI Type	Benchmark
2.1	Verify all pages and resources by default require authentication except those specifically intended to be public.	ALL	Boolean	True
2.2	Verify that forms containing credentials are not filled in by the application. Pre-filling by the application implies that credentials are stored in plaintext or a reversible format, which shall be explicitly prohibited.	ALL	Boolean	True
2.3	Verify all authentication controls are enforced on the server side.	ALL	Boolean	True
2.4	Verify all authentication controls fail securely to ensure attackers cannot log in.	ALL	Boolean	True
2.5	Verify password entry fields allow, or encourage, the use of secure and complex passwords such as passphrases, and do not prevent long passphrases or highly complex passwords from being entered.	ALL	Boolean	True
2.6	Verify all account identity authentication functions (such as update profile, forgot password, disabled	ALL	Boolean	True
#	Key Performance Indicator	Level	KPI Type	Benchmark
	) lost token, help desk or IVR) that might regain access to the account are as resistant to attack as the primary authentication mechanism.	ALL	Boolean	True
2.7	Verify that the changing password functionality includes the old password, the new password, and a password confirmation.	ALL	Boolean	True
2.8	Verify that all authentication decisions can be logged, without storing sensitive session identifiers or passwords.	Medium	Boolean	True
2.9	Verify that account passwords are hashed with a one-way hash and can adequately	Medium	Boolean	True

	defeat brute force and password hash recovery attacks			
2.10	Verify that credentials are transported using a suitably encrypted channel and that all pages/functions that require a user to enter credentials do so using an encrypted link.	ALL	Boolean	True
2.11	Verify that the forgotten password function and other recovery paths do not reveal the current password and that the new password is not sent in clear text to the user.	ALL	Boolean	True
2.12	Verify that information enumeration is not possible via login, password reset, or forgot account functionality.	ALL	Boolean	True
2.13	Verify that there are no default passwords in use for the application framework or any components used by the application (such as "admin/password").	ALL	Boolean	True
2.14	Verify that anti-automation is in place to prevent breached credential testing, brute forcing, and account lockout attacks.	ALL	Boolean	True
2.15	Verify that the system can be configured to disallow the use of a configurable number of previous passwords.	Medium	Boolean	True
2.16	Verify that measures are in place to block the use of commonly chosen passwords and weak passphrases.	ALL	Boolean	True
2.17	Verify that all authentication credentials for accessing services external to the application are encrypted and stored in a protected location	ALL	Boolean	True
2.18	Verify that a secure encrypted channel protects administrative interfaces if they accessible to untrusted parties.	ALL	Boolean	
2.19	Verify that browser autocomplete, and integration pages/functions that process sensitive data.	ALL	Boolean	True

4.4 Session Management Requirements

Session management refers to the mechanism by which a web application controls and maintains the state of the communication channel with which a user or entity uses to interact with it. All verified applications shall satisfy the following high level session management requirements:

- Sessions shall be unique to each communicating user or entity and should not be guessed or shared.
- Sessions shall be invalidated when they are no longer required and timed out after reconfigured periods of inactivity.

#	Key Performance Indicator	Level	KPI Type	Benchmark
3.1	Verify that there is no custom session manager, or that the custom session manager is resistant against all common session management attacks.	All	Boolean	True
3.2	Verify that sessions are invalidated when the user logs out.	All	Boolean	True
3.3	Verify that sessions timeout after a specified period of inactivity.	All	Boolean	True
3.4	Verify that all pages that require authentication have easy and visible access to logout functionality.	Critical	Boolean	True
3.5	Verify that the session id is never disclosed in URLs, error messages, or logs. This includes. verifying that the application does not support URL rewriting of session cookies.	Critical	Boolean	True
3.6	Verify that all successful authentication and re-authentication generate a new session and session id.	All	Boolean	True
3.7	Verify that only session ids generated by the application framework are recognized as active by the application.	Medium	Boolean	True
3.8	Verify that session ids are sufficiently long, random, and unique across the correct active session base.	All	Boolean	True
3.9	Verify that session ids stored in cookies have their path set to an appropriately restrictive value for the application, and authentication	All	Boolean	True

	session tokens. Additionally set the "HttpOnly" and "secure" attributes			

4.5 Access control verification requirements

Authorization is the concept of allowing access to resources only to those permitted to use them. All verified applications must satisfy the following high-level requirements:

- A user or entity accessing resources holds valid credentials to do so.
- Users are associated with a well-defined set of roles and privileges.
- Role and permission metadata is protected from replay or tampering.

#	Key Performance Indicator	Level	KPI Type	Benchmark
4.1	Verify that the principle of least privilege exists - users should only be able to access functions, data files, URLs, controllers, services, and other resources, for which they possess specific authorization. This implies protection against spoofing and elevation of privilege.	All	Boolean	True
4.2	Verify that access to sensitive records is protected, such that only authorized objects or data is accessible to each user (for example, protect against users tampering with a parameter to see or alter another user's account).	All	Boolean	True
4.3	Verify that directory browsing is disabled unless deliberately desired. Additionally, applications should not allow discovery or disclosure of file or directory metadata, such as Thumbs.db, .DS_Store,. git or .svn folders.	All	Boolean	True
4.4	Verify that access controls fail securely.	All	Boolean	True
4.5	Verify that the same access control rules implied by the presentation layer are enforced on the server side.	All	Boolean	True

4.6	Verify that all user and data attributes and policy information used by access controls cannot be manipulated by end users unless specifically authorized.	Medium	Boolean	True
4.7	Verify that there is a centralized mechanism (including libraries that call external authorization services) for protecting access to each type of protected resource.	Critical	Boolean	True
4.8	Verify that all access control decisions can be logged, and all failed decisions are logged.	Medium	Boolean	True
4.9	Verify that the application or framework uses strong random anti-CSRF tokens or has another transaction protection mechanism.	All	Boolean	True

4.6 Malicious Input Handling Verification Requirements

The most common web application security weakness is the failure to properly validate input coming from the client or from the environment before using it. This weakness leads to almost all the major vulnerabilities in web applications, such as cross-site scripting, SQL injection, interpreter injection, locale/Unicode attacks, file system attacks, and buffer overflows.

All verified applications must satisfy the following high-level requirements:

- All input shall be validated to be correct and fit for the intended purpose.
- Data from an external entity or client should never be trusted and should be handled. accordingly.

#	Key Performance Indicator	Level	KPI Type	Benchmark
5.1	Verify that the runtime environment is not susceptible to buffer overflows, or that security controls prevent buffer overflows.	All	Boolean	True
5.2	Verify that server-side input validation failures result in request rejection.	All	Boolean	True
5.3	Verify that all SQL queries, HQL, OSQL, NOSQL and stored procedures, calling of stored	Critical	Boolean	True

	procedures are protected using prepared statements or query parameterization, and thus not susceptible to SQL injection			
5.4	Verify that the application is not susceptible to LDAP Injection, or that security controls prevent LDAP Injection.	All	Boolean	True
5.5	Verify that the application is not susceptible to OS Command Injection, or that security controls prevent OS Command Injection.	All	Boolean	True
5.6	Verify that the application is not susceptible to Remote File Inclusion (RFI) or Local File Inclusion (LFI) when content is used that is a path to a file.	All	Boolean	True
5.7	Verify that the application is not susceptible to common XML attacks, such as XPath query tampering, XML External Entity attacks, and XML injection attacks.	All	Boolean	True
5.8	Ensure that all string variables placed into HTML or other web client code is either properly contextually encoded manually, or utilize templates that automatically encode contextually to ensure the application is not susceptible to reflected, stored and DOM Cross-Site Scripting (XSS) attacks.	Critical	Boolean	True
5.9	Verify that the application has defenses against HTTP parameter pollution attacks, particularly if the application framework makes no distinction about the source of request parameters (GET, POST, cookies, headers, environment, etc.)	All	Boolean	True
5.10	Verify that all input data is validated, not only HTML form fields but all	All	Boolean	True

	sources of input such as REST calls, query parameters, HTTP headers, cookies, batch files, RSS feeds, etc; using positive validation (whitelisting), then lesser forms of validation such as grey listing (eliminating known bad strings), or rejecting bad inputs (blacklisting).			
5.11	Make sure untrusted HTML from WYSIWYG editors or similar are properly sanitized with an HTML sanitizer and handle it appropriately according to the input validation task and encoding task.	All	Boolean	True
5.12	Verify that authenticated data is cleared from client storage, such as the browser DOM, after the session is terminated.	All	Boolean	True

4.7 Cryptography verification requirements

Cryptography refers to the scrambling of readable text, also called plaintext or cleartext, into an undecipherable cyphertext or encrypted text to uphold the confidentiality and integrity of sensitive data. This also refers to the unscrambling of data from cyphertext to plaintext.

All verified applications shall satisfy the following high-level requirements:

- That all cryptographic modules fail in a secure manner and that errors are handled correctly.
- That a suitable random number generator is used when randomness is required.
- That access to keys is managed in a secure way.

#	Key Performance Indicator	Level	KPI Type	Benchmark
6.1	Verify that all cryptographic modules fail securely, and errors are handled in a way that does not enable oracle padding.	Medium	Boolean	True
6.2	Verify that all random numbers, random file names, random GUIDs, and random strings are generated using the cryptographic module's approved random number generator in order to prevent them from being easily guessable by an attacker.	Medium	Boolean	True

6.3	Verify that cryptographic algorithms used by the application have been validated against an internationally approved cryptography verification standard.	Medium	Boolean	True
6.4	Verify that all keys and passwords are replaceable and are generated or replaced at installation time.	Medium	Boolean	True
6.5	Verify that random numbers are created with proper entropy even when the application is under heavy load, or that the application degrades gracefully in such circumstances.	Critical	Boolean	True

4.8 Error handling and logging verification requirements

The primary objective of error handling and logging is to provide a useful reaction by the user, administrators, and incident response teams. The objective is not to create massive amounts of logs, but manageable high-quality logs, with more signal than discarded noise.

High quality logs will often contain sensitive data and must be protected as per local data privacy laws or directives. This should include:

- Not collecting or logging sensitive information if not specifically required.
- Ensuring all logged information is handled securely and protected as per its data classification.
- Ensuring that logs are not forever but have an absolute lifetime that is as short as possible.

All verified applications shall satisfy the following high-level requirements:

#	Key Performance Indicator	Level	KPI Type	Benchmark
7.1	Verify that security logs are protected from unauthorized access and modification.	Medium	Boolean	True
7.2	Verify that the application does not log sensitive data that could assist an attacker, including user's session identifiers, passwords, hashes, or API tokens.	Medium	Boolean	True
7.3	Verify that an audit log or similar allows for non-repudiation of key transactions.	All	Boolean	True

7.4	Verify that the logs are stored on a different partition than the application is running with proper log rotation.	Critical	Boolean	True
7.5	Verify that time sources are synchronized to ensure logs have the correct time	All	Boolean	True

4.9 Data Protection Verification Requirements

Where an application transmits or stores sensitive information on insecure devices, such as shared computers, phones, and tablets, measures should be taken to ensure that data stored on these devices is encrypted and cannot be easily illicitly obtained, altered, or disclosed. All verified applications must satisfy the following high-level data protection requirements:

#	Key Performance Indicator	Level	KPI Type	Benchmark
8.1	Verify that all forms containing sensitive information have disabled client-side caching, including autocomplete features.	Medium	Boolean	True
8.2	Verify that all sensitive data processed by the application is identified, and that this data is securely accessed and encrypted where required.	Critical	Boolean	True
8.3	Verify that all sensitive data is sent to the server in the HTTP message body or headers (i.e., URL parameters are never used to send sensitive data).	All	Boolean	True
8.4	Verify that on the server, all cached or temporary copies of sensitive data stored are protected from unauthorized access or purged/invalidated after the authorized user accesses the sensitive data.	Medium	Boolean	True
8.5	Verify that there is a method to remove each type of sensitive data from the application at the end of the required retention policy.	Critical	Boolean	True
8.6	Verify the application minimizes the number of parameters in a request, such as hidden fields, cookies and header values.	Medium	Boolean	True

8.7	Verify that data stored in client-side storage (such as HTML5 local storage, session storage, Index edDB, regular cookies or Flash cookies) does not contain sensitive data or personally identifiable information.	All	Boolean	True
8.8	Verify that the accessing of sensitive data is logged	Medium	Boolean	True
8.9	Verify that sensitive information maintained in memory is overwritten with a mask as soon as it no longer required, to mitigate memory dumping attacks.	Critical	Boolean	True

4.10 Communications security verification requirements

All verified applications must satisfy the following high-level requirements:

- That TLS is used where sensitive data is transmitted.
- That strong algorithms and ciphers are used where sensitive data is transmitted.

#	Key Performance Indicator	Level	KPI Type	Benchmark
9.1	Verify that a path can be built from a trusted CA to each Transport Layer Security (TLS) server certificate, and that each server certificate is valid where sensitive data is transmitted.	ALL	Boolean	True
9.2	Verify that TLS is used for all external connections that are authenticated or that involve sensitive data or functions and ensure that the strongest alternative algorithm is used where TLS cannot be negotiated.	Critical	Boolean	True
9.3	Verify that backend TLS connection failures are logged.	Critical	Boolean	True
9.4	Verify that certificate paths are built and verified for all client certificates using configured trust anchors and revocation information.	Critical	Boolean	True
9.5	Verify that all connections to external systems that involve sensitive	Medium	Boolean	True

	information or functions are authenticated.			
9.6	Verify that HTTP Strict Transport Security headers are included on all requests and for all subdomains in order to protect against rotocol downgrade attacks and cookie hijacking	Critical	Boolean	True
9.7	Verify that proper certification revocation, such as Online Certificate Status Protocol (OCSP) Stapling, is enabled and configured.	ALL	Boolean	True
9.8	Verify that only strong algorithms, ciphers, and protocols are used, through all the certificate hierarchy, including root and intermediary certificates of your selected certifying authority	Critical	Boolean	True

4.11 HTTP security configuration verification requirements

All verified applications shall satisfy the following high-level requirements:

- The application server shall be suitably hardened from a default configuration.
- HTTP responses shall contain a safe character set in the content type header.

#	Key Performance Indicator	Level	KPI Type	Benchmark
10.1	Verify that the application accepts only a defined set of required HTTP request methods, such as GET and POST, and that unneeded methods (e.g. TRACE, PUT, and DELETE) are explicitly blocked.	Medium	Boolean	True
10.2	Verify that every HTTP response contains a content type header specifying a safe character set (e.g., UTF-8, ISO 8859-1).	All	Boolean	True
10.3	Verify that HTTP headers added by a trusted proxy or SSO devices, such as a	Medium	Boolean	True

	bearer token, are authenticated by the application.			
10.4	Verify that a suitable X-FRAME-OPTIONS header is in use for sites where content should not be viewed in a 3rd-party X-Frame.	Medium	Boolean	True
10.5	Verify that the HTTP headers or any part of the HTTP response do not expose detailed version information of system components.	Medium	Boolean	True
10.6	Verify that all API responses contain X-Content- Type-Options: nosniff and Content-Disposition: attachment; filename="api.json" (or other appropriate filename for the content type).	Medium	Boolean	True
10.7	Verify that a content security policy (CSPv2) is in place that helps mitigate common DOM, XSS, JSON, and JavaScript injection vulnerabilities.	Critical	Boolean	True
10.8	Verify that the X-XSS-Protection: 1; mode=block header is in place to enable browser reflected XSS filters.	Medium	Boolean	True

4.11 Files and resources verification requirements

All verified applications must satisfy the following high-level requirements:

- Untrusted file data should be handled accordingly and in a secure manner.
- File data obtained from untrusted sources must be stored outside the Webroot and with limited permissions.

#	Key Performance Indicator	Level	KPI Type	Benchmark
11.1	Verify that untrusted file data submitted to the application is not used directly with file I/O commands, particularly to protect against path traversal, local file include, tile mime type, and OS command injection vulnerabilities.		Boolean	True
11.2	Verify that files obtained from untrusted sources are validated to be of expected		Boolean	True

	type and scanned by antivirus scanners to prevent upload of known malicious content			
11.3	Verify that untrusted data is not used within inclusion, class loader, or reflection capabilities to prevent remote/local file inclusion vulnerabilities.		Boolean	True
11.4	Verify that files obtained from untrusted sources are stored outside the webroot, with limited permissions, preferably with strong validation.		Boolean	True
11.5	Verify that the web or application server is configured by default to deny access to remote resources or systems outside the web or application server.		Boolean	True
11.6	Verify the application code does not execute uploaded data obtained from untrusted sources.		Boolean	True
11.7	Do not use Flash, Active-X, Silverlight, NACL, (client-side Java or other client-side technologies not supported natively via W3C browser standards.		Boolean	True
11.8	Verify that all untrusted or r party data is not embedded into the application, and that URL redirects and forwards are used together with appropriate X-FRAME for the purpose of such data.		Boolean	True

4.12 Mobile application security verification requirements

All mobile applications must satisfy the following high-level requirements:

- Mobile applications should have the same level of security controls within the mobile client as found in the server, by enforcing security controls in a trusted environment.
- Sensitive information assets stored on the device should be done so in a secure manner.
- All sensitive data transmitted from the device should be done so with the transport layer security in mind.

#	Key Performance Indicator	Level	KPI Type	Benchmark
12.1	Verify that ID values stored on the device and retrievable by other applications, such as the UDID or IMEI number are not used as authentication tokens	All	Boolean	True
12.2	Verify that the mobile app does not store sensitive data onto potentially unencrypted shared resources on the device (e.g. SD card or shared folders).	All	Boolean	True
12.3	Verify that sensitive data is not stored unprotected on the device, even in system protected areas such as key chains.	All	Boolean	
12.4	Verify that secret keys, API tokens, or passwords are dynamically generated in mobile applications.	All		True
12.5	Verify that the mobile app prevents leaking of sensitive information (for example, that no screenshots are saved of the current application as the application is backgrounded or that no sensitive information is written in console).	Medium	Boolean	True
12.6	Verify that the application is requesting minimal permissions for required functionality and resources	Medium	Boolean	True
12.7	Verify that the application sensitive code is laid out unpredictably in memory (For example ASLR).	Medium	Boolean	True
12.8	Verify that there are anti-debugging techniques present that are sufficient to deter or delay likely attackers from injecting debuggers into the mobile app (For example GDB).	Critical	Boolean	True

12.9	Verify that the app does not export sensitive activities, intents, or content providers for other mobile apps on the same device to exploit.	All	Boolean	True
12.10	Verify that sensitive information maintained in memory is overwritten with a mask as soon as it no longer required, to mitigate memory dumping attacks.	Medium	Boolean	True
12.11	Verify that the app validates input to exported activities, intents, or content providers.	All	Boolean	True

4.13 Accountability and Responsibility

- The overall responsibility for adherence to this standard lies with the Senior Manager:
- Information Security. In his or her absence however, the designated person shall assume responsibility.
- This describes the overall accountability and responsibility of adherence and may describe responsibilities of other personnel.

5. Process for Monitoring

This procedure's effective implementation and monitoring will be done through the executive committee, and internal audits will be conducted to determine compliance and implementation.

Monitoring controls	Purpose	Responsible	Frequency
Monthly reviews	Report and review adherence of the procedure	IT Service Desk	Monthly
Monthly reports	Review and determine the effectiveness and adherence of the procedure	IT Service Quality and Change Manager	Monthly
Management Review	Measure implementation and adherence of the procedure	Senior Manager: Digital Infrastructure and Operations	Annually

Note: This procedure shall be reviewed in three-year cycle and if there is a need to review the procedure before three-year cycle laps due to any circumstances being legal requirements, changes in the businesses, the need to reflect current practices or activities, the procedure will be unlocked for review accordingly.

Disclaimer: In instances where document links are not accessible, directly access the documents on the Procedure Management Document Store on the Airports Company South Africa SOC Limited intranet.

6. Accountabilities and Responsibilities

6.1 Accountabilities

The overall accountability for development and implementation of this procedure lies with Chief Executive Officer and Chief Information Officer with the support of the Senior Manager Digital Infrastructure and Operations as a responsible person for actual development and implementation of this procedure, however, in the absence of the Chief Information Officer, a delegated person shall assume responsibility as per delegation of authority procedure.

Authorities	Employees	IT Service Quality and Change Manager	Senior Manager: Digital Infrastructure and Operations	Chief Technology Officer	Chief Information Officer
Development and implementation of this procedure	-	*Accountable*	*Responsible*	*Consulted*	*Consulted*
Implementation and adherence of this procedure	*Responsible*	*Responsible*	*Accountable*	*Informed*	*Informed*
Approval and authorisation	-	*Responsible*	*Responsible*	*Accountable*	*Responsible*
Communicate the procedure to all impacted stakeholders or employees.	*Informed*	*Informed*	*Responsible*	*Informed*	*Accountable*

7. Reporting of Non-Conformance

Any deviation from this procedure shall be identified and registered with corrective and preventative measures for continual improvement in accordance with Non-Conformance and Non-Compliance Procedure Document - Z001 001M.

8. Related Policy Documents

Document Control Procedure - Z001 006M
Record Keeping Requirements Procedure - Z001 008M

9. Related Legislation and Standard

Quality Management System ISO 9001

10. Change Control and Verification

This procedure shall only be changed with the authorisation of the Group Executive: Chief Information Officer and in accordance with Change Control and Verification Procedure- Z001 003M

11. Records

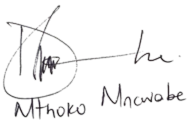
Each Process Owner and managers as identified are responsible for maintaining, storage and protection of their respective documents/ information. Records shall be identifiable, easily retrievable and maintained as per the Retention schedule as regulated or required by the organisation, statutory or regulatory requirements. Refer Record Keeping Requirements Procedure - Z001 008M.

Record Name	Storage Location	Record Number	Responsible Person	Retention Time
Web Application Security Procedure	Master in Corporate Procedure Document Store	G010 011M	Procedure Assurance Officer	Five (5) years

12. Revision History

Date last revised	Revision Status	Compiler	Summary of changes
January 2024	Version: 1	**Cyber Security Manager** **Name and Surname** Coster Baloyi	First Issue

13. Endorsement (See Master in Corporate Policy Document Store)

Activity	Name	Signature	Date
Compiled by	**Position:** Cyber Security Manager **Name and Surname** Coster Baloyi		12/02/2024
Quality Assurance: Policy Documents	**Position:** Specialist: Policy Assurance and Ethics **Name and Surname** Thabana Mahlo		19/02/2024
Supported by	**Position:** Group Manager Cyber Security **Name and Surname** Tefo Moreki		20/02/2024
Authoriser	**Position:** Chief Information Officer **Name and Surname** Mthoko Mncwabe		21/02/2024