

Table of Contents

1. Objective	3
2. Scope	3
3. Compliance Table	3
4. Review	3
5. Execution Responsibility	3
6. System Lifecycle	3
7. Systems	4
7.1 Discover Information Protection Needs	4
7.2 Define System Security Requirements	5
7.3 Design System Security Architecture	5
7.4 Develop Detailed Security Design	5
7.5 Implement System Security	5
7.6 Security Objectives	6
7.7 Security Design Guidelines	6
7.8 Threat Modeling	7
7.9 Security Architecture and Design Review	7
7.10 Security Code Review	8
7.11 Security Testing	8
7.12 Security Deployment Review	8
8. Appendix	8
8.1 Principles of Security Engineering	8
9. Key Terms and Definitions	11

DOCUMENT CONTROL

VERSION CONTROL

Document Control ID	EFR-Security Engineering Procedure-021
Issued Date	
Effective Date	
Owner	Information Security & Compliance Department

REVISION TABLE

Date	Version	Brief Description	Author
11/11/2024	1.0	Draft-Security Engineering Procedure	Ms. Ratisha Kukreja
03/12/2024	1.0	Final-Security Engineering Procedure	Ms. Ratisha Kukreja

RELEASE AUTHORIZATION

Task	Author	Title	Date
Prepared by	Ms. Ratisha Kukreja	Consultant	

REVIEWER AUTHORIZATION

Task	Author	Title	Date
Reviewed By	Mr. Antony Arul Selvaraj	Consultant	

APPROVAL AUTHORIZATION

Name	Title	Signature	Date

Important Note: This document is intended solely for the use of the individual or entity to whom it is transmitted to and others authorized to receive it. It may contain confidential or legally privileged information. If you are not the intended recipient you are hereby notified that any disclosure, copying, distribution or taking any action in reliance on the contents of this document is strictly prohibited and may be unlawful. If you have received this document in error, please notify us immediately.

1. Objective

EFR shall consider and follow system-level security principles in the design, development, and operation of information systems.

2. Scope

Security Engineering principles are applicable to all IT systems, including general support systems and major applications. The established security engineering principles shall be applied, where applicable, to outsourced information systems through the contracts and other binding agreements between EFR and the supplier to whom EFR outsources.

3. Compliance Table

Policy/Procedure Name	ISO 27001:2022 Control	UAE IA Control	Requirements
Security Engineering Procedure	A 8.27, A 8.28, A 8.29	-	NIST SP800-144 Guidelines.

4. Review

The procedure needs to be reviewed by EFR once a year, else whenever there are any changes implemented in the existing procedure.

5. Execution Responsibility

The principles presented herein can be used by:

- **Users** - When developing and evaluating functional requirements, or when operating information systems within EFR.
- **System Engineers and Architects** - When designing, implementing, monitoring, or modifying an information system.
- **IT Specialists** - During all phases of the system's life cycle.
- **IT security practitioners** - Ensure the correct implementation of IT system information system security requirements.
- **Program Managers, Information System Security Officers and IS auditors** - To ensure adequate security measures have been considered for all phases of the system life cycle. Also, provide the required resources and meet responsibilities under the principle of due care.
- **Security awareness trainers** - Incorporate security and privacy awareness training programs for the organization's personnel.

6. System Lifecycle

All systems shall be a part of the following phases within their lifecycle:

- **Initiation:** During the initiation phase, the need for a system is expressed and the purpose of the system is documented.

- **Development/Acquisition:** During this phase, the system is designed, purchased, programmed, developed, or otherwise constructed.
- **Implementation:** During implementation, the system is tested and installed or fielded.
- **Operation/Maintenance:** During this phase, the system performs its work. Typically, the system is also being modified by the addition of hardware and software and by numerous other events.
- **Disposal:** The disposal phase of the IT system life cycle involves the disposition of information, hardware, and software.

Principles for engineering secure systems should be integrated into the system lifecycle at appropriate phases.

7. Systems

EFR shall ensure to incorporate the following Information Systems Security Engineering (ISSE) process activities within their system lifecycle.

- Discover Information Protection Needs
- Define System Security Requirements
- Design System Security Architecture
- Develop Detailed Security Design
- Implement System Security
- Assessing Information Protection Effectiveness

Each of the six ISSE process activities is explained below. These activities must be performed by a collaborative effort of system engineers and ISSE professionals. Management support, as appropriate, should be obtained for facilitating each phase of the security engineering process.

7.1 Discover Information Protection Needs

The information systems security engineer can obtain a portion of the information required for this activity from the Systems Engineering (SE) process. The objectives of this activity are to understand and document the customers' needs and to develop solutions that will meet these needs.

In the Discover Information Protection Needs activity of the ISSE process, the information systems security engineer must document all elements of the activity. These elements include:

- Roles
- Responsibilities
- Threats
- Strengths
- Security services
- Priorities

- Design constraints

7.2 Define System Security Requirements

The information systems security engineer identifies one or more solution sets that can satisfy the information protection needs of the information protection. The solutions shall be selected by appropriately considering the needs of external systems such as Public Key Infrastructure (PKI) or other cryptographic-related systems.

A solution set consists of preliminary security CONOPS (Concept of Operations), the system context, and the system requirements.

7.3 Design System Security Architecture

The information systems security engineer:

- Performs a functional decomposition of the requirements that can be used to select the components required to implement the designated functions.
- Determines, at a functional level, the security services that should be assigned to the system to be protected as well as to external systems. Such services include encryption, key management, and digital signatures.

7.4 Develop Detailed Security Design

The design activity is iterative, and it involves both the systems engineer and the ISSE professionals to accomplish the below listed activities.

- Mapping security mechanisms to system security design elements and the security requirements onto the security design.
- Qualifying external and internal element and system interfaces.
- Ensuring that the design meets the customer's design constraints.
- Addressing both technical and non-technical information security mechanisms within the design components.

The information systems security engineer must consider:

- Trade-offs that might have to be made among cost, priorities, performance, schedule, and remaining security risks.
- The effects and costs of long-lead-time items and life cycle support requirements.

This activity should specify the system and components and not the products or vendors.

7.5 Implement System Security

- This activity moves the system from the design phase to the operational phase, a system effectiveness assessment that produces evidence that the system meets the requirements.
- The assessment is accomplished through the following actions of the information systems security engineer.

- Verifying that the implemented system does address and protect against the threats itemized in the original threat assessment.
- Application of information protection assurance mechanisms related to system implementation and testing.
- Providing inputs to and reviewing the evolving system life cycle support plans.
- Providing inputs to and reviewing the operational procedures.
- Taking part in multidisciplinary examinations of all system issues and concerns.
- Additional tasks related to the Implement System Security activity conducted by the systems and design engineers in cooperation with the information systems security engineer include:
 - Conducting unit testing of components.
 - Developing test procedures so that the designed system performs as required.
 - Developing documentation and placing documentation under version control.
 - Developing test plans and procedures.

7.6 Security Objectives

Security objectives are goals and constraints that affect the confidentiality, integrity, and availability of your data and application.

Security objectives are unique for each application. The outcome of this phase shall be a list of application vulnerabilities and potential problems. Certain security objective categories can be (but not limited to):

- Tangible assets to protect
- Intangible assets to protect
- Compliance requirements
- Quality of service requirements

When an application supports multiple roles, a 'Role Matrix' shall be created to understand what each role should be allowed to do. Finally, the security objectives can be generated from the Role Matrix, to ensure the integrity of the application's roles mechanism.

Security objectives may not remain static and are influenced by later design and implementation activities.

7.7 Security Design Guidelines

To avoid many of the vulnerabilities introduced by poor design choices, the design activity will use proven design practices, patterns, and principles. Design guidelines represent proven practices that have evolved over time to reduce risks associated with designing applications.

Depending on the application being designed, the types of issues that must be addressed shall vary. Some of the key application design issues may be (but not limited to) based on:

- Input / Data Validation
- Authentication
- Authorization
- Configuration Management
- Sensitive Data
- Cryptography
- Exception Management
- Auditing and Logging

7.8 Threat Modeling

Threat modeling will help in understanding and identifying the threats and vulnerabilities that are relevant to the specific application scenario.

The five threat modeling steps are:

Step 1: Identify Security Objectives

Clear objectives help to focus the threat modelling activity and determine how much effort to spend on subsequent steps.

Step 2: Create an Application Overview

Itemizing the application's important characteristics and actors helps to identify relevant threats during step 4.

Step 3: Decompose the Application

A detailed understanding of the mechanics of the application makes it easier to uncover more relevant and more detailed threats.

Step 4: Identify Threats

Use details from steps 2 and 3 to identify threats relevant to the application scenario and context.

Step 5: Identify Vulnerabilities

Review the layers of the application to identify weaknesses in the context of those threats. Identify common vulnerable areas by using vulnerability categories.

7.9 Security Architecture and Design Review

The architecture and design review process analyses the architecture and design from a security perspective. It examines several aspects including deployment and infrastructure, overall application architecture and design, and each tier in the application.

EFR shall use knowledge of attack patterns & security best practices for each application to ensure compliance with your team's design guidelines.

7.10 Security Code Review

All codes will be subjected to code inspections where the emphasis is on identifying security vulnerabilities. This will be a continuous activity during the development and test phases of the application life cycle.

EFR may use static analysis to find an initial set of bugs and improve the understanding of where bugs are most likely to be discovered during further review. Manually review any custom security mechanism or any feature designed specifically to mitigate a known security threat.

7.11 Security Testing

New technology should be analyzed for security risks and the design should be reviewed against known attack patterns.

EFR may use a risk-based approach and use the output from the threat modeling activity to help establish the scope of your testing activities and define your test plans. EFR may use automated application security testing tools to identify weaknesses in the application that can compromise the applications.

7.12 Security Deployment Review

When the application is being deployed, EFR will ensure that weak or inappropriate configuration settings are not present, which may introduce security vulnerabilities.

Since application security is dependent upon the security of the underlying infrastructure on which the application is deployed, the deployment review, depending upon the application, will cover configuration of both the network and the host.

Security of the application platform should be reviewed for appropriate application types. Some application types will require review of application security categories as well as server security categories. For example, application-level Web.config file settings should be reviewed.

8. Appendix

8.1 Principles of Security Engineering

The principles described here do not always apply to all systems. Yet, each principle should be carefully considered throughout the life cycle of every system. Below table associates each principle with the relevant life cycle planning phase(s):

SI. No	Principle	Initiation	Development / Acquisition	Implementation	Operation / Maintenance	Disposal
1.	Establish a sound security policy as the "foundation" for design	√√	√	√	√	√
2.	Treat security as an integral part of the overall system design	√√	√√	√√	√√	√
3.	Clearly delineate the physical and logical security boundaries governed by associated security policies	√√	√√	√	√	

Sl. No	Principle	Initiation	Development / Acquisition	Implementation	Operation / Maintenance	Disposal
4.	Ensure that developers are trained in how to develop secure software	√√	√√	√		
5.	Reduce risk to an acceptable level	√√	√√	√√	√√	√√
6.	Assume that external systems are insecure	√√	√√	√√	√√	√
7.	Identify potential trade-offs between reducing risk and increased costs and decrease in other aspects of operational effectiveness	√√	√√		√√	
8.	Implement tailored system security measures to meet organizational security goals	√	√√	√	√√	√
9.	Protect information while being processed, in transit, and in storage	√	√√	√	√√	√
10.	Consider custom products to achieve adequate security	√	√√	√	√	
11.	Protect against all likely classes of "attacks."	√	√√	√	√	√
12.	Where possible, base security on open standards for portability and interoperability	√	√√	√		
13.	Use common language in developing security requirements	√√	√√		√√	
14.	Design security to allow for regular adoption of new technology, including a secure and logical technology upgrade process		√√	√	√√	
15.	Strive for operational ease of use	√	√√	√	√√	
16.	Implement layered security (Ensure no single point of vulnerability)	√	√√	√	√√	√
17.	Design and operate an IT system to limit damage and to be resilient in response	√	√√		√√	
18.	Provide assurance that the system is, and continues to be, resilient in the face of expected threats	√	√√	√	√√	√
19.	Limit or contain vulnerabilities		√√	√	√	
20.	Isolate public access systems from mission critical resources	√	√√	√	√	
21.	Use boundary mechanisms to separate computing		√√	√	√√	

Sl. No	Principle	Initiation	Development / Acquisition	Implementation	Operation / Maintenance	Disposal
	systems and network infrastructures					
22.	Design and implement audit mechanisms to detect unauthorized use and to support incident investigations	√	√√	√√	√	
23.	Develop and exercise contingency or disaster recovery procedures to ensure appropriate availability	√	√	√	√√	
24.	Strive for simplicity	√	√√	√	√√	
25.	Minimize the system elements to be trusted	√	√√	√	√√	
26.	Implement least privilege	√	√	√	√√	
27.	Do not implement unnecessary security mechanisms	√	√√	√√	√	√
28.	Ensure proper security in the shutdown or disposal of a system		√		√	√√
29.	Identify and prevent common errors and vulnerabilities		√√	√√	√	
30.	Implement security through a combination of measures distributed physically and logically		√√	√	√	√
31.	Formulate security measures to address multiple overlapping information domains	√	√√	√	√	
32.	Authenticate users and processes to ensure appropriate access control decisions both within and across domains	√	√	√	√√	
33.	Use unique identities to ensure accountability	√	√	√	√√	

Note: One check “√” signifies the principle can be used to support the life-cycle phase, and two checks “√√” signifies the principle is very important and key to successful completion of the life-cycle phase.

9. Key Terms and Definitions

Sl. No	TERM	DEFINITION
1.	Assurance	Grounds for confidence that the other four security goals (integrity, availability, confidentiality, and accountability) have been adequately met by a specific implementation.
2.	General Support System	An interconnected information resource under the same direct management control, which shares common functionality. It normally includes hardware, software, information, data, applications, communications, facilities, and people and provides support for a variety of users and/or applications.
3.	CONOPS	Concept of Operations is a document describing the characteristics of a proposed system from the viewpoint of an individual who will use that system. It is used to communicate quantitative and qualitative system characteristics
4.	Security Engineering	Specialized field of engineering that focuses on the security aspects in the design of systems that need to be able to deal robustly with possible sources of disruption, ranging from natural disasters to malicious acts.
5.	Security Requirements	The types and levels of protection that are necessary for equipment, data, information, applications, and facilities to meet security policy
6.	Threat	Any circumstance or event with the potential to cause harm to an IT system in the form of destruction, disclosure, adverse modification of data, and/or denial of service.

--End of Document--