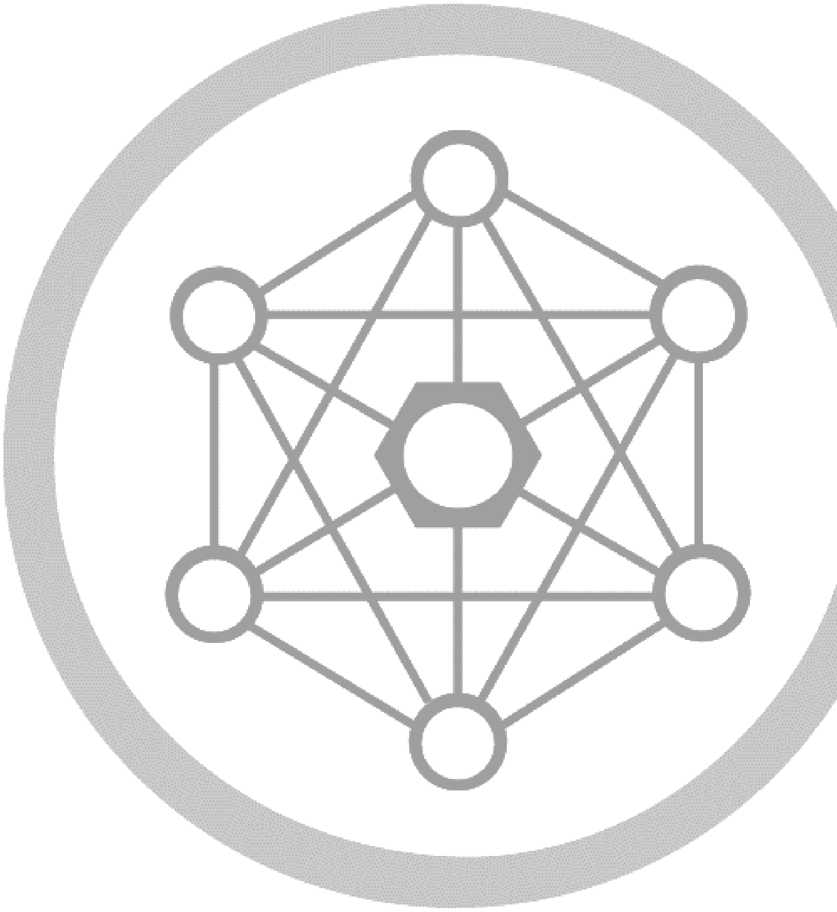
Secure Software Development Lifecycle

Int.prc.002.02 | 01.03.2024

UPM 24.x

ZINFI Confidential & Proprietary

Shared Under NDA



Contents

Introduction

ZINFI adopts a state-of-the-art agile systems development lifecycle (SDLC) methodology for the development and implementation of channel management solutions, with security as the top priority. In the past, security was applied at the commissioning stage of the development process and was force-fitted into the final design, resulting in an inconsistently secured application. Security was eventually integrated into every step of the SDLC, from initiation to development to deployment and eventual disposal of the system, following a security-by-design (SBD) approach.

Security-by-design is an approach to software and hardware development that seeks to minimize systems' vulnerabilities and reduce the attack surface by designing and building security into every phase of the SDLC. This includes incorporating security specifications in the design, continuous security evaluation at each phase and adherence to best practices.

The benefits of integrating security into SDLC include:

- Early identification and mitigation of security vulnerabilities and misconfigurations of systems.
- Identification of shared security services and tools to reduce cost, while improving security posture through proven methods and techniques.
- Facilitation of informed key stakeholder decisions through comprehensive risk management in a timely manner.
- Documentation of important security decisions throughout the lifecycle of the system, ensuring that security is fully considered during all phases.
- Improved systems operability that could otherwise be hampered by isolated security of systems.

Specific to cybersecurity, security-by-design addresses security considerations throughout the system's lifecycle, and includes security measures designed specifically for identification, protection, detection, response and recovery capabilities to strengthen the cyber resiliency of our systems.

Definitions

Technical terms used in this document include:

Process – The Institute of Electrical and Electronics Engineers (IEEE) defines a process as "a sequence of steps performed for a given purpose." A secure software process can be defined as the set of activities performed to develop, maintain and deliver a secure software solution. Activities may not necessarily be sequential; they could be concurrent or iterative.

Process model – A process model provides a reference set of best practices that can be used for both process improvement and process assessment. Process models do not define processes;

rather, they define the characteristics of processes. Process models usually have an architecture or a structure. Groups of best practices that lead to achieving common goals are grouped into process areas, and similar process areas may further be grouped into categories.

The prospect of building a secure system is more likely when we follow solid software engineering practices with an emphasis on good design, quality practices such as inspections and reviews, use of thorough testing methods, appropriate use of tools, as well as risk management, project management and people management.

Standards – Standards are established by an authority or a custom or by general consent as examples of best practices. Standards provide material suitable for the definition of processes.

Assessments, evaluations, appraisals – All three of these terms imply the comparison of a process being practiced to a reference process model or standard. Assessments, evaluations and appraisals are used to understand process capability in order to improve processes. They help determine whether the processes being practiced are adequately specified, designed, integrated and implemented to support the needs—including the security needs—of the software product.

Software assurance – The National Institute of Standards and Technology (NIST) defines software assurance as “the level of confidence that software is free from vulnerabilities, either intentionally designed into the software or accidentally inserted at any time during its life cycle, and that the software functions in the intended manner.”

Security assurance – Although the term “security assurance” is often used, there does not seem to be an agreed-upon definition for this term. The System Security Engineering Capability Maturity Model (SSE-CMM) describes security assurance as the process that establishes confidence that a product’s security needs are being met. In general, the term refers to the activities, methods and procedures that provide confidence in the security-related properties and functions of a developed solution. Security assurance usually includes activities for the requirements, design, implementation, testing, release and maintenance phases of an SDLC.

Focus areas

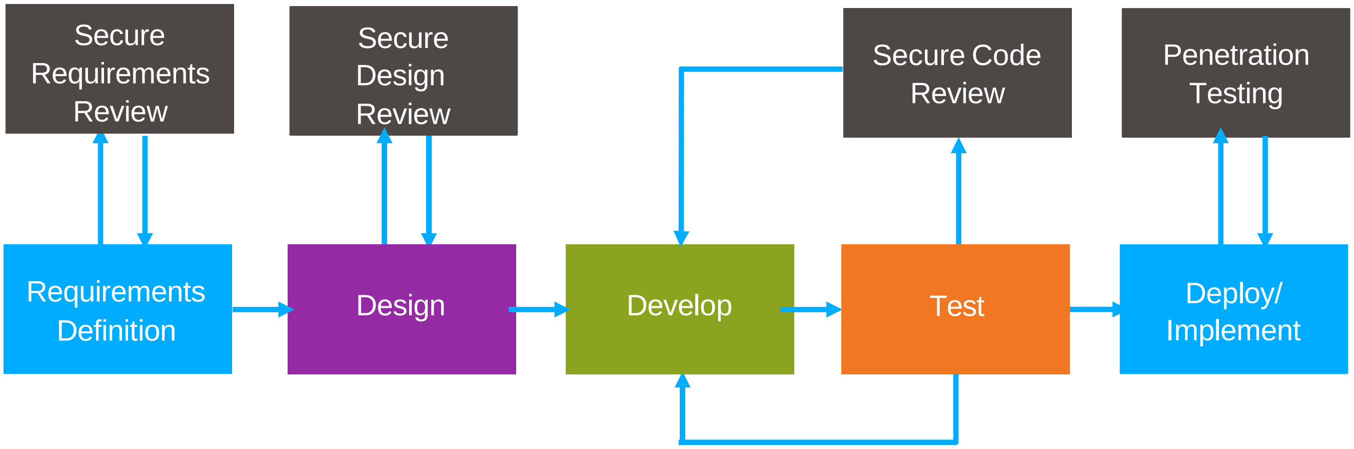
Security engineering activities – Security engineering activities include activities needed to engineer a secure solution. Examples include elicitation and definition of security requirements, secure design based on design principles for security, use of static analysis tools, secure reviews and inspections, and secure testing.

Security assurance activities – Assurance activities include verification, validation, application review, artifact review and evaluations.

Security organizational and project management activities – Organizational activities include the development of organizational policies, incorporation of senior management insight, and the establishment of organizational roles and other activities that support security. Project management activities include project planning as well as tracking resource allocation and usage to ensure that the security engineering, security assurance and risk identification activities are planned, managed and tracked.

Security risk identification and management activities – Identification and management of security risks is one of the most important activities in a secure SDLC and, in fact, is the driver for subsequent activities. Security risks in turn drive other security engineering activities, project management activities and security assurance activities.

Base Framework Overview



The base framework traditionally is defined by the following sub-processes:

Secure requirements review (requirements definition) – The secure requirements review involves conducting a review of functional requirements that specify the business logic and overall behavior for each project implementation. After requirements are gathered for a project, an assessment is conducted to derive relevant security requirements.

Secure design review (system design) – Design-level flaws are lesser-known concepts, but their presence is a very big risk to the application. Such flaws are hard to find in static or dynamic application scans and instead require deep understanding of application architecture and layout to uncover manually. The focus of secure design review is to explain the measures necessary to prevent flaws while designing the implementation. The aim of this step is to produce a secure design checklist or a threat model for reviewing the application design.

Secure code review (develop/test) – Secure code review is the process of auditing the source code for the application to verify that the proper security controls are present, work as intended and have been invoked in all the right places. Code review ensures that the application has been developed to be "self-defending" in its given environment. A general rule of thumb is that a penetration test should not discover any additional application vulnerabilities relating to the developed code after the application has undergone a proper security code review.

Penetration testing (deploy/implement) – The main objective of penetration testing is to identify security weaknesses. It is used to test the application against the organization's security policy, its adherence to compliance requirements, its employees' security awareness and the organization's ability to identify and respond to security incidents.

Migrating to Secure Agile

ZINFI has upgraded to a SDLC model based on agile development lifecycle as the de facto methodology for our development cycles. Agile development lifecycle describes a set of principles for systems development under which requirements and solutions evolve through the collaborative effort of self-organizing cross-functional teams. It arises from the need to develop quick iterations of working systems to users who have changing requirements and priorities.

Agile development lifecycle comprises six phases:

- **Concept** – This is a pre-iteration phase where the need for the system is expressed and the functional specifications of the system are documented.
- **Inception/warmup** – The first week of an agile project is often referred to as Iteration 0 and is used to set up the environment and gather support and funding for the project.
- **Construction iterations** – The system is incrementally and iteratively built and delivered to meet the changing needs of the stakeholders. Continual testing is also performed during this phase.
- **Transition** – Final testing and rework are performed on the system before it is released into production. Finalized documentation and training are also performed at this phase.
- **Production** – In this phase, the system is operational and producing the work as per specifications. Modifications and enhancements are managed through a formal change management process. Maintenance of the software and system upgrades are also performed at this phase.
- **Retirement** – When the system is redundant or obsolete, the system will be disposed. This includes orderly termination of the system, safeguarding of vital systems information, and migration of data to a new system, or preservation of data in accordance with applicable records management regulation and policies.

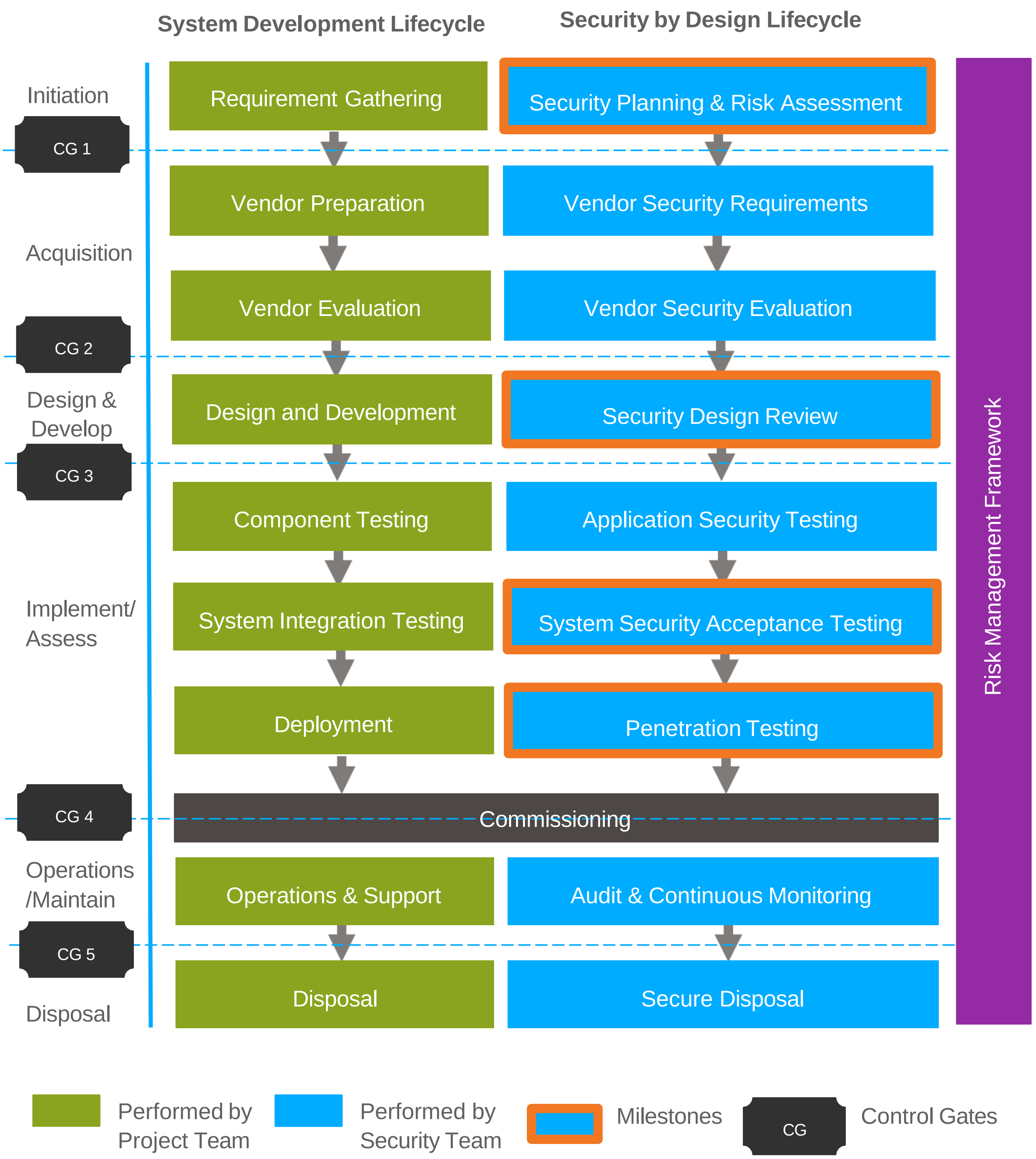
The security-by-design (SBD) framework as described in this document is adaptable to our agile development lifecycle. The SBD lifecycle parallels the SDLC phases by incorporating security considerations into processes at every phase. Because security risks need to be identified as early as the planning phase and addressed accordingly throughout the subsequent phases, SBD spans all phases of the SDLC. At a broad level, security risks can be addressed through:

- Changing the requirements or deployment to avoid the security risk
- Implementing alternatives or mitigating controls
- Accepting the risk through proper risk management processes

- Iterative processes where security is evaluated at each phase and it is determined whether the security processes need to be repeated to produce a satisfactory output.

Introducing security alongside each SDLC phase ensures that security risks are visible and well-understood by senior management and key personnel, and appropriate decisions are taken in a timely manner to reduce risk to an acceptable level.

SBD Framework



		System Development Lifecycle	Security by Design Lifecycle	
Initiation		Requirement Gathering	Security Planning & Risk Assessment	Risk Management Framework
	CG 1			
Acquisition		Vendor Preparation	Vendor Security Requirements	
		Vendor Evaluation	Vendor Security Evaluation	
	CG 2			
Design & Develop		Design and Development	Security Design Review	
	CG 3			
Implement/ Assess		Component Testing	Application Security Testing	
		System Integration Testing	System Security Acceptance Testing	
		Deployment	Penetration Testing	
	CG 4	Commissioning		
Operations /Maintain		Operations & Support	Audit & Continuous Monitoring	
	CG 5			
Disposal		Disposal	Secure Disposal	

🟩 Performed by Project Team
🟦 Performed by Security Team
🟧 Milestones
CG Control Gates

Our SBD approach consists of three components:

- **Lifecycle** – Alignment of security-related processes with SDLC to guide our projects to meet SBD objectives.
- **Activities** – Security-related activities that support the security lifecycle processes.

  Under each security process is a set of security-focused activities that describe the key security actions to be taken. Each activity covers, at minimum:

  (a) Description – Describe the actions to be taken in parallel with SBD processes and activities.

  (b) Roles and responsibilities – Describe key roles and responsibilities within each activity and the actions that they are responsible for.

  (c) Expected outputs – Describe the required security-related artifacts that are expected from this activity, which may be inputs into other related activities.

  (d) Inter-dependencies – Describe the inter-dependencies with other SDLC/SBD activities and outputs, and how they work together to enhance security of the system.

- **Control gates** – Decision points or specific milestones of the SBD phases where the security implementations are evaluated. These provide the organization with an opportunity to verify that security considerations are addressed, adequate security controls are built in, and identified risks are clearly understood before the system development advances to the next lifecycle phase.

Phases and Control Gates

Initiation

In the initiation phase, threats, security requirements and potential constraints of functionality and integration are considered. Security is looked at from the perspective of business risks with inputs from the security team.

Security process:

- Security planning and risk assessment

Activities under this process include:

- Security planning, which establishes common understanding of security goals and objectives, identifies key security roles and develops a high-level security schedule.
- Classification of the system to the appropriate security classification.
- Threat and risk assessment (TRA) to ensure threats, risk and security decisions are documented, assessed and approved by key stakeholders. The TRA includes:
  - ○ Review of functional requirements specification

  - Identification of threats and vulnerabilities
  - Risk identification, analysis and evaluation
  - Recommendations of appropriate security controls

Control gates:

The approving authority for this phase is the steering committee. Recommended control validations for this phase include:

- A TRA report that is approved by steering committee. This is the main deliverable for this phase. It will be used extensively to develop the security requirements, controls and design of the system.
- A series of checks to ensure all high-level security requirements have been included or expressed as a set of security controls in the TRA report.
- Evaluation to determine whether the project is sufficiently supported with the security resources currently available or with resources projected to be available in the desired timeframe.

Key milestone:

The TRA report is the key milestone that needs to be approved by the project steering committee prior to the submission of vendor requirements.

Acquisition

The acquisition phase of the development lifecycle is concerned primarily with the identification of security requirements, evaluation of proposed security controls, and reviewing and finalizing security design prior to acquiring or developing the system.

Security processes:

- Vendor security requirements
- Vendor security evaluation

Activities under this process include:

- Determination of security design objectives and specifications for vendors. Security requirements are clearly articulated, and their purpose and objectives are clearly stated, so that vendors are able to provide adequate measures or controls to meet the requirements.
- Evaluation and assessment of the adequacy of proposed security controls of submitted proposals in meeting requirements for the vendor. The activity includes documentation review, proposal evaluation and clarifications, and assessment of security controls proposed. Recommendations are incorporated into the vendor evaluation report.

Control gates:

The objective of control gates in this phase is to match the security requirements expressed against the security functionality defined by the vendors. All security controls should be included in vendor proposals. The approving authority of the control gate is the steering committee. Recommended control validations for this phase include:

- All the agreed-upon security controls are included in the vendor proposal.
- Vendor's planned activities and outcomes are compliant with organization security policy and procedures.
- Key stakeholders formally accept the risks based on the vendor proposal.

Design and Development

The design and development phase begins after the vendor has been awarded. As part of the design of the system, a critical security design review is conducted to check that the system architecture is secured, and appropriate security controls are incorporated into the design of the system.

Security processes:

- Critical security design review

Activities under this process include:

- Security review of system architecture. The systems architecture is broken down into smaller components and its inner workings are documented to identify trust boundaries, information entry and exit points, and data flows.
- Review of security controls put in place as part of the system design. The activity includes a series of documentation reviews of security controls proposed in the system design, assessment of the design's effectiveness and recommendations. Security controls are justified and documented based on the TRA and security requirements.

Control gates:

The objective of control gates in this phase is to match the security requirements expressed to the security functionality defined by the vendors. All security controls should be included in the vendor proposal. The approving authority of the control gate is the steering committee. Recommended control validations for this phase include:

- The system design is consistent with the UCM's enterprise architecture, including the security components of UCM's architecture.
- The system design addresses the agreed-upon security requirements.
- The TRA report reflects the updated risks after consideration of the security architecture and security controls that have been put in place.

Key milestone:

An updated TRA, including updated risk, assessments and recommendations must be approved by the steering committee before the system can proceed for implementation.

Implementation/Assessment

The implementation/assessment phase begins after the architecture design of the system has been approved. As the system is being implemented, security source code reviews and application testing are conducted to ensure that security has been properly built using a bottom-up approach. A final round of security source code review and application testing is an integral part of acceptance testing, and the system is tested against a set of security test cases.

Security process:

- Application security testing
- System security acceptance testing
- Penetration testing

Activities under this process include:

- Source code review – a review of code in search of security issues due to insecure coding practices or malicious intent or coding errors.
- Application security testing – a process in which each part of the system is isolated to demonstrate that the individual parts are correct.
- System security acceptance testing – testing of the security requirements and controls that have been approved as part of the systems design and to ensure they are acceptable for deployment.
- Penetration testing – the practice of testing the system, network or web application to find vulnerabilities that an attacker could exploit.

Control gates:

In the implementation phase, the system is built and tested. The key stakeholders rely on the outcome of security tests to assess whether the security controls put in place are effective. The approving authority for this control gate is the steering committee.

Recommended control validations for this phase include:

- The security controls defined by the agreed-upon requirements are implemented in the system correctly.
- The mitigation actions arising from source code review reports, security test reports and penetration test reports are addressed, the level of risk accepted and formally approved by the steering committee.
- Users are adequately trained in the security components of the systems.

Key milestone:

System security acceptance testing and penetration testing are performed and results from both tests, including mitigation actions, are reported to the steering committee and approved prior to the commissioning of the system. All project documentation (outputs from current and previous SBD phases) is handed over and accepted by the operations team (e.g., systems administrator) prior to entering the operations/maintenance phase.

Operations/Maintenance

In this phase, systems are checked to ensure they are in place and operating. Enhancements and/or modifications to the system, from a software and hardware perspective, are developed and tested.

Security process:

Audit and continuous monitoring

Activities under this process include:

- Performance of regular general and technical security controls reviews to determine if the security controls in place continue to be effective over time.
- Performance of proper change management to prevent unintended consequences to the security baseline and to reduce the security risks posed by changes to the systems.
- Performance of proper configuration management to ensure that the security baseline of the system remains effective.
- Performance of continuous monitoring such as vulnerability assessments to validate system security.

Control gates:

In this phase, while using the system, we reassess its status based on user feedback, technology changes, policy changes, new threats and vulnerabilities, and other business-related issues. The approving authority for this control gate is the system owners. Recommended control validations for this phase include:

- Validation of security reviews to ensure that built-in controls remain effective, and reporting of the results to the steering committee.
- Validation of security assessment and reviews reports to ensure that systems and environmental changes are addressed.
- Regular review of TRA reports and risk register to ensure that risks remain valid and are continually addressed.

Disposal

This final phase encompasses the disposal of a system and closure of related contracts. Information and system disposal will be addressed explicitly in this phase. The process and activities in this phase ensure the orderly termination of the system while preserving vital information about the system so that the relevant information may be reactivated, migrated or archived in accordance with regulations and policies.

Security process:

The process addresses the proper disposal of the information and the application in a manner that prevents any possibility of unauthorized leakage of sensitive data. This also includes the proper preservation and archiving of data processed by the system in accordance with the organization’s security requirements.

Activities under this process include:

- Handling of information as per the disposal plan. Specific archival methods are selected that facilitate information retrieval in the future as per policies governed by both parties.
- Verification that the disposal of the application complies with license agreements.

Control gates:

In the disposal phase, the key concern is that the system be terminated in an orderly manner, and that vital information about the system is preserved according to applicable records management regulations and policies for future access. All media is accorded correct sanitization methods before the application is finally disposed according to policy. The approving authority of this gate is the system owners.

Recommended control validations for this phase include:

- Validation that information in the system has been correctly preserved and accorded with the appropriate security classification.
- Validation that media sanitization records have been properly recorded and filed.
- Validation that records have been disposed by comparing to the actual application inventory.

Annexure – Roles and Responsibilities

Roles and Responsibilities

Steering committee – The steering committee provides project leadership to ensure the successful delivery of the project and is accountable for approval of key security deliverables and milestones.

System owner – The system owner is responsible for the system and its operations and maintenance.

Security officer / consultant – The security officer/consultant is the subject matter expert on all security tasks.

System administrator – The system administrator is responsible for the day-to-day operations of the commissioned system.

User – The system user represents the users who will interact with the system, typically through an interface, to extract some functional benefit.