

Role of System Readiness Level (SRL) in Integration, Interoperability, and Standardization for System Security.

Padma Lochan Pradhan

Dept of Computer Engineering, MIT Academy of Engineering, Alandi, Pune, PIN-412105, India;
padma.pradhan@mitaoe.ac.in.

Abstract: This abstract summarizes the application of System Readiness Level (SRL) as a governing metric for evaluating the maturity of System Security through the lenses of integration, interoperability, and standardization. In the development of complex, multi-domain systems, individual component maturity (Technology Readiness Level) often fails to account for the security vulnerabilities that emerge during subsystem convergence. This paper/discussion examines the System Readiness Level (SRL)—a composite index derived from the interaction between Technology Readiness Levels (TRL) and Integration Readiness Levels (IRL)—as a critical framework for System Security. By quantifying the maturity of secure interfaces, SRL provides a systematic method for ensuring that security controls are not only technically sound but also integrated into the system architecture without creating new attack vectors. Furthermore, the framework enhances interoperability by standardizing the criteria for secure data exchange across disparate platforms. Through the adoption of SRL, organizations can transition from fragmented security assessments to a standardized, holistic maturity model, ultimately reducing acquisition risk and ensuring that the final system is operationally ready to defend against evolving cyber threats.

Keywords: Technology Readiness Level, The System Readiness Level, Integration Readiness Level, System of Systems, risk mitigation, standardization, and interoperability

1. Introduction

The System Readiness Level (SRL) is an advanced systems engineering metric used to evaluate the overall maturity and operational readiness of complex, integrated systems. In the domain of system security, SRL plays a pivotal role by bridging the gap between individually secure components and a fully functional, secure system architecture. Traditional evaluation models [1], such as the Technology Readiness Level (TRL), primarily focus on the maturity of standalone technologies. However, modern security environments—characterized by distributed, interconnected, and heterogeneous systems—require a more comprehensive assessment approach that considers not only [2] component performance but also their interactions.

To address this limitation, SRL integrates TRL with Integration Readiness Level (IRL), enabling the evaluation of both component maturity and interface compatibility. This combined perspective is crucial because many security vulnerabilities emerge not from individual components but from poorly tested [3] integrations, incompatible interfaces, or weak data exchange mechanisms (Table 1).

Table 1. SRL and System Security: Integrated Perspective

Dimension	SRL Contribution to Security
Integration	Ensures secure component interaction and minimizes interface vulnerabilities
Interoperability	Enables trusted, policy-aligned communication across systems
Standardization	Enforces consistent security frameworks and compliance

In contemporary digital infrastructures—such as cloud computing platforms, cyber-physical systems,

and multi-agency security networks—systems must operate seamlessly across organizational and technological boundaries [4]. Ensuring secure integration, reliable interoperability, and adherence to standardized protocols is therefore essential. SRL provides a quantitative and systematic framework to assess these aspects, helping organizations determine whether a system is truly ready for deployment in real-world, security-critical environments [5]. By moving beyond isolated component validation, SRL 1.1 supports a holistic, system-level approach to security, enabling stakeholders to identify hidden risks, improve coordination among subsystems, and achieve a higher level of confidence in the system's overall security posture [6].

1.1. Concept

To understand how System Readiness Level (SRL) governs security, it is essential to look at the "building blocks" that form the metric. These concepts shift the [7] focus from individual tools to the connective tissue of a secure system (Table 2).

Table 2. SRL and System Security: Integrated Perspective

Level	Description	Focus Area
TRL 1	Basic principles observed	Scientific research
TRL 2	Technology concept formulated	Feasibility study
TRL 3	Experimental proof of concept	Lab validation
TRL 4	Component validation in lab	Prototype testing
TRL 5	Component validation in relevant environment	Integration readiness
TRL 6	System/subsystem model demonstrated	Pre-deployment
TRL 7	System prototype demonstrated in operational environment	Field testing
TRL 8	Actual system completed and qualified	Certification
TRL 9	Actual system proven through successful operation	Full deployment

1.1.1. The SRL Matrix (The Calculation Foundation):

SRL is not a subjective guess; it is a mathematical composite. It is calculated by multiplying two key matrices:

- The TRL Vector: A list of the maturity levels of every individual component (e.g., Firewalls, IDS, Databases).
- The IRL Matrix: A grid that maps how every component connects to every other component.
- Security Impact: If you have a high-tech AI threat detector (TRL 9) but its connection to the automated block-list is untested (IRL 1), the resulting SRL will be very low, signalling a massive security risk.

1.1.2. Integration Readiness Level (IRL):

This is the most critical sub-concept for security interoperability. It defines seven levels of how well two systems "shake hands."

- Lower Levels (1–3): Focus on defining the interface and checking if data can physically move.
- Higher Levels (5–7): Focus on whether the integration is "mission-ready," meaning it can handle encrypted traffic, resist injection attacks, and maintain performance under load.

1.1.3. Semantic Interoperability:

In security, it isn't enough for data to move; it must be understood [8].

- ✓ Concept: A key part of readiness is ensuring that "System A" defines a "Critical Threat" the same way "System B" does.
- ✓ Standardization Link: This concept drives the adoption of standards like STIX/TAXII (for threat intelligence) so that interoperability doesn't fail due to translation errors.

1.1.4. System-of-Systems (SoS) Context:

Modern security doesn't exist in a vacuum. SRL views security as a "System-of-Systems" where independent platforms (Cloud, On-prem, Mobile) must unite.

- The Concept: Readiness isn't reached until the entire ecosystem can maintain a unified security posture.
- Governance: This enables leadership to identify which "legacy system" is hindering the organization's overall security readiness.

1.1.5. The "Maturity Gap":

This is the difference between the most mature component and the least mature integration. Security Relevance: In a chain, the weakest link is the security level. SRL highlights these gaps, forcing architects to standardize the "weakest" interfaces to bring the whole system up to a deployable level [9].

2. Background

The background of System Readiness Level (SRL) is rooted in the evolution of complex systems engineering, moving from a focus on individual tools to a holistic view of integrated ecosystems (Table 3). It was developed to solve a critical problem: mature components often fail when connected, creating unforeseen security and operational risks [14].

Table 3. 5E for Mission Readiness Level (MRL)

Key Term	Secure System Role
TRL	Evaluates the "Tool"
IRL	Evaluates the "Connection"
SRL	Evaluates the "Mission Capability"
SoRL	Emphasizes the Sustainability and Innovation (ESG) Society is accepting the New Technology
ORL	Evaluates the "Mission Maintenance-ORL (Operational Readiness Level) for Sustainability

2.1. Historical Context & Origins

- a. The TRL Foundation: In the 1970s and 80s, NASA developed the Technology Readiness Level (TRL) scale to assess the maturity of individual technologies. While effective for parts, it couldn't measure how those parts functioned as a complete system.
- b. The DoD Shift: By 1999, the US Department of Defense (DoD) adopted TRLs but quickly realized that for modern military acquisitions [15]—like the Littoral Combat Ship—the primary risk was not the technology itself, but the integration of diverse, multi-vendor technologies into a secure "system-of-systems" (SoS) called a meta system.
- c. Introduction of SRL: To address this gap, researchers (notably Sauser & Co. in 2006) proposed the SRL index. It introduced the Integration Readiness Level (IRL) as a middle layer to mathematically bridge component maturity (TRL) with overall system readiness.

2.2. Evolutionary Drivers in Security

- ✓ Increasing Complexity: As security moved from isolated 1940s-era mainframes to interconnected global networks, the "perimeter" disappeared. Readiness metrics evolved to ensure that interoperable systems didn't inadvertently open doors to cyberattacks through immature interfaces [16].
- ✓ Standardization Needs: The rise of diverse IoT and cloud platforms in the 2000s necessitated standardized protocols like TCP/IP and STIX/TAXII. SRL provided the framework to measure how "ready" a system was to adopt these standards without breaking existing security controls [17].
- ✓ From Technical to Operational: Modern assessments have expanded beyond just "does it work?" to include Societal Readiness (SRL) and Operational Readiness (ORL), ensuring that security systems are not just technically sound but also fit for human and regulatory environments.

2.3. Historical Evolution:

The history of System Readiness Level (SRL) is a journey from measuring "parts" to measuring "wholes." It emerged to solve the "integration trap"—where perfect components failed once connected (Table 4).

Table 4. Historical Evolution

Era	Key Metric	Focus
1970s	TRL	Does this specific gadget work?
2006	IRL + TRL = SRL	Do these gadgets work <i>together</i> ?
Today	SRL for Security	Is the entire interconnected network <i>secure and standardized</i> ?

2.3.1. The NASA Foundation (1970s–1980s):

The concept began with the Technology Readiness Level (TRL), created by Stan Sadin at NASA. Its purpose was to provide a common scale (1–9) for the maturity of space flight hardware. However, TRL only looked at individual pieces of technology in isolation [18].

2.3.2. The DoD Integration Crisis (1990s–Early 2000s):

As the U.S. Department of Defense (DoD) began building "System-of-Systems" (like integrated missile defense), they realized that components with a TRL of 9 (flight-proven) were failing when integrated. The failure wasn't the tool; it was the connection.

2.3.3. The Sauser Model (2006):

Researchers at the Stevens Institute of Technology, led by Brian Sauser, formally proposed the SRL index. They introduced a critical missing link: the Integration Readiness Level (IRL).

The Breakthrough: They proved that system readiness is a mathematical product of how mature the parts are (TRL) and how mature the interfaces between them are (IRL) [19].

2.3.4. Modern Evolution: Security & Standardization (2010s–Present):

In the last decade, the history of SRL has shifted toward Cybersecurity and Standardization:

- Interoperability Mandates: Governments began using SRL to ensure that multi-vendor security systems could talk to each other without creating "security gaps."
- Expanding the Scale: The model has evolved into "System-of-Systems" readiness, focusing on how cloud, edge, and on-premise security fabrics work as a standardized unit.

3. Literature Survey

A literature survey on System Readiness Level (SRL) reveals its evolution from a NASA-centric component tool to a complex, multi-dimensional framework used by major defense and aerospace organizations. The literature highlights a shift toward valuing integrative security over individual tool maturity [20].

3.1. The Foundational Shift (TRL to SRL):

Early research focused exclusively on Technology Readiness Levels (TRL) to assess component maturity. However, seminal papers from the Stevens Institute of Technology (Sauser et al., 2006) introduced [21] the concept of the Integration Readiness Level (IRL), arguing that a system's true readiness is a mathematical function of both its parts and its connections.

Core Literature: From TRL to SRL: The Concept of System Readiness Levels provides the primary methodology for this transition.

3.2. Interoperability and System Security:

Recent surveys, such as those published in IEEE Xplore and MDPI, address the role of SRL in securing "System-of-Systems" (SoS) like IoT and cloud environments [22].

- Semantic Gaps: Researchers argue that a lack of "semantic interoperability"—where different tools don't share a common language—is a major architectural flaw that prevents effective security deployment.
- Privacy & Trust: Literature in ResearchGate explores how SRL-based assessments must incorporate privacy frameworks (like GDPR) to be truly "mission-ready" in interoperable systems.

3.3. Critical Critique and Limitations:

A significant body of academic work critiques the mathematical validity of the standard SRL model emphasised by Sauer in 2006.

- a) The Ordinal Data Error: Critics from the Naval Postgraduate School and IEEE argue that because TRL and IRL are "ordinal" (ranked) rather than "cardinal" (measurable) numbers, standard matrix multiplication is mathematically flawed and can be misleading, as cited by Kujawski in 2013.
- b) Subjectivity: To counter human bias in manual assessments, more recent literature proposes using Monte-Carlo simulations and Markov Chain-based methods to reduce time lags and improve accuracy.

3.4. Application Case Studies:

Defense Programs: The methodology is widely cited in reviews of the Littoral Combat Ship (LCS) mission modules to monitor development status across complex, multi-vendor platforms [23].

Energy Sector: Pilot applications have also been documented by the National Energy Technology Laboratory (NETL) to manage advanced fossil energy technology transitions.

4. Research Motivation

The research motivation for applying System Readiness Level (SRL) to system security stems from the "Integration Paradox": as individual security tools become more advanced, the overall system often becomes more [24] vulnerable due to complex, unverified connections (Table 5).

Table 5. Summary of Motivation Drivers

Driver	From (Legacy)	To (SRL-Driven)
Focus	Tool Performance	System Harmony
Data	Subjective Opinions	Mathematical Indices
Risk	Component Failure	Integration Vulnerability

4.1. The Failure of Component-Centric Security:

Traditional security assessments focus on Technology Readiness (TRL)—asking, "Is this firewall good?" However, history shows that security breaches rarely occur because a mature tool failed; they occur at the seams where tools connect. The motivation is to shift focus [25] from the "parts" to the "links".

4.2. Managing "System-of-Systems" Complexity:

Modern security is no longer a single software package; it is a sprawling ecosystem of Cloud, IoT, and Legacy on-premise systems.

- The Problem: There was no mathematical way to prove these diverse systems were "ready" to work together.
- The Motivation: SRL provides a unified metric to quantify the maturity of these massive, multi-vendor environments, ensuring that "Interoperability" doesn't come at the cost of "Security."

4.3. Reducing the "Security Maturity Gap":

Organizations often waste budget by buying "Level 9" (cutting-edge) AI tools but connecting them via "Level 1" (untested/manual) processes.

- ✓ The Problem: This creates a false sense of security.
- ✓ The Motivation: Research into SRL aims to identify these gaps, forcing a Standardization of interfaces so that the weakest link doesn't compromise the strongest component.

4.4. Moving from Subjective to Objective Governance:

Before SRL, "readiness" was often a subjective opinion shared in meetings [26].

The Motivation: To create a rigorous, evidence-based framework that allows security leaders to say with mathematical certainty: "We are at Level 7 readiness; we cannot deploy until the integration between System A and B is standardized."

5. Research Gaps

While the System Readiness Level (SRL) framework is robust, its application to modern System Security reveals several critical research gaps [27] that academia and industry are still struggling to bridge (Table 6):

- 1) The "Static vs. Dynamic" Gap: Most SRL calculations are static snapshots taken during development. However, security readiness changes instantly when a new zero-day vulnerability is discovered or a system configuration is altered.

The Gap: There is a lack of Real-time SRL models that can dynamically update a readiness score based on live threat intelligence or continuous monitoring data.

- 2). The Mathematical "Ordinal Data" Fallacy: As noted in the literature, TRL and IRL are ordinal ranks (1–9). Multiplying them to get a "mean" SRL is mathematically controversial.

The Gap: Research has yet to produce a universally accepted, mathematically rigorous alternative (such as Fuzzy Logic or Bayesian Networks) that can replace simple matrix multiplication without becoming too complex for non-experts to use.

- 3). Oversight of "Semantic Interoperability": Current IRLs (Integration Readiness Levels) focus heavily on whether data *can* move (technical integration) rather than whether it is *understood* (semantic integration).

The Gap: In security, if a SIEM interprets a "Critical" alert from a firewall differently than an "Emergency" alert from a cloud monitor, the system is not truly ready. There is a need for Semantic-aware IRLs specifically for security ontologies.

- 4). Human and Organizational Readiness [28] (The "Soft" Factors): SRL is heavily tech-centric. However, a security system is only as ready as the team operating it.

The Gap: There is no standardized way to integrate Human Readiness Levels (HRL) into the SRL score. A system might be technically a Level 9, but if the security operations center (SOC) isn't trained to use it, the "true" operational readiness is much lower.

- 5). Lack of Standardized Benchmarks for AI: With the rise of AI-driven security (AIOps), traditional TRL/IRL scales struggle to measure the "maturity" of a self-learning model that changes its behavior over time.

The Gap: Defining Non-deterministic Readiness—how do you measure the integration maturity of a system that learns and evolves after it has been deployed?

Table 6. Summary of Research Gaps

Gap Area	Current State	Research Need
Temporal	Static snapshots	Dynamic/Continuous SRL
Logic	Questionable multiplication	Non-linear Math Models
Data	Focus on connectivity	Semantic/Contextual Maturity
Personnel	Focus on hardware/software	Human-in-the-loop Metrics

6. Problem Analysis

In the context of System Security, problem analysis focuses on why traditional readiness assessments fail to prevent breaches in integrated environments. The core problem is that security is emergent; it doesn't exist in the individual tools but arises (or fails) from how they interact (Figure 1).

6.1. The Decomposition [29] Fallacy:

A primary issue in security engineering is the assumption that if individual components are "hardened" (TRL 9), the resulting system will be secure (Table 7).

Table 7. Problem Analysis

Summary of the Problem Landscape: Problem Area	Symptom	Analysis Root Cause
Connectivity	Data leaks between tools	High TRL, but Low IRL (immature connections).
Governance	Deployment delays/failures	Lack of a unified metric (SRL) to track progress.
Compliance	Standards aren't followed	The system wasn't "Ready" for the standard.

- a) The Problem: System security often degrades during integration. For example, connecting a mature cloud storage service to a mature on-premise app often creates an insecure "seam" (e.g., misconfigured APIs or credential leakage).
- b) SRL Role: Without an SRL framework, managers lack a metric to see that while TRL is high, the

Integration Readiness (IRL) is low, leading to a false sense of security.

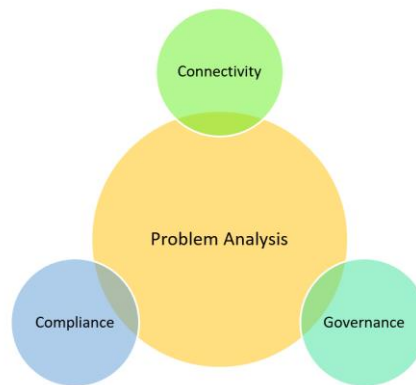


Figure 1. Problem Analysis

6.2. Standardization vs. Complexity Trade-off:

Standardization (e.g., using NIST or ISO frameworks) is meant to simplify systems, but implementing these standards across legacy and modern platforms is a complex task [30].

- a. The Problem: Organizations often "force" a standard onto a system that isn't ready for it. This leads to "clunky" security controls that users bypass because they hinder performance.
- b. The Analysis: The problem isn't the standard itself, but the lack of a readiness assessment to determine if the system's architecture can support the standard without breaking critical workflows.

6.3. Interoperability Blind Spots:

Interoperability requires systems to share sensitive data.

- ✓ The Problem: The more "interoperable" a system is, the larger its attack surface. Traditional analysis focuses on keeping people *out*, but interoperability requires letting other systems *in* [31–34].
- ✓ The Risk: If one system in an interoperable "System-of-Systems" (SoS) has a low readiness level, it becomes a "Trojan Horse" that can compromise more mature systems through trusted connections.

6.4. Qualitative vs. Quantitative Gap:

Most security readiness is currently assessed through qualitative checklists (e.g., "Yes, we have a firewall").

- a) The Problem: Checklists don't capture the mathematical risk of complex interactions.
- b) The Analysis: There is a critical need for a quantitative index (like the SRL score) that can aggregate thousands of technical checkpoints into a single, actionable maturity value for decision-makers.

7. Objective of this Research

The primary objective of this research is to develop a quantitative framework that uses System Readiness Level (SRL) to ensure that security is maintained throughout the processes of integration, interoperability, and standardization. Specifically, the research aims to achieve the following:

- 1). Quantify the "Security Gap": To move beyond subjective checklists by creating a mathematical model (combining TRL and IRL) that identifies exactly where a system's security posture is weakened by immature connections, even when individual tools are advanced.
- 2). Standardize Interoperability Protocols: To define the Integration Readiness Level (IRL) criteria required for secure data exchange. This ensures that when two disparate systems "shake hands," they do so using standardized, hardened protocols that minimize the attack surface.
- 3). Establish a Maturity-Based Deployment Gate: To create a decision-making benchmark that prevents the premature deployment of "System-of-Systems (Meta System)." The objective is to ensure that a system is only moved to an operational environment when it reaches a specific, verified SRL score (e.g., SRL 7 or higher) [32].

- 4). Optimize Resource Allocation: To provide a roadmap for security architects to identify the "weakest link" in an architecture. This allows organizations to stop over-investing in mature components (high TRL) and start investing in the standardization of interfaces (low IRL) where the actual vulnerabilities lie.
- 5). Validate the "System-of-Systems (SoS-SRL)" Security Posture: To prove that security is an emergent property of a well-integrated system. The research seeks to demonstrate that high system readiness directly correlates with higher resilience against lateral movement and cross-platform cyber threats.

Summary Goal: To transform "System Readiness" from a generic engineering term into a rigorous security metric that guarantees a system is technically mature, interoperable, and compliant with global standards before it goes live.

8. Research Methodology

To achieve the research objectives, the methodology follows a mixed-methods approach, combining mathematical modelling (quantitative) with expert validation and architectural simulation (qualitative) (Table 8).

Table 8. Phase-wise Development

Phase	Activity	Output
Phase 1	System Mapping	Connectivity Diagram
Phase 2	TRL/IRL Assignment	Readiness Matrices
Phase 3	Index Calculation	Final SRL Score
Phase 4	Validation	Case Study Report

- 1) Architectural Modelling (System-of-Systems Definition): The first step is to define the boundary of the security system.
 - a) Decomposition: Break the system down into its constituent security technologies (e.g., Firewall, IAM, SIEM).
 - b) Identification: Map every interface where these components exchange data.
 - c) Contextualization: Define the security standards (e.g., NIST 800-53) that the system must adhere to for "Standardization."
- 2) Data Collection & Metric Assignment: Assigning values to the building blocks of the SRL:
 - ✓ TRL Assessment: Use technical documentation and lab testing to assign a Technology Readiness Level (1–9) to each component.
 - ✓ IRL Mapping: Use API documentation and protocol analysis to assign an Integration Readiness Level (1–7) to every link between components.
 - ✓ Expert Delphi Method: Use a panel of security experts to refine these scores, ensuring the levels accurately reflect the "security robustness" of the integration.
- 3). Mathematical Computation (The SRL Index): The research applies a Matrix Calculus approach to generate a composite score:
 - a. Normalization: Convert TRL and IRL scores into decimal values (0.0 to 1.0).
 - b. Matrix Multiplication: Multiply the TRL vector by the IRL matrix to find the "System Maturity" mean.
 - c. Sensitivity Analysis: Perform a "What-if" analysis to see how a failure in one standardized protocol (lowering an IRL) impacts the total system security score.
- 4). Simulation & Validation (Testbed): To prove the framework works, the methodology includes a Simulation Phase:
 - a) Scenario Injection: Introduce "threats" (e.g., lateral movement) into a simulated environment with a High SRL vs. a Low SRL.
 - b) Benchmark Comparison: Compare the SRL results against traditional security checklists to see which method better predicts actual system failures.
- 5) Iterative Refinement: Based on simulation results, the IRL definitions are refined to include more

"security-specific" criteria (like encryption maturity or handshake speed), ensuring the model is tailored for the high-stakes environment of system security.

8.1. System Design

In the context of System Security, the System Design phase is the stage where the mathematical theory of SRL is translated into a technical blueprint (Table 9). The goal is to design an architecture that maximizes connectivity while preserving the security perimeter (Figure 2).

Table 9. Scope of System Design

Design Feature	Impact on Integration	Impact on Security
Service Mesh	Simplifies service-to-service links	Provides mTLS and traffic visibility
API Gateway	Standardizes external access	Centralizes threat filtering
Micro-segmentation	Isolates system modules	Prevents lateral movement if a link fails

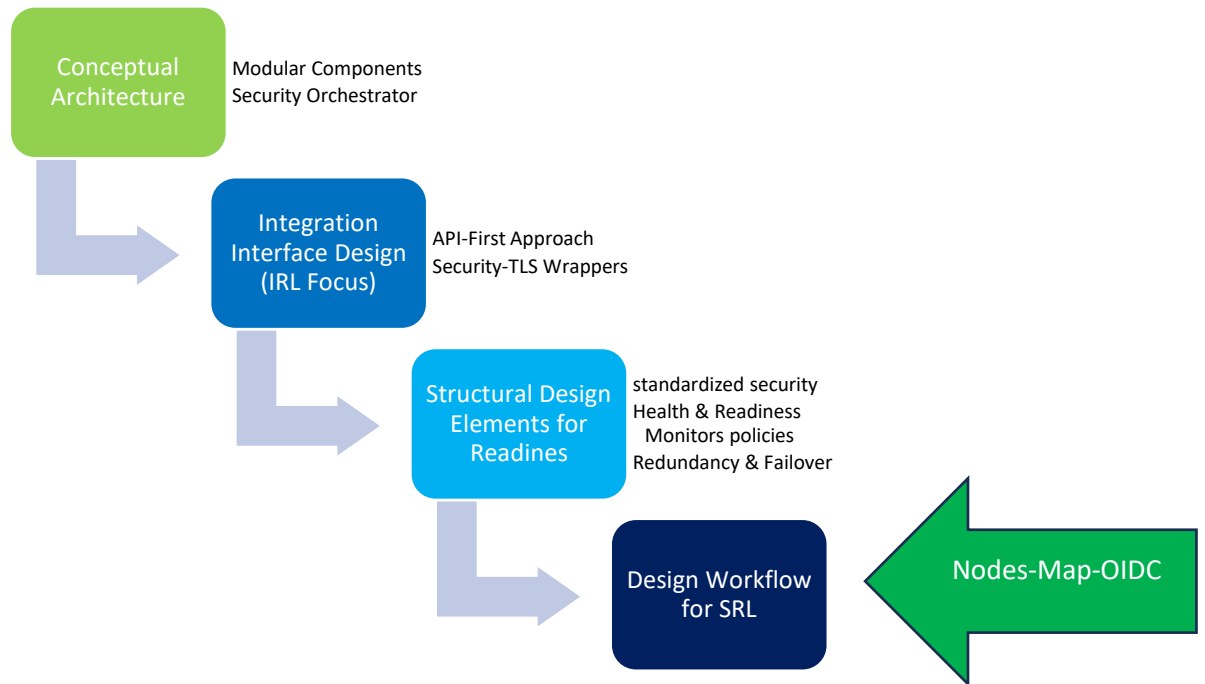


Figure 2. System Design

8.1.1. Conceptual Architecture:

The "Security Fabric": Unlike traditional siloed designs, an SRL-driven design focuses on a decentralized security fabric.

- ✓ Modular Components: Each security tool (IAM, EDR, Firewall) is designed as a standalone module with a high TRL.
- ✓ Middleware/Orchestration Layer: A central "Security Orchestrator" acts as the integration hub. By designing a single, standardized point of contact, you raise the IRL across the entire system because you are managing one robust interface rather than dozens of "point-to-point"(POP) connections.

8.1.2. The Integration Interface Design (IRL Focus):

To ensure high readiness, every connection between systems must follow a Standardized Interface Pattern:

- a) API-First Approach: All components must communicate via documented, versioned APIs.
- b) Security Wrappers: Every integration point is designed with a "wrapper" that handles encryption (TLS 1.3), authentication (Mutual TLS), and logging before data reaches the core component.
- c) Semantic Layer: The design includes a "Data Transformer" that ensures different tools (e.g., a Cisco firewall and a Microsoft SIEM) use a Common Information Model (CIM).

8.1.3. Structural Design Elements for Readiness:

To move a system toward SRL 7+ (Demonstrated in Operational Environment), the design must incorporate:

- Policy Decision Points (PDP): A centralized engine that pushes standardized security policies to all integrated modules simultaneously.
- Health & Readiness Monitors: Built-in telemetry that reports the "Live SRL" of the system, alerting admins if a component's integration becomes unstable or fails a handshake.
- Redundancy & Failover: Designing "Graceful Degradation" so that if one interoperable link fails (Low IRL), the rest of the system remains at a high security readiness level.

8.1.4. Design Workflow for SRL

- Identify Nodes: List all hardware/software entities.
- Define Links: Map the data flow between nodes.
- Apply Standards: Select protocols (e.g., OIDC for identity, STIX for threat intelligence).
- Calculate Design-SRL: Use the TRL/IRL matrix on the blueprint to predict the maturity of the finished system.

8.2. System Architecture

The system architecture for a secure environment uses System Readiness Level (SRL) to transition from a collection of isolated "hardened" tools to a unified, mission-ready ecosystem. It focuses on the connective tissue between components, ensuring that interoperability does not come at the cost of security (Figure 3).

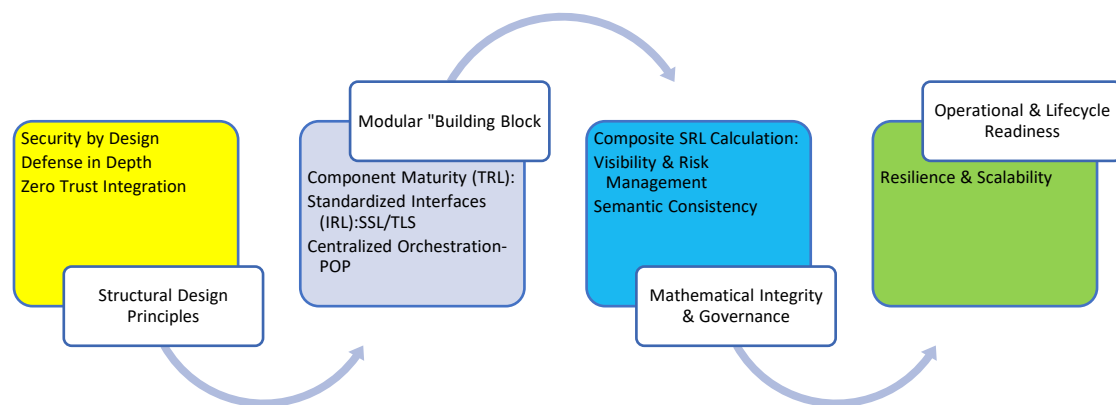


Figure 3. SRL- System Architecture

8.2.1. Structural Design Principles:

A secure architecture built on SRL principles adheres to foundational engineering standards such as ISO 27001 Control 8.27, which mandates integrating security into all layers from inception.

- Security by Design: Security is a primary focus from the start, embedded in the architecture rather than added later.
- Defense in Depth: The architecture employs multiple layers of controls (encryption, access management, firewalls) so that a failure in one does not compromise the whole.
- Zero Trust Integration: It adopts a "never trust, always verify" approach, assuming the network is already compromised and requiring verification for every access request.

8.2.2. The Modular "Building Block" Approach:

To maintain a high SRL, the architecture is designed with modular components that interact through standardized interfaces.

- ✓ Component Maturity (TRL): Individual modules (e.g., identity providers, databases) are selected

based on their independent stability and proven performance in relevant environments.

- ✓ Standardized Interfaces (IRL): Communication between modules is managed through APIs and standardized protocols (SSL/TLS, IPsec). High Integration Readiness Levels (IRL) are achieved when these interfaces are well-defined and validated for data structure and security structure.
- ✓ Centralized Orchestration: A "hub-and-spoke" or API Gateway model is often used to avoid complex point-to-point "spaghetti" connections, simplifying the security perimeter and making the system easier to audit.

8.2.3. Mathematical Integrity & Governance:

The architecture includes built-in mechanisms to measure its own maturity:

- ❖ Composite SRL Calculation: The system maturity is tracked by multiplying the TRL vector by the IRL matrix. This reveals "lagging" components that require engineering attention to prevent them from becoming weak links.
- ❖ Visibility & Risk Management: Tools like Design Structure Matrices (DSM) are used within the architecture to provide a more accurate and evidence-based assessment of readiness.
- ❖ Semantic Consistency: The architecture ensures that all integrated systems use a Common Information Model (CIM), preventing security failures caused by mismatched data interpretations (e.g., a "critical" alert in one system being ignored by another).

8.2.4. Operational & Lifecycle Readiness:

Fail-Secure Design: The system is architected to "fail securely" and maintain critical functionality even during partial disruptions.

Resilience & Scalability: By using modular integration patterns, the architecture allows for independent updates to features without disrupting the operational continuity of the entire security fabric.

8.3. Framework

The System Security Readiness Framework (SSRF) is a structured model designed to bridge the gap between high-level security standards and technical implementation. It uses the System Readiness Level (SRL) as the primary engine (Table 10). to ensure that security is "baked in" during integration (Figure 4).

8.3.1. The Multi-Layered Structure:

The framework is organized into three distinct layers that determine how security maturity is calculated:

- a) Technology Layer (TRL Focus): Assesses the maturity of individual security "nodes" (e.g., encryption modules, biometric sensors). It ensures that the basic tools are functional and resilient against standalone testing.
- b) Integration Layer (IRL Focus): Evaluates the "connective tissue." This is where Standardization occurs. It checks if the data pipelines between nodes use secure protocols (e.g., TLS 1.3) and if they are semantically aligned.
- c) Systemic Layer (SRL Focus): The overarching view. It combines the TRL and IRL scores to provide a single value representing the Interoperability of the entire security fabric.

8.3.2. The Functional Components:

A robust SRL framework for security must include these four functional pillars:

Table 10. System Security Readiness Framework

Pillar	Focus	Goal
Governance	Policies & Standards	Ensure compliance with frameworks like NIST 800-53 or ISO 27001.
Technical	APIs & Protocols	Achieve high IRL through standardized, encrypted interfaces.
Semantic	Data Ontologies	Ensure all systems interpret "threat levels" and "logs" identically.
Operational	Human-in-the-Loop	Validate that security teams can manage the integrated system.

8.3.3. The Framework Lifecycle (The "Readiness Gate"):

The framework acts as a Quality Gate in the development lifecycle:

- Requirement Mapping: Define the desired security posture and necessary standards.
- Maturity Assessment: Periodically calculate the current SRL based on component and link status.
- Gap Identification: Highlight specific interfaces where the IRL is low (e.g., an unencrypted link between a database and an app).
- Remediation: Standardize the weak links to raise the overall system score.
- Deployment Verification: Only deploy when the composite SRL meets the "Mission Ready" threshold (usually SRL 7 or above).

8.3.4. Integration with Global Standards:

The framework doesn't replace existing standards; it operationalizes them. While a standard tells you *what* to do (e.g., "Encrypt data in transit"), the SRL framework measures *how mature* your specific implementation of that standard is across your entire interconnected system.

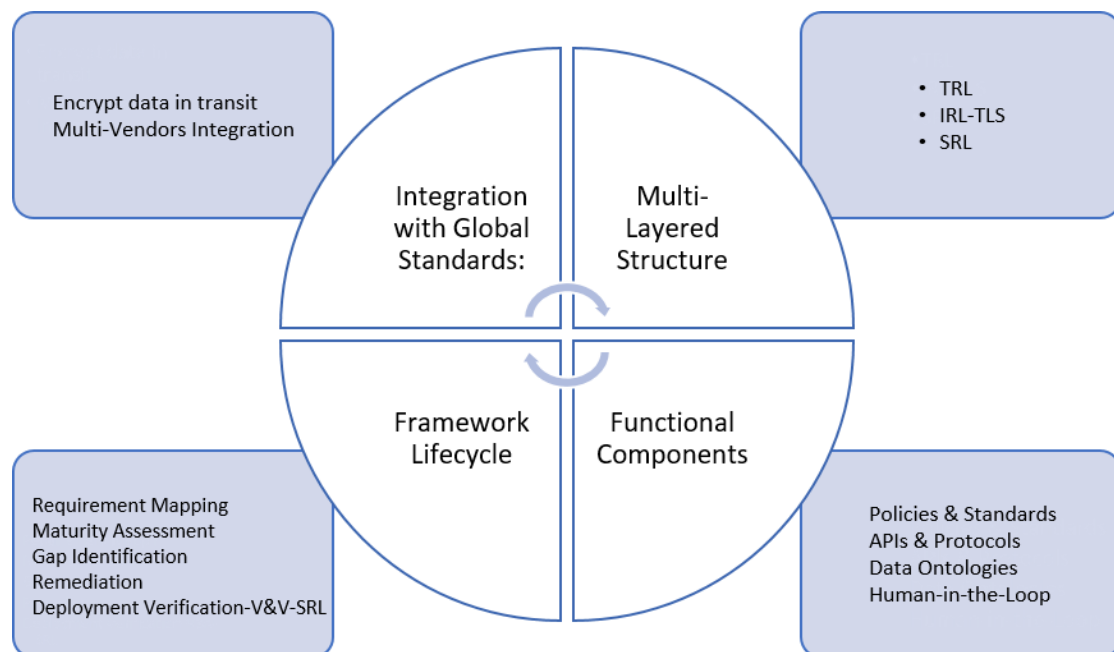


Figure 4. Block diagram of SRL- Framework with Functional Components

8.4. Open-Source Software Activities:

Open-Source Software (OSS) is a critical enabler of achieving high System Readiness Levels (SRLs) by providing transparent, flexible foundations for secure integration, interoperability, and standardization. Role in the SRL Framework: OSS directly influences the maturity of a system by lowering the barriers to entry for advanced security technologies while demanding rigorous governance to manage its inherent risks (Table 11).

- Transparency as a Readiness Bench: The ability to inspect source code allows for independent audits, enabling organizations to verify that a component meets security standards before integration. This transparency supports reaching IRL 4-5 (Validation in a relevant environment) by proving that security claims are technically sound.
- Enabling Standards Adoption: Most leading security standards (like TLS, STIX/TAXII, or OSCAL) have reference implementations in open source. Using these mature implementations can instantly raise a system's Technology Readiness Level (TRL) and ensure that its integration points are pre-standardized.
- Preventing Vendor Lock-in: OSS allows organizations to swap components without rebuilding the entire architecture. This modularity ensures that the SRL remains high even as individual tools are updated or replaced, maintaining long-term architectural stability. Popular Open-Source Security Tools for SRL Assessment: Organizations often use specialized OSS tools to measure

and automate their system's readiness:

Table 11. OSS for SRL Assessment

Tool	Primary Focus	Role in Readiness Assessment
OpenSCAP	Compliance Auditing	Benchmark systems against standardized security policies like NIST.
OpenVAS	Vulnerability Assessment	Provides the technical evidence needed to validate TRL maturity for network nodes.
DefectDojo	Vulnerability Orchestration	Aggregates results from multiple tools into a single view, facilitating SRL scoring.
OWASP Dependency-Check	Supply Chain Security	Identifies vulnerable third-party libraries, protecting the system's "seams" during integration.
OSSEC	Intrusion Detection	Monitors system activity in real-time to maintain SRL 9 (Mission Ready) status.

Risks and Management: While OSS fosters innovation and interoperability, it introduces specific risks that can drag down an SRL score if not managed:

- Supply Chain Vulnerabilities:** A single compromised library can lower the IRL of multiple connections. Continuous monitoring using Software Bill of Materials (SBOM) tools like Syft or Grype is required for high-readiness systems.
- Maintenance Decay:** "Abandoned" open-source projects can lead to a sudden drop in TRL as new threats emerge.
- Licensing Complications:** Incompatible OSS licenses can create legal barriers to integration, effectively preventing a system from reaching full operational readiness.

Final Scannable Scenarios: To move a security system to SRL 7 (Operational Demonstration), it is recommended to use standardized OSS frameworks for the integration layer while maintaining an automated inventory of all dependencies.

8.5. Integration

In the context of System Security, integration is the process of physically and logically connecting disparate security components into a unified architecture. When managed through the System Readiness Level (SRL) (Table 12), integration shifts from a "plug-and-play" activity to a rigorous, risk-managed discipline (Figure 5).

Table 12. Integration as a Standardization

Integration Type	Security Focus	Readiness Impact
Data Integration	Encryption & Integrity	Ensures information isn't tampered with in transit.
Identity Integration	Single Sign-On (SSO)	Prevents fragmented, weak password policies.
Control Integration	Automated Orchestration	Allows one tool to "command" another during a breach.

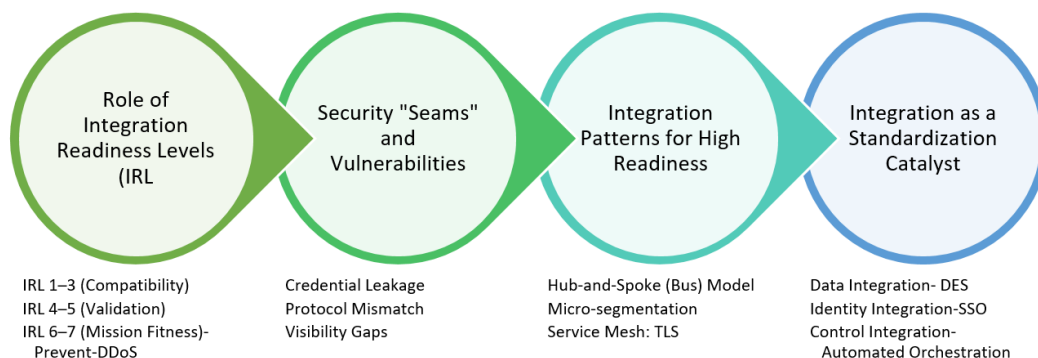


Figure 5. Block Diagram of SRL- System Integration

8.5.1. The Role of Integration Readiness Levels (IRL):

Integration is the primary variable in the SRL equation. We cannot have a high SRL without high

IRLs.

- IRL 1–3 (Compatibility): Focuses on the "handshake." Can the firewall talk to the SIEM? At this stage, the integration is purely technical.
- IRL 4–5 (Validation): Focuses on the "protocol." Is the data being sent over a standardized, encrypted channel? This is where security standards (like TLS 1.3 or IPsec) are enforced.
- IRL 6–7 (Mission Fitness): Focuses on the "performance." Does the integration hold up under a DDoS attack? This level confirms the integration is resilient in a realistic threat environment.

8.5.2. Security "Seams" and Vulnerabilities:

Integration is where most security failures occur. These "seams" are the gaps between two mature (High TRL) products.

- ✓ Credential Leakage: Insecurely passing API keys between integrated services.
- ✓ Protocol Mismatch: One system using an outdated, vulnerable protocol (e.g., SMBv1) to talk to a modern system.
- ✓ Visibility Gaps: Logs from an integrated cloud service failing to reach the central monitoring tool, leaving a "blind spot" in the security posture.

8.5.3. Key Integration Patterns for High Readiness:

To ensure a secure and standardized integration, architects use specific patterns:

- The Hub-and-Spoke (Bus) Model: Using a central Enterprise Service Bus (ESB) or API Gateway. This standardizes security at a single point, raising the IRL for all connected "spokes" simultaneously.
- Micro-segmentation: Designing the integration so that even if one link is compromised, the breach cannot move laterally to other integrated components.
- Service Mesh: In cloud environments, a service mesh handles the "integration" layer (mTLS, retries, and logging) automatically, allowing developers to focus on component logic (TRL).

8.5.4. Integration as a Standardization Catalyst:

Integration forces systems to adopt standards. For a system to reach SRL 7, its integrations must be repeatable and documented. This discourages "bespoke" or "custom-coded" connections, which are difficult to patch and prone to human error, in favour of industry-standard protocols.

8.6. Interoperability

In the context of System Security, interoperability is the ability of different security systems to exchange information and use it to perform automated, coordinated actions (Figure 6). While integration is about the "connection," interoperability (Table 13) is about the "conversation."

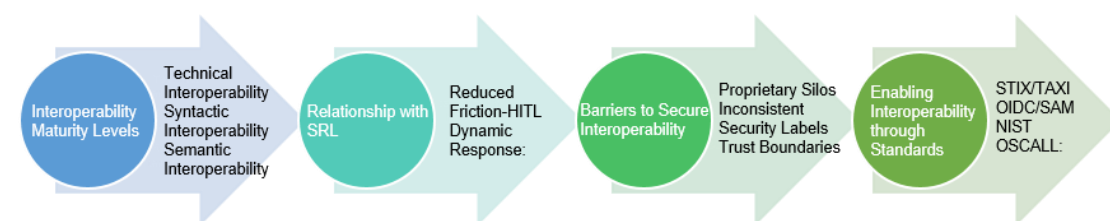


Figure 6. SRL- System Interoperability with its core components

Table 13. Feature of Integration and Interoperability

Feature	Integration	Interoperability
Focus	Making a connection	Making the connection <i>useful</i>
Action	Data moves from A to B	System B acts on data from System A
SRL Impact	Increases IRL	Increases Systemic Capability

8.6.1. The Interoperability Maturity Levels:

Within the SRL framework, interoperability determines how effectively the system functions as a "System-of-Systems" (SoS). It is typically categorized into three levels (Meta Systems) (Ref. Fig.10):

- a) Technical Interoperability: The ability of hardware and software components to physically communicate (e.g., matching bitrates, ports, and cables).
- b) Syntactic Interoperability: Ensuring that data formats are compatible (e.g., both systems can parse the same JSON or XML schema).
- c) Semantic Interoperability: The "Gold Standard" for security. It ensures that the *meaning* of the data is shared. For example, if a Cloud Guardrail flags a "Policy Violation," the on-premise Firewall must understand exactly what that means and block the corresponding traffic.

8.6.2. The Relationship with SRL:

Interoperability acts as a multiplier for System Readiness. A system can have a high SRL only if its components are highly interoperable.

- a. Reduced Friction: High interoperability eliminates the need for manual "human-in-the-loop" data translation, which is often where security delays and errors occur.
- b. Dynamic Response: In a ready system, interoperability allows for Automated Threat Sharing. If one node detects a new malware signature, it interoperates with the rest of the network to update block lists in real-time.

8.6.3. Barriers to Secure Interoperability:

The research identifies several "interoperability traps" that lower the total SRL score:

- a. Proprietary Silos: Vendor-locked systems that use closed protocols. These force the use of "custom wrappers," which are often less secure and harder to update.
- b. Inconsistent Security Labels: Different systems use different severity scales (e.g., 1–10 vs. Low/Med/High). This lack of standardization leads to "Alert Fatigue" or missed critical threats.
- c. Trust Boundaries: The challenge of maintaining a Zero Trust posture while allowing diverse systems to talk to each other.

8.6.4. Enabling Interoperability through Standards:

To achieve "Mission Ready" status (SRL 8-9), systems must use standardized "languages":

1. STIX/TAXII: For sharing cyber threat intelligence.
2. OIDC/SAML: For cross-platform identity interoperability.
3. NIST OSCAL: For standardized, machine-readable security compliance data.

8.7. Correlation

In system security, correlation is the logical link that binds Integration, Interoperability, and Standardization into a single measurable value: the System Readiness Level (SRL). It represents the mathematical and functional relationship where an improvement in one area exponentially increases the maturity of the others (Table 14).

- 1). The Mathematical Correlation (The SRL Formula): The most direct correlation is expressed in the SRL calculation. If we treat a security system as a network of nodes:
 - a) TRL (Technology): The maturity of the individual tools.
 - b) IRL (Integration): The maturity of the links.
 - c) The Result: A high TRL but a low IRL results in a low SRL.
 - d) The Insight: There is a linear correlation between the maturity of the "weakest link" and the overall security readiness of the system. You cannot "buy" your way to a secure system by only purchasing high-TRL tools if your integration (IRL) remains immature.
- 2). Functional Correlation: The Security "Force Multiplier": Correlation describes how Standardization acts as the engine for Interoperability:
 - a. Standardization → Interoperability: By adopting standard protocols (like OAuth 2.0 or HTTPS), the complexity of integration drops. This has a positive correlation with interoperability—the more standardized the interfaces, the more effectively disparate systems can "talk" to one another.
 - b. Interoperability → Security Resilience: When systems are highly interoperable, they can correlate threat data in real-time. For example, a login failure on a VPN (System A) correlated with a strange database query (System B) allows for an automated lockdown that neither system could

trigger alone.

3). The "Maturity Gap" Correlation: Research shows a strong correlation between the complexity of a system and its vulnerability surface:

- ✓ Negative Correlation: As "custom" (non-standardized) integrations increase, the SRL typically decreases because the system becomes harder to patch, audit, and secure.
- ✓ Positive Correlation: As standardization increases, the "Time-to-Readiness" decreases. Standardized systems reach SRL 7 (Operational Demonstration) significantly faster than bespoke architectures because they use pre-validated integration patterns.

Table 14. Summary of Correlations

Relationship	Type	Security Outcome
Standardization & IRL	Direct	Using standards like TLS automatically raises the maturity of the link.
Interoperability & SRL	Exponential	Seamless data exchange makes the <i>entire</i> system smarter than the sum of its parts.
Custom Integration & Risk	Inverse	The more "unique" a connection is, the lower its readiness and the higher its security risk.

8.7.1. Brief Summary: Synthesis of Integration, Interoperability, and Standardization

Integration, interoperability, and standardization together form a unified foundation for secure and mature systems (Figure 7), as reflected by the System Readiness Level (SRL) (Table 15).

- a) Integration ensures that system components are combined into a cohesive architecture, but its effectiveness depends on how securely these components interact.
- b) Interoperability enables seamless and secure communication between heterogeneous systems, ensuring correct data exchange and coordinated functionality.
- c) Standardization provides the common rules, protocols, and security frameworks that govern both integration and interoperability, ensuring consistency and compliance.

8.7.2. Synthesis of Secure Integration, Interoperability, and Correlation (Interpretation of IIC)

Table 15. Conceptual Flow of IICS

Dimension	Role	Security Outcome
Integration	Connect subsystems securely	Prevent insecure entry points
Interoperability	Enable cross-system communication	Ensure trusted data exchange
Correlation	Link and analyze events	Detect coordinated threats
Standardization	Enforce uniform practices	Achieve compliance and resilience

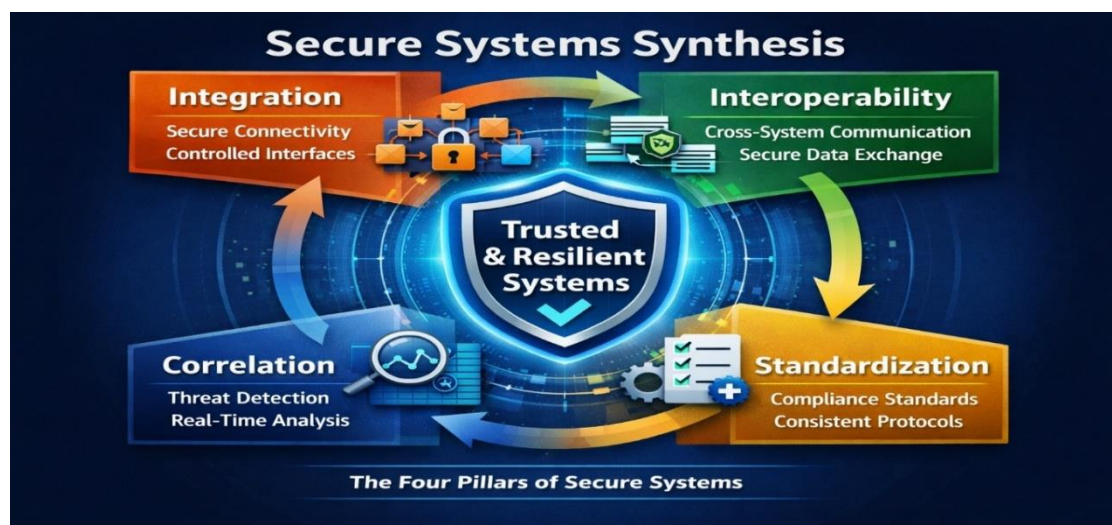


Figure 7. Synthesis of Secure Integration, Interoperability, Correlation, and Standardization

9. CASES

9.1. SoRL (Social Readiness) for ESG (Environmental, Social, and Governance)

Now the Author is taking care and emphasizing the Sustainability and Innovation sub-area, such as Environmental, Social, Governance, and Green Technology, for the betterment of Social, Business, Economics, and Technology to protect the Real-World-Assets at all times and every time for global transformation (Table 16).

Social Readiness Level (SRL-SoRL) assesses how ready a society, market, or community is to accept, adopt, and trust a new technology or system. Integrating SoRL with standard engineering metrics ensures that a sustainable system is not just technically viable but also socially responsible and culturally accepted. SoRL (Social Readiness Level) for Sustainability.

- a. SoRL 1–3 (Identify & Frame): Social and ethical risks are identified; stakeholder groups (e.g., local communities, workers) affected by the green technology are mapped.
- b. SoRL 4–6 (Co-Create & Test): Prototyping involves public feedback; community concerns about land use, toxic materials, or job displacement are actively addressed.
- c. SoRL 7–9 (Adopt & Scale): The sustainable system achieves deep public trust, widespread market adoption, and full alignment with social justice and labor regulations.

Table 16. Integrated Sustainability & Social Readiness Matrix

Phase	Technology Readiness (TRL) (Core Tech)	Integration Readiness (IRL) (Connections)	System Readiness (SRL-Sys) (Total System Maturity)	Social Readiness (SRL-Soc) (Public & Market Trust)
1. Concept	TRL 1–3: Green concepts defined; environmental benefits calculated.	IRL 1–3: Supply chain flows mapped; data protocols designed.	SoRL 0.1–0.3: High-risk phase; tech exists but lacks verified infrastructure.	SoRL 1–3: Social impacts identified; key community stakeholders mapped.
2. Prototype	TRL 4–6: Lab testing of low-carbon or circular materials.	IRL 4–6: Physical testing of green parts joining standard parts.	SoRL 0.4–0.7: Subsystems integrated; initial Lifecycle Assessments (LCA) run.	SoRL 4–6: Public feedback collected; community concerns addressed.
3. Deployment	TRL 7–9: Full-scale tech operating in real-world conditions.	IRL 7–9: Supply chains track Scope 3 emissions; connections meet environmental laws.	SoRL 0.8–1.0: Fully optimized, certified green system (e.g., ISO 14040).	SoRL 7–9: Widespread community adoption; deep institutional trust.

9.1.1. Operational Readiness: - Green Technologies

Operational Readiness Level (ORL) evaluates how prepared an organization, workforce, and infrastructure are to operate, support, and maintain a system throughout its active lifecycle. In sustainability, ORL ensures that green technologies are not just technically sound but can actually be kept running efficiently by real workers using realistic maintenance supply chains (Table 17).

ORL (Operational Readiness Level) for Sustainability

- a) ORL 1–3 (Logistics Planning): Operational concepts are defined; baseline resource requirements, energy costs, and required operator skillsets for the green tech are identified.
- b) ORL 4–6 (Procedures & Training): Standard Operating Procedures (SOPs) for sustainability metrics are drafted; maintenance staff are trained on handling low-carbon or hazardous circular components.
- c) ORL 7–9 (Full Fleet Operations): Maintenance supply chains are fully mature; logistics networks are optimized to handle continuous recycling, repair, and zero-waste disposal targets.

Table 17. Comprehensive Lifecycle Readiness Matrix

Phase	Technology (TRL) (Core Tech)	Integration (IRL) (Connections)	System (SRL) (Total Maturity)	Social (SRL-Soc) (Trust & Culture)	Operational (ORL) (Support & Maintenance)
1. Concept	TRL 1–3: Green concepts defined; environmental benefits calculated.	IRL 1–3: Supply chain flows mapped; data protocols designed.	SRL 0.1–0.3: High-risk phase; tech exists but lacks verified infrastructure.	SRL 1–3: Social impacts identified; key community stakeholders mapped.	ORL 1–3: Operational concepts defined; baseline maintenance and resource needs identified.
2. Prototype	TRL 4–6: Lab testing of low-carbon or circular materials.	IRL 4–6: Physical testing of green parts joining standard parts.	SRL 0.4–0.7: Subsystems integrated; initial Lifecycle Assessments (LCA) run.	SRL 4–6: Public feedback collected; community concerns addressed.	ORL 4–6: Green SOPs drafted; operators trained on energy and waste management systems.
3. Deployment	TRL 7–9: Full-scale tech operating in real-world conditions.	IRL 7–9: Supply chains track Scope 3 emissions; connections meet environmental laws.	SRL 0.8–1.0: Fully optimized, certified green system (e.g., ISO 14040).	SRL 7–9: Widespread community adoption; deep institutional trust.	ORL 7–9: Continuous operations validated; zero-waste recycling and repair logistics fully scaled.

9.1.2. Synthesis of TRL, IRL, SRL, SoRL, and OR (5-Dimensional-Activities)

The synthesis of TRL, IRL, SRL, SoRL (Social Readiness), and ORL creates a unified, 5-dimensional blueprint for systems engineering. This framework ensures a technology is mechanically sound, interconnectable, socially trusted, and operationally viable (Table 18).

The Unified Readiness Architecture: Systems fail when these metrics are misaligned (e.g., high TRL but low SoRL or ORL). True system maturity requires balancing all five vectors across three core engineering milestones:

[TRL: Can we build it?] -----\

[IRL: Can we connect it?] -----+--> [SRL: Does the whole system work?]

[SoRL: Will society accept it?] | ESG

[ORL: Can we run it daily?] ---/ Green Technology

1. The Core Engine (TRL & IRL) focuses on technical feasibility and interfaces.
2. The Operational Envelope (ORL & SoRL) focuses on human adoption, organizational capability, and public trust.
3. The Governance Umbrella (SRL): Synthesizes all inputs mathematically and programmatically to calculate total risk.

Table 18. Comprehensive 5-Dimensional Readiness Matrix

Phase	Technical Feasibility (TRL)	Interface Maturity (IRL)	Systemic Governance (SRL)	Social/Market Trust (SoRL)	Operational Viability (ORL)
1. Ideation & Concept	TRL 1–3: Basic principles observed; analytical and mathematical formulations validated.	IRL 1–3: Interface requirements defined; data exchange protocols conceptualized.	SRL 0.1–0.3 (Low): System concept defined; high uncertainty and architectural risk.	SoRL 1–3: Stakeholders mapped; ethical, cultural, and market biases identified.	ORL 1–3: Operational concept defined; baseline staffing and facility needs estimated.
2. Validation & Prototype	TRL 4–6: Component testing in laboratory and simulated operational environments.	IRL 4–6: Physical/logical interfaces tested; cross-component data flow validated.	SRL 0.4–0.7 (Medium): Subsystems integrated; system-level risk registers established.	SoRL 4–6: Public/user feedback integrated; regulatory compliance frameworks drafted.	ORL 4–6: SOPs drafted; training programs initiated; maintenance tooling verified.
3. Production & Scale	TRL 7–9: Actual system completed, qualified, and proven through successful mission operations.	IRL 7–9: All interfaces fully verified, secure, compliant, and operating under load.	SRL 0.8–1.0 (High): Optimized system; ready for legal certification and mass deployment.	SoRL 7–9: Widespread market adoption; deep public trust; legal clearance achieved.	ORL 7–9: Continuous operations active; supply chain secure; lifecycle maintenance scaled.

9.2. Mathematical Synthesis (Calculated SoR) and Interpretation

To prevent cognitive bias, organizations combine these values into a single Composite System Readiness Index. While TRL and IRL form the core matrix, SoRL and ORL act as systemic multipliers: $\{\text{System Readiness Index}\} = \text{factor}(\{\text{TRL}, \text{IRL}\}) \times (\{\text{SoRL}\}_{\text{factor}}) \times \{\text{ORL}\}_{\text{factor}}$ (Table 19)

- The Weakest Link Principle: If our TRL is 9, but our IRL or ORL is 2, our total calculated SRL drops heavily.
- The Gatekeeper Principle: A system cannot pass a major milestone review (e.g., moving from Prototype to Production) if any single readiness index lags more than two levels behind the others.

Table 19. Risk Matrix Comparison (Refer to Table 3)

Metric	Primary Risk Target	Eliminates This Threat
TRL	Component Functionality	Building on a fundamentally broken technology.
IRL	Connection Compatibility	Subsystems that work alone but fail when wired together.
SRL_Social	Public & Market Acceptance	Spending millions on a tool that communities actively reject.
SRL (Eng.)	Architecture Maturity	Overestimating system health due to localized component success.
ORL	Field Sustainability	Operations collapsing post-launch due to lack of training or parts.

9.2.1. In 2006, Sauser concluded that the relationship and dependency of TRL to SRL

"Sauser (2006)" refers to the foundational academic paper titled *"From TRL to SRL: The Concept of Systems Readiness Levels"*. Authored by Dr. Brian Sauser alongside Jose Ramirez-Marquez, Dinesh Verma, and Ryan Gove at the Stevens Institute of Technology, this work fundamentally changed systems engineering by introducing a mathematical way to measure entire system maturity instead of just individual components [25,26].

9.2.2. The Problem Sauser Addressed

Before 2006, complex engineering projects (especially within NASA and the U.S. Department of Defense) relied heavily on NASA's Technology Readiness Level (TRL). Sauser argued that TRL had a major blind spot [32–34]:

- ❖ TRL measures an isolated component in a lab.
- ❖ It does not measure how components interact or communicate.
- ❖ Engineering failures frequently happen at the interfaces (connections) where components meet, even if the individual parts are mature.

9.2.3. The Two New Metrics Introduced

To solve this gap, Sauser's 2006 framework added two concepts to complement TRL: [33,34]

1). Integration Readiness Level (IRL)

This measures the maturity of the *interfaces* between two components. The original 2006 model defined 7 distinct levels of interface maturity (later expanded to 9 to match TRL) [25,26]:

- IRL 1: An interface concept is defined.
- IRL 3: Compatibility between components is established.
- IRL 5: Control structure for integration is executed.
- IRL 7: Complete integration is demonstrated in an operational environment. [34]

2). System Readiness Level (SRL)

The SRL is a composite, quantitative index (ranging from 0.0 to 1.0) that calculates the *overall readiness of the entire system*. [32–35]

9.2.4. The Sauser Mathematical Matrix

Instead of subjectively guessing a system's maturity, Sauser introduced a repeatable, deterministic matrix formula:

- The TRL Vector: A single-column matrix listing the TRL of every individual technology component in the system.
- The IRL Matrix: A symmetric, pairwise matrix (often mapped using a Design Structure Matrix) that rates the connection maturity between every single intersecting component.

The mathematical synthesis synthesizes these values into a normalized score:

$[SRL] = [IRL] \times [TRL]$: - Design and Developed by Kujawski Model

$[SRL] = IRL_{ij} \times TRL_{ij}$: - Defined, Designed, and Developed by Sauser Matrix Model

$SRL = \text{factor}(\{TRL \text{ Vector}\} \times \{IRL \text{ Matrix}\})$: Developed by Sauser Matrix Model

Performing the matrix multiplication: $SRL = (IRL \times TRL) / n$: Initiated by the Sauser Team.

Why this matters: If a system has components at TRL 9 but they are poorly integrated (IRL 1), Sauser's matrix penalizes the score, pulling the overall SRL down. This keeps project managers from prematurely deploying complex systems [35–37].

Lasting Impact and Criticism

- a) Industry Standard: Sauser's work laid the groundwork for modern systems engineering lifecycle tracking used widely in aerospace, defense, energy, and global systems management.
- b) The Academic Debate: In the years following 2006, other researchers (such as Langford) criticized the model. They pointed out that performing algebraic matrix math on ordinal scales (like 1–9 rankings) is mathematically flawed. However, the framework remains highly valued as a prescriptive risk management tool.

10. Result Analysis

Result Analysis evaluates the calculated readiness index to identify critical bottlenecks, blind spots, and system risks before deployment. It translates the raw 1-to-5 or 1-to-9 scores across TRL, IRL, SRL, SoRL, and ORL into actionable engineering decisions. Types of Readiness Analysis. (Table 20)

- a. Gap Analysis: Identifies the numeric distance between your least mature vector and your most advanced vector.
- b. Sensitivity Analysis: Tests how a failure in one vector (e.g., a drop in SoRL due to public backlash) crashes the total SRL
- c. Critical Path Analysis: Determines the exact sequence of readiness upgrades required to hit the next major deployment deadline.
- d. Asymmetry Evaluation: Highlights dangerous imbalances, such as high technical readiness paired with low operational training.

Table 20. Matrix for Result Interpretation & Action Planning

Analysis Outcome	Numerical Pattern	System Diagnosis	Immediate Strategic Action
Technically Blind	High TRL/IRL Low SoRL/ORL	The technology works perfectly in a lab but cannot be maintained or accepted by the public.	Halt engineering; pivot funding to operator training and stakeholder engagement.
Socially Premature	High SoRL Low TRL/IRL	High market demand and public enthusiasm, but the core technology is unproven or unstable.	Manage public expectations; intensify laboratory prototyping and interface stress-testing.
Logistically Trapped	High TRL/SoRL Low IRL/ORL	Proven technology with high public trust, but it cannot connect to legacy infrastructure or supply chains.	Redesign data/physical interfaces; secure spare-part pipelines and draft operational SOPs.
Systemically Balanced	Matched Levels (All within ± 1 tier)	Harmonized development; risks are evenly distributed across technical, social, and operational spheres.	Authorize progression to the next lifecycle phase (e.g., moving from Prototype to Scale).

Metrics for Quantifying Success

- a) Readiness Variance (Delta R): Measures the spread between the highest and lowest scores; a variance greater than 2 indicates a high-risk system.
- b) Time-to-Ready (TTR): Predicts the number of months required to bring the lagging vector up to par with the leading vector.
- c) Investment Efficiency Index: Calculates the system maturity gained per dollar spent on a specific readiness vector.

11. Discussion

A Discussion phase provides a critical forum for system architects, data analysts, and executive stakeholders to debate the nuances, trade-offs, and strategic implications revealed during the Result Analysis. It moves the project from passive data calculation to active, high-level governance. Core Objectives of the Discussion. (Table 21)

- a) Arbitrate Conflicting Priorities: Resolves tension between engineering teams (driving TRL/IRL) and business/operations units (driving SoRL/ORL).
- b) Define Risk Tolerance Boundaries: Determine whether the organization will accept a lopsided readiness profile to beat competitors to market.
- c) Allocate Strategic Resources: Redirects budgets and personnel to the specific readiness vectors identified as critical path bottlenecks.
- d) Formulate Contingency Mitigations: Establish "Plan B" triggers if a lagging vector fails to mature before the scheduled deployment date.

Table 21. Sample Discussion Agenda for Review Boards

Time	Segment	Focus Question	Key Deliverable
00:00–00:15	Data Alignment	What did the Result Analysis reveal as our absolute weakest readiness vector?	Consensus on the core systemic bottleneck.
00:15–00:45	Impact Assessment	How does this bottleneck delay our target deployment date or violate compliance?	A quantified risk-impact statement.
00:45–01:15	Resource Reallocation	Can we pull budget from our advanced vectors to accelerate the lagging vector?	A revised, cross-disciplinary funding model.
01:15–01:30	Action & Sign-off	What specific verification criteria must the lagging vector meet in the next 30 days?	A signed Readiness Action Plan (RAP).

Strategic Discussion Framework: When conducting a readiness review board, stakeholders evaluate the system using four core lenses:

1. The Technology vs. Operations Trade-off
 - a. The Debate: Pushing a system to TRL 8 gives the organization a cutting-edge technological advantage. However, if ORL remains at level 4, field technicians will lack the training and tools to fix it when it breaks.
 - b. Key Question: *Do we delay the system launch to let operations catch up, or do we launch now and accept high initial maintenance costs and downtime?*
2. The Innovation vs. Public Trust Paradox
 - a) The Debate: Highly innovative, green, or automated systems often outpace public understanding or regulatory frameworks (High TRL, Low SoRL).
 - b) Key Question: *How do we transparently communicate system safety to local communities to prevent a public backlash that could freeze the entire project?*
3. Legacy Integration Hurdles
 - a) The Debate: A new system may be internally perfect, but connecting it to decades-old legacy infrastructure (Low IRL) introduces severe cybersecurity and data-handling vulnerabilities.
 - b) Key Question: Should we build custom, complex software bridges (high risk), or heavily invest in upgrading the legacy environment itself?

12. Future Look

The Future Look explores how emerging technological, regulatory, and societal shifts will redefine system maturity assessments over the next decade. As systems become more autonomous, interconnected, and resource-constrained, the traditional boundaries of TRL, IRL, SRL, SoRL, and ORL are morphing into a dynamic, continuous evaluation ecosystem (Table 22).

Major Paradigms Reshaping Readiness

- 1). Cognitive and AI Autonomy: Traditional readiness frameworks assume static, deterministic code. Future systems powered by adaptive AI and neural networks will continuously learn and evolve after deployment.
 - a. Dynamic TRL/IRL: TRL will no longer be a one-time sign-off. Systems will require continuous recertification as AI models drift, retrain, or update in the field.

- b. Algorithmic Trust (SoRL): Social readiness will shift from measuring user acceptance to measuring user *trust* in automated decisions, ethical AI alignments, and algorithmic transparency.
- 2). Digital Twins and Continuous Simulation: The reliance on physical prototyping to advance readiness levels is being replaced by ultra-precise, real-time Digital Twins.
 - a) Instantaneous SoRL: Real-time data loops will feed from active field components directly back into digital twins. This creates a living, fluctuating SoRL score rather than a static quarterly report.
 - b) Predictive ORL: Maintenance teams will simulate operational wear-and-tear years in advance, allowing ORL to reach maturity before physical hardware is even manufactured.
- 3). Regulatory and Climate Hardening: Global sustainability mandates (such as strict Scope 3 supply chain disclosures and circular economy laws) are turning readiness into a legal gatekeeper.
 - A. The "Green" SoRL Threshold: Systems will be legally blocked from advancing past prototype phases if their lifetime carbon intensity or e-waste footprint exceeds strict regional limits.
 - B. Geopolitical Resiliency: Interoperability (IRL) will explicitly measure a system's ability to seamlessly switch suppliers or data routes during geopolitical disruptions or localized climate crises.

Table 22. Future Readiness Evolution Roadmap

Timeframe	Key Metric Transformation	System Engineering Impact
Near-Term (1–3 Years)	Integration of ESG metrics directly into standard TRL/SRL/SoRL/ORL compliance calculators.	Sustainability becomes a non-negotiable passing requirement for design reviews.
Mid-Term (3–7 Years)	Widespread adoption of live Digital Twins to automate IRL and ORL tracking.	Physical test-benches are heavily downscaled in favor of massive cloud-based stress simulations.
Long-Term (7–10+ Years)	Emergence of "Autonomous SRL" for self-healing, self-updating AI systems.	Systems dynamically self-assess and patch their own readiness gaps without human intervention.

13. Conclusion

The System Readiness Level (SRL) represents a paradigm shift in system security, moving the focus from the excellence of individual "parts" to the resilience of the "interconnected whole." It provides a rigorous, mathematical bridge that ensures integration is secure, interoperability is meaningful, and standardization is effectively implemented.

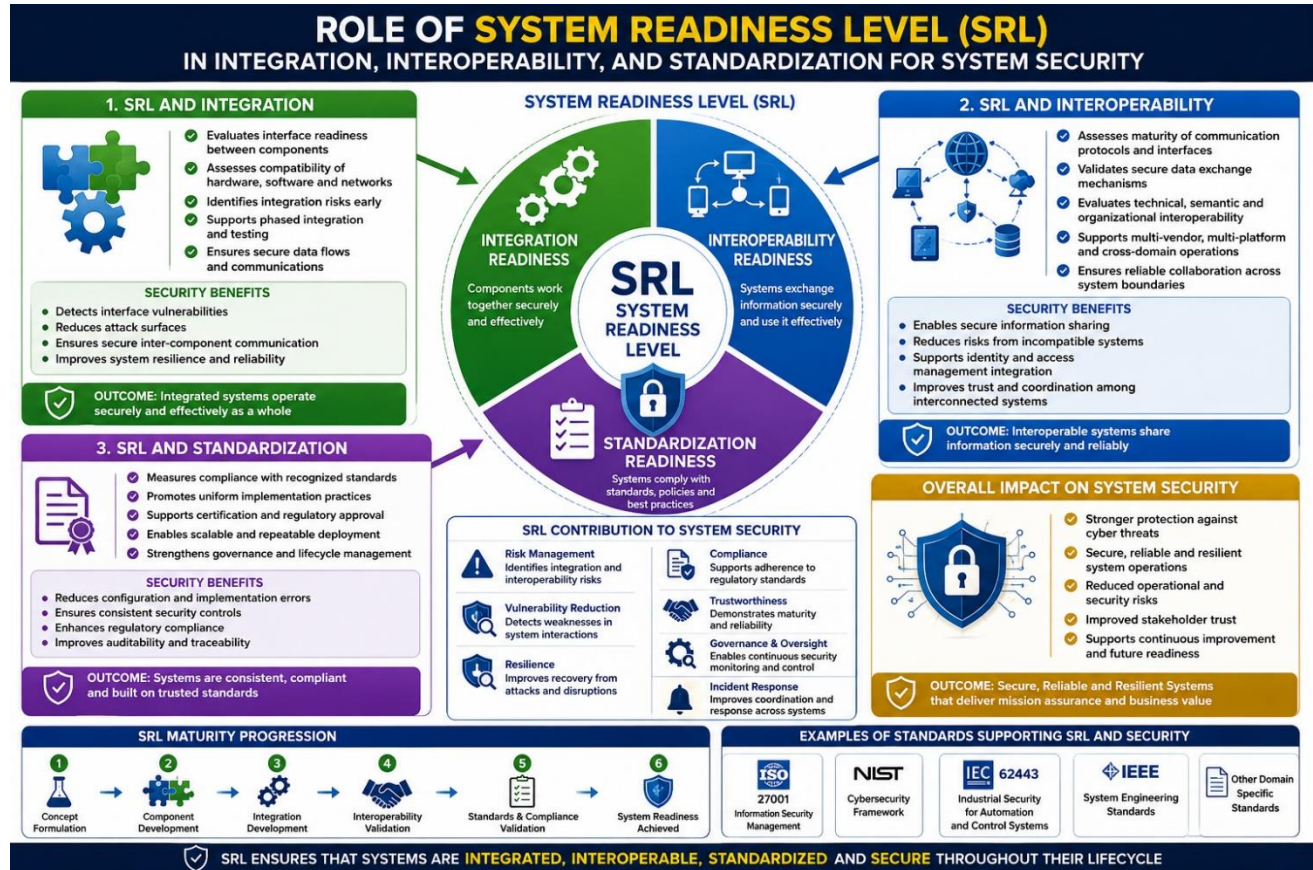
Key Takeaways

- a) Beyond the Tool: High-maturity components (TRL 9) provide a false sense of security if the links between them (IRL) are immature. SRL exposes these "hidden seams" where most modern breaches occur.
- b) The Power of Standardization: Standardized protocols (like TLS 1.3 or STIX/TAXII) are the primary drivers of system readiness. They reduce complexity, minimize the attack surface, and allow for a truly interoperable "System-of-Systems."
- c) Predictive Governance: By using the SRL Performance Matrix, organizations can move from subjective checklists to data-driven decision-making, ensuring that systems are only deployed when they are mathematically proven to be "Mission Ready."
- d) Resilience as a Result: A high SRL score is a leading indicator of fault tolerance and defensive agility, proving that the architecture can survive partial failures and adapt to evolving threats.

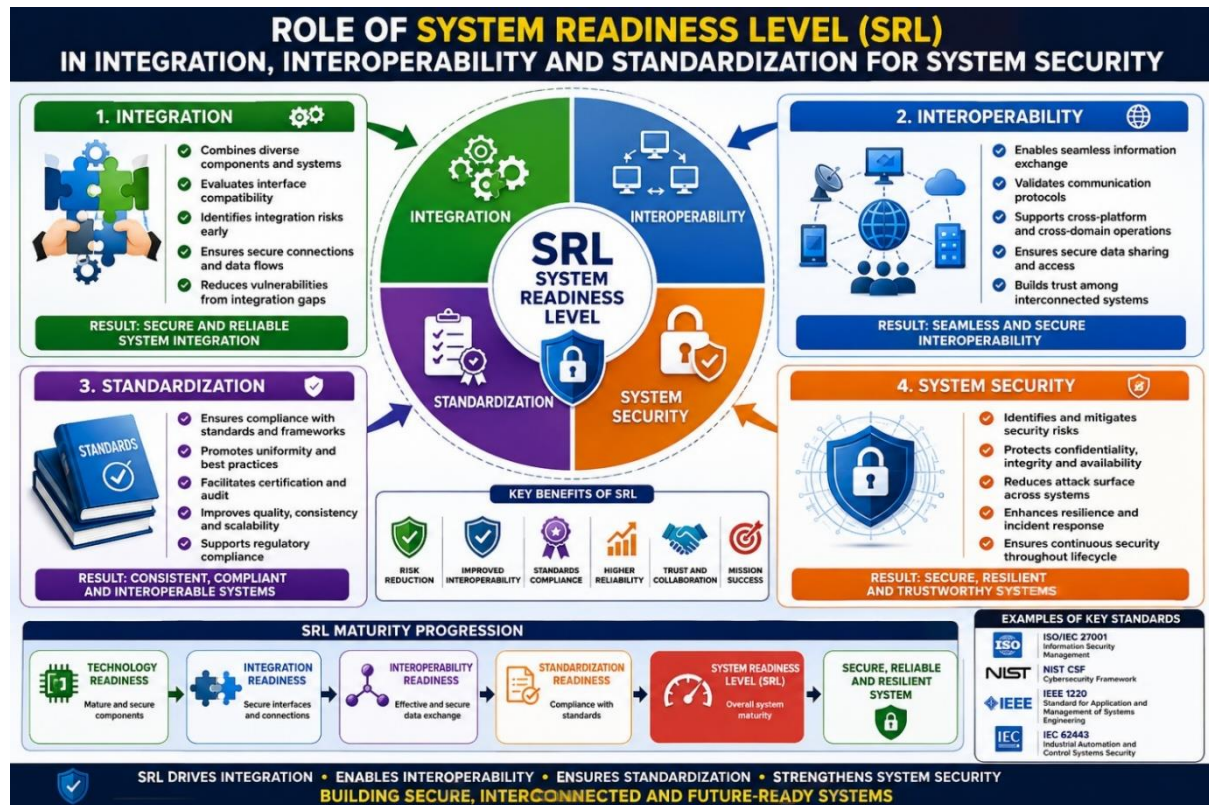
Final Synthesis

In an era of increasing architectural complexity—spanning Cloud, IoT, and AI—the SRL framework is no longer an optional engineering exercise; it is a foundational security requirement. It ensures that as our systems become more connected, they become more robust, rather than more vulnerable.

APPENDIX-I SRL for IIS



APPENDIX-II-SRL for IIS-Future Ready System



Funding:

Not Available.

Conflict of Interest:

No Conflict of Interest.

Data Availability Statement:

As per Publisher Policy.

Acknowledgment

The Author (Dr. PL Pradhan) like to thanks the Editors and Reviewers for their extreme supports and cooperation to improve this valuable research paper for publication.

References

1. Austin, M. and D. York, "Integration readiness levels evaluation and systems architecture: A literature review," *Int. J. Adv. Eng. Res. Sci.*, vol. 5, no. 4, pp. 114–122, Apr. 2018.
2. Chagas Jr., M. F., *et al.*, "Identifying Integration Readiness Levels Oriented Work Packages to Establish Earned Readiness Management," in *Proc. IEEE SysCon*, 2023.
3. Engineering Trustworthy Secure Systems: Considerations for a Multidisciplinary Approach in the Engineering of Trustworthy Secure Systems, NIST SP 800-160, vol. 1, rev. 1, 2022.
4. Fisher, M., *et al.*, "Verifiable Self-Certifying Autonomous Systems," in *Proc. IEEE ISSREW*, 2018.
5. Institute of Electrical and Electronics Engineers, *IEEE P3130—Security Requirements for Connected Vehicle Operating Systems*, Draft Standard, 2024.
6. Institute of Electrical and Electronics Engineers, *IEEE Std 1012-2024—System, Software, and Hardware Verification and Validation*, 2024.
7. Institute of Electrical and Electronics Engineers, *IEEE Std 1581-2025—Static Component Interconnection Test Protocol and Architecture*, 2025.
8. Institute of Electrical and Electronics Engineers, *IEEE Std 1619.2-2025—Wide-Block Encryption for Shared Storage Media*, 2025.
9. Institute of Electrical and Electronics Engineers, *IEEE Std 1904.4-2025—Service Interoperability in Ethernet Passive Optical Networks*, 2025.
10. Institute of Electrical and Electronics Engineers, *IEEE Std 7009-2024—Fail-Safe Design of Autonomous Systems*, 2024.
11. Institute of Electrical and Electronics Engineers, *IEEE Std 802.1AS-2025—Timing and Synchronization for Time-Sensitive Applications*, 2025.
12. Institute of Electrical and Electronics Engineers Systems Council, *Proceedings of IEEE Systems Security Symposium (SSS)*, 2020.
13. Institute of Electrical and Electronics Engineers, *P7015—Standard for Data and Artificial Intelligence (AI) Literacy, Skills, and Readiness*, Work in progress, 2026.
14. International Organization for Standardization, *ISO/IEC/IEEE 24748-4-2025—Systems and Software Engineering—Life Cycle Management*, 2025.
15. Koo, J.-I. and S.-J. Jeong, "Improved Technology Readiness Assessment Framework for System-of-Systems From a System Integration Perspective," *IEEE Access*, vol. 12, pp. 23827–23853, 2024.
16. Kujawski, E., "The trouble with the System Readiness Level (SRL) index for managing the acquisition of defense systems," *Syst. Eng.*, vol. 16, no. 3, pp. 329–340, 2013.

17. Li, B., *et al.*, “Physical-Layer Security in Space Information Networks: A Survey,” *IEEE Internet Things J.*, vol. 7, no. 1, pp. 33–52, 2020.
18. Malone, P. K., “Applying System Readiness Levels to Cost Estimates—A Case Study,” in *Proc. IEEE Aerospace Conf.*, 2018.
19. Mankins, J. C., “Technology readiness levels: A white paper,” National Aeronautics and Space Administration, Washington, DC, USA, 1995.
20. Mufti, Y., M. Niazi, *et al.*, “A Readiness Model for Security Requirements Engineering,” *IEEE Access*, vol. 6, pp. 28611–28631, 2018.
21. Nelson, P. R., “Improving Security and Interoperability of Interwoven Systems Through Rigorous Selective Encapsulation,” in *Proc. IEEE FASW**, 2018.
22. Neto, A. S. and M. de Castro, “Using systems architecture views to assess integration readiness levels (IRL),” *IEEE Syst. J.*, vol. 16, no. 2, pp. 2195–2206, Jun. 2022.
23. National Institute of Standards and Technology, *Cybersecurity Framework (CSF) 2.0*, 2024.
24. Olechowski, A., S. D. Eppinger, and N. Joglekar, “Technology readiness levels at 40: A study of state-of-the-art use, challenges, and opportunities,” in *Proc. ICED*, 2015.
25. Sauser, B., D. Verma, J. Ramirez-Marquez, and R. Gove, “From TRL to SRL: The concept of system readiness levels,” in *Proc. Conf. Syst. Eng. Res. (CSER)*, 2006.
26. Sauser, B., J. E. Ramirez-Marquez, D. Henry, and D. DiMarzio, “A systems approach to expanding the technology readiness level within defense acquisition,” *Int. J. Defense Acquisition Manage.*, vol. 1, pp. 39–58, 2008.
27. “A Systematic Review on Cyber Security Integration in Information Technology Governance Frameworks,” Institute of Electrical and Electronics Engineers, 2025.
28. “A Unified Assessment Methodology for Incident Forecasting Through Cyber Threat Intelligence Integration,” Institute of Electrical and Electronics Engineers, 2025.
29. “Cyber Physical Systems Security,” Institute of Electrical and Electronics Engineers, 2026.
30. “Security in 6G-Based Autonomous Vehicular Networks: Detecting Network Anomalies With Decentralized Federated Learning,” Institute of Electrical and Electronics Engineers, 2024.
31. “Strategic Integration of Security Product Standard: A Comparative Study of Early vs. Late Implementation in Software Development,” in *Proc. IEEE Conf.*, 2024.
32. Department of Defense, *Technology Readiness Assessment (TRA) Guidance*, 2011.
33. Institute of Electrical and Electronics Engineers, *IEEE International Conference on Cyber Security and Resilience (CSR)*, 2026.
34. E. Kujawski, “Analysis and Critique of the System Readiness Level,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 43, no. 4, pp. 979–987, Jul. 2013.
35. Sauser, J. E. Ramirez-Marquez, D. Henry, and D. DiMarzio, “A system maturity index for the systems engineering life cycle,” *Int. J. Industrial and Systems Engineering*, vol. 3, no. 6, pp. 673–691, 2008.
36. B. Sauser, J. E. Ramirez-Marquez, R. Magnami, and W. Tan, “A systems approach to expanding the technology readiness level within defense acquisition,” *IEEE Systems Journal*, vol. 3, no. 2, pp. 249–258, Jun. 2009.

37. W. Tan, B. Sauser, and J. E. Ramirez-Marquez, "Analyzing component importance in system maturity assessment," *IEEE Transactions on Engineering Management*, vol. 58, no. 2, pp. 275–288, May 2011.