November 2022

INTERNAL

Secure Coding Policy

FUNDS AXIS

Policy title:	Network Security Policy

Issue	1.0
Approved by:	Trevor Dempster
Approval Date:	November 2022
Next Review Date:	November 2023

Scope:	The policy applies to Funds-Axis Limited and all contractors and other people working on behalf of the company.
Associated documentation:	\ Secure Development Environment Guidelines \ Secure Development Policy \ Principles for Engineering Secure Systems
Responsibility for Implementation & Training:	Day to day responsibility for implementation: ISO Day to day responsibility for training: ISO

Distribution methods:	Methods used to communicate this policy: \ Training

FUNDS AXIS

1. Introduction

Developing bespoke software provides a level of flexibility and functionality that is not always possible to reproduce using commercial off the shelf systems. Funds-Axis dedicates a significant amount of its resources to designing, coding, testing, releasing and maintaining its software according to industry best practice standards. As part of this, we have a responsibility to ensure that the computer code we produce is as secure as it reasonably can be, Insecure software allows vulnerabilities to be found which can provide a way in for malicious actors and requires our time to create and distribute patches.

In order to write software that minimises such vulnerabilities, there are a number of guiding principles that must be followed, in addition to the more detailed programming techniques that apply with specific languages. This policy defines these high-level principles as a starting point for the definition of lower-level procedures for the creation of secure code, as part of an effective managed approach.

This control applies to all systems people and processes that constitute the organisations information systems, including board members, directors, employees, suppliers and other third parties who have access to Funds-Axis systems.

The following policies and procedures are relevant to this document:

- Secure Development Environment Guidelines
- Secure Development Policy
- Principles for Engineering Secure Systems

2. Secure Coding Policy

It is the policy of Funds-Axis to write software in such a way that the number of potential vulnerabilities in the code is minimised.

Secure coding within Funds-Axis will be based on the principles established by best practice organisations including (but not limited to) the following:

- OWASP (The Open Web Application Security Project*)
- SEI CERT (Software Engineering Institute Computer Emergency Response Team)
- UK NCSC (National Cyber Security Centre)
- USA NIST (National Institute of Standards and Technology)

Secure coding practices in use within the organisation will cover as a minimum the following topics (based on OWASP Secure Coding Practices V2.0):

- Input validation
- Output encoding
- Authentication and password management
- Session management
- Access control
- Cryptographic practices
- Error handling and logging
- Data protection
- Communication security
- System configuration
- Database security
- File management
- Memory management
- General coding practices

These general principles will be supplemented by technology-specific advice and guidance produced by the vendors of the technology in use, and third parties with particular expertise in them.

Secure coding practices will be established and documented for each development project, and will be communicated to third parties that create software on Funds-Axis's behalf. Account will be taken of available threat intelligence and existing known vulnerabilities when defining these practices.

Good practice in writing code will be followed at all times including, where appropriate:

- The use of structured programming techniques
- Clear documentation and commenting of code
- Consistent naming of items such as classes, methods and variables
- Avoiding hard coding of credentials

\ Correct handling of errors

Appropriate software testing will be carried out to confirm that the documented coding techniques have been properly implemented prior to the release of the software to production.

Where possible, the use of secure coding techniques with me mandated via settings and automated within development tools, such as integrated development environments (IDE).

External software libraries used as part of the development process must be examined to assess them against the secure coding practices adopted for the relevant project.

A process must be in place for the management of vulnerabilities discovered after the release of the software into production.

Software fixes and updates must be subject to the same secure coding practices as the original development.