Journal of
Cyber Security

Tech Science Press

https://doi.org/10.32604/jcs.2026.087653

ARTICLE

Secure AI Development Lifecycle (SAIDL): A Multi-Layer Framework for Mitigating AI-Enabled Cyberattacks in Intelligent Systems

Dhaya Ramakrishnan[1,*], **Kanthavel Radhakrishnan**[1] **and Adline Freeda Russel**[2]

[1]School of ECE, Papua New Guinea University of Technology, Lae, Papua New Guinea
[2]IT Department, KCG College of Technology, Chennai, India

*Corresponding Author: Dhaya Ramakrishnan. Email: dhayavel2005@gmail.com

Received: 20 June 2026; Accepted: 17 September 2026; Published: 20 October 2026

ABSTRACT: Artificial intelligence (AI) is an integral part of vital domains such as health care, financial services, logistics, manufacturing industries, national defense, and smart infrastructure. AI technologies deliver tangible advantages for tasks such as automation, predictive analysis, and decision-making; however, at the same time, they pose new challenges related to the potential exploitation of emerging vulnerabilities by malicious actors. The emergence of various types of AI-driven cyberattacks, ranging from automated phishing attacks to adversarial machine learning (AML) and deepfake social engineering, has shown that current cybersecurity approaches cannot ensure effective defense against threats associated with AI technologies. Current security approaches primarily focus on protecting conventional software and do not necessarily address the security needs that arise specifically from AI development, implementation, and management. We propose adopting the Secure AI Development Lifecycle (SAIDL) approach, a multi-layered framework for incorporating adequate security controls at all phases of the AI development lifecycle. The proposed SAIDL framework comprises six interrelated layers that should be addressed in AI cybersecurity efforts: Data Security, Model Security, Training Security, Deployment Security, Runtime Monitoring, and Incident Response and Recovery. Each layer has its own security mechanisms designed to counteract AI-driven attacks without compromising the performance, integrity, and robustness of the models. Moreover, the paper describes an AI Security Risk Model along with an AI Resilience Score to estimate the readiness of organizations toward AI-driven threats. In general, the SAIDL architecture presents a secure-by-design solution for AI technologies through the integration of cybersecurity best practices, AML measures, continuous monitoring, and governance strategies. Therefore, the framework provides a conceptual and analytical approach for coordinating security controls across the AI lifecycle and assessing their potential contribution to the resilience of intelligent systems.

KEYWORDS: Artificial intelligence security; adversarial machine learning; AI-enabled cyberattacks; secure AI development lifecycle; cybersecurity framework; trustworthy AI; threat intelligence; intelligent systems

1 Introduction

AI is one of the central technologies used in modern digital ecosystems, helping organizations automate processes, analyse data, make predictions, and conduct decision-making activities in domains such as healthcare, finance, logistics, manufacturing, defense, and smart infrastructure. While AI-based technologies increase the effectiveness and efficiency of operations and service delivery, their broad use in organizations results in an expansion of the threat landscape associated with cyber threats that conventional security models cannot mitigate [1,2].

The development of new machine learning, deep learning, and generative AI techniques has revolutionized both cybersecurity and cyberattacks. AI is used for various aspects of cybersecurity, including threat

Copyright © 2026 The Authors. Published by Tech Science Press. This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

detection, malware detection, and incident handling; yet, cybercriminals use AI-based techniques to conduct advanced cyberattacks, such as phishing campaigns, impersonation using deepfakes, intelligent malware creation, and vulnerability discovery [3–6].

Unlike traditional software systems, the security of AI systems is dependent on factors such as data quality, model architecture, training processes, and inference processes. This has resulted in attack vectors that target all stages of the AI lifecycle. AML attack vectors, which include but are not limited to data poisoning, adversarial evasion, model extraction, membership inference, model inversion, and backdoor insertion, may impact the confidentiality, integrity, and availability of AI systems [7,8]. Moreover, the increasing prevalence of LLMs has introduced threats such as prompt injection, jailbreak attacks, prompt leakage, and agent manipulation [9–12].

Several organizations have published frameworks and guidelines addressing AI security and cybersecurity. Notable examples include the National Institute of Standards and Technology Artificial Intelligence Risk Management Framework (NIST AI RMF), MITRE Adversarial Threat Landscape for Artificial-Intelligence Systems (MITRE ATLAS), and Open Worldwide Application Security Project Large Language Model (OWASP LLM) Security. Although these frameworks provide valuable guidance, existing AI security approaches tend to focus on specific attack vectors, governance requirements, or individual security controls rather than providing comprehensive security orchestration across the entire AI lifecycle.

With the increasing reliance on AI in critical applications, it is necessary to develop security frameworks that ensure security across all stages of the AI lifecycle. This paper proposes a security architecture to address this need. The proposed SAIDL is a multi-layer security framework for AI systems designed to ensure cybersecurity throughout the creation, training, deployment, and operation of AI models and intelligent algorithms. There are six major layers in this architecture: the Data Security Layer, Model Security Layer, Training Security Layer, Deployment Security Layer, Runtime Monitoring Layer, and Incident Response & Recovery Layer.

The contributions of this study can be summarized as follows:

- Formulation of SAIDL, a security framework for mitigating AI-enabled attacks with a focus on the AI lifecycle.
- Formulation of a multi-layered, defense-in-depth security approach to protect AI systems throughout their lifecycle.
- Formulation of a mathematical model to measure threat mitigation, residual risk, and resilience throughout the AI lifecycle.
- Formulation of a Secure AI Security Index (SSI) that measures AI security performance and resilience.
- Provision of a structured roadmap for implementing secure-by-design AI systems in security-critical environments.

Novelty and Research Contributions

What makes the SAIDL framework novel is the incorporation of security controls, adaptive threat intelligence, resilience evaluation, and quantitative security assessment into a single framework. Unlike current approaches, which are more focused on governance or security control management, SAIDL is designed to incorporate security considerations throughout all stages of the AI system lifecycle. The novelty of SAIDL is mainly architectural, integrative, and quantitative in nature rather than algorithmic. This methodology establishes connections between various security measures at different stages of the AI system lifecycle in a systematic manner with the help of adaptive threat intelligence feedback. Moreover, the proposed mathematical formulation helps establish how threats evolve across different security layers and provides quantifiable measurements of mitigation, threat, resilience, and security for an AI system using the SSI.

The main innovations of this framework include:

- Six layers of security architecture designed to counter specific threats against AI throughout all phases of the lifecycle.
- An Adaptive Threat Intelligence Feedback Engine to update security controls based on encountered threats.
- A quantitative mathematical model describing the concepts of threat reduction, resilience creation, and security improvement adaptively.
- A single metric, called SSI, for measuring AI security performance and resilience.

Overall, this combination creates a solid foundation for the implementation of AI security by design and distinguishes SAIDL from existing AI security frameworks.

Despite the absence of a novel machine learning or cybersecurity algorithm within SAIDL, its innovation lies in providing an integrated security architecture for the entire AI lifecycle, allowing AI security techniques to be incorporated into a coherent security operation. Unlike other existing AI security guidelines, SAIDL brings together six security layers, an adaptive threat intelligence feedback mechanism, mathematical security modeling techniques, and the SSI. These elements provide an overall perspective on how AI can be secured throughout its lifecycle, unlike existing AI security guidelines, which are mostly prescriptive.

Theoretical and Practical Progression: In terms of theoretical progression, the advancement of this study can be stated as the conceptualization of AI security as a lifecycle-oriented security transformation, in which the threat state is progressively reduced through the integration of security layers. The mathematical model proposed in this research provides a framework for representing mitigation capability, residual threat state, security value, security lifecycle resilience, and threat intelligence adaptation. The SSI allows the overall level of AI security to be assessed in a unified manner. Practically, SAIDL provides a framework for coordinating and aligning security controls across all aspects, including data preparation, model development, training, deployment, execution time, incident management, and recovery. The Inter-Layer Threat Intelligence Feedback mechanism enables security information received during runtime and incident response to inform future security decision-making. Hence, the particular contribution of this research is the creation of an integrated approach that combines architectural security orchestration with quantitative security assessment. This creates a foundation for shifting from fragmented and standards-oriented approaches to AI security toward a comprehensive secure-by-design architecture.

The rest of this paper is structured as follows. Section 2 reviews relevant works on adversarial machine learning, AI-driven attacks, security of large language models, and frameworks for secure AI development. Section 3 introduces the proposed SAIDL framework, including its structure, security layers, mathematical formulation, and security orchestration mechanism. Section 4 describes the evaluation methodology, datasets, attack strategies, baseline models, and metrics. Section 5 presents the practical evaluations and analyzes the results. Section 6 discusses practical applications, limitations, and potential directions for future research related to the proposed framework. Section 7 presents the conclusion.

2 Literature Review

The broad use of AI technology in critical infrastructures has brought about significant changes in the field of cybersecurity while also posing various security risks to these systems. Issues related to AML, AI-based attacks, LLM security, and secure AI engineering have become focal points of recent research. Despite substantial progress in these areas, ensuring the security of AI systems throughout their lifecycle remains a challenging problem.

2.1 AML and AI Security Threats

The field of AI security is characterized by the increasing prevalence of AML attacks [13]. Poisoning, adversarial examples, model extraction and inversion, and backdoor attacks against ML systems can violate the principles of confidentiality, integrity, and availability [14–16]. Recent studies have shown that many existing AI solutions lack sufficient security validation and cannot be adequately protected using traditional cybersecurity techniques. Several studies have investigated the continuous evolution of AML attacks and defense mechanisms. Recent survey studies have examined adversarial attacks, defense mechanisms, applications, and future research directions in this field. Other recent studies have investigated adversarial machine learning from an industrial perspective, focusing on how research-based attacks and defenses can be implemented in industrial settings. Although these studies provide a solid foundation for understanding individual attacks and defenses in adversarial machine learning, they alone cannot provide the basis for a complete lifecycle-oriented security architecture.

2.2 AI-Enabled Cyberattacks

Generative AI has led to advances in cyberattacks through the automation of phishing, intelligent malware generation, adaptive reconnaissance, and automated vulnerability detection [17,18]. With advancements in generative AI, it is now being applied not only for cyber defense but also as a tool for conducting cyberattacks. This highlights the importance of incorporating security measures at each stage of AI technology development and deployment.

2.3 Security Challenges in Large Language Models

Prompt injection, jailbreak attacks, prompt leakage, prompt manipulation, and backdoor attacks are among the emerging security issues associated with large language models [19–23]. Recent studies have confirmed that many leading LLMs remain vulnerable to malicious prompts that can override safety measures and steer model operations toward unintended outputs. In addition, recent studies have broadened the scope of LLM security from traditional applications to tool-using and autonomous AI agents. For agentic AI, the consequences of prompt injection can be greater because agents may have access to tools, external data, and system resources beyond the prompt input. In light of these concerns, recent security studies have sought to identify design patterns and multiple layers of security for defending agentic LLMs against prompt injection and similar attacks. This indicates that AI security now encompasses not only monitoring but also the coordination of control processes beyond the model interface. Nevertheless, security technologies designed for LLMs still primarily address specific threat vectors.

2.4 Secure AI Development and Lifecycle Protection

To address emerging security threats in AI systems, researchers have proposed approaches based on secure design principles that incorporate aspects such as adversarial testing, governance, deployment security, monitoring, and AI/MLSecOps [24,25]. Nevertheless, most existing approaches concentrate on implementing specific security measures without offering a holistic approach to ensuring cybersecurity across the entire AI lifecycle, including data collection, model training, deployment, runtime monitoring, and incident recovery. Emerging secure AI development practices highlight the necessity of integrating security considerations throughout the entire AI lifecycle rather than treating security solely as a deployment-stage task. Existing methodologies offer useful techniques for secure development, risk assessment, testing, validation, and deployment. However, these approaches differ significantly in terms of scope and level of involvement in the AI development process. In particular, there is a lack of mechanisms that connect security information gathered during runtime and incident management with processes such as data preparation, model development, and training.

AI governance and risk management frameworks provide an important organizational basis for the responsible deployment of AI. The NIST AI RMF, for example, provides a voluntary framework that enables trustworthiness to be considered throughout the process of designing, developing, using, and evaluating AI, while the Generative AI Profile of the same framework addresses risks specifically associated with generative AI. Such frameworks are useful for governance and risk management but are not intended to serve as comprehensive architectures for coordinating security controls across all layers of the AI lifecycle. This need motivates the usefulness of SAIDL.

Empirical evaluations have become increasingly important for establishing the level of security provided by proposed AI security techniques under realistic attack conditions. In recent years, many researchers have performed empirical evaluations of security techniques based on adversarial examples, benchmark datasets, and attack scenarios. The results show that both security and the effects of defense techniques on system performance should be evaluated when measuring the level of security. However, differences in evaluation approaches across attacks and application areas make comparisons of security levels across the entire AI lifecycle more difficult.

2.5 Research Gap

Despite the extensive research dedicated to AML attacks, AI-based cyberattacks, and LLM security issues, several challenges remain. First, current approaches focus on individual attack categories and overlook security considerations across the entire AI lifecycle. Second, existing frameworks prioritize components such as governance and threat detection but provide limited practical guidance regarding actual deployment. Third, there is a lack of quantifiable methods for assessing AI security. Fourth, with the emergence of new AI developments, such as generative AI, autonomous agents, and applications incorporating LLMs, new attack vectors continue to emerge. Moreover, most existing AI security frameworks tend to focus on governance principles, risk management, or specific security measures without providing an operational architecture that includes both security orchestration throughout the lifecycle and quantitative security assessment. To address this problem, SAIDL provides an architecture that incorporates all these elements.

Although many AI security guidelines and threat-oriented frameworks have already been developed, an important research gap remains in integrating security controls throughout the entire AI lifecycle. Current approaches usually cover specific threats, lifecycle stages, security governance aspects, or application-specific security concerns, but they do not provide a unified architecture that combines these security aspects using threat intelligence feedback and quantitative security assessment.

The proposed SAIDL framework attempts to fill this gap by combining six security layers related to data security, model security, training security, deployment security, runtime monitoring, and incident response and recovery. The framework also includes a Threat Intelligence Feedback component that facilitates communication among the security layers and a mathematical formulation for estimating threat reduction, residual threat, security contribution, resilience, and SSI. Therefore, the contribution of SAIDL is not the development of a new standalone security control or machine learning approach but the unification of these elements within a unified security architecture.

Consequently, the research gap addressed by the current work can be described as follows: existing AI security techniques provide several useful controls and recommendations; however, there is a lack of approaches that orchestrate these controls, propagate threat information from one lifecycle phase to another, and quantitatively measure the effectiveness of the resulting level of AI security. The SAIDL approach was developed specifically to address these problems.

Difference from Existing AI Security Approaches: There are several important distinctions between SAIDL and other AI security approaches. The NIST AI Risk Management Framework (AI RMF) is aimed

primarily at risk management related to AI technologies by providing governance, mapping, measurement, and management of AI risks. In contrast, SAIDL focuses more on the orchestration of security measures throughout the AI lifecycle through interconnected security layers. MITRE ATLAS provides a repository of knowledge and a threat-informed framework regarding adversarial techniques and tactics against AI-powered systems, whereas SAIDL uses threat information to develop a protection and mitigation mechanism across the AI lifecycle. OWASP LLM security guidance focuses on risks and vulnerabilities specifically associated with large language models, whereas SAIDL addresses all stages of the AI lifecycle, including data preparation, model creation, training, deployment, monitoring, and incident recovery. Other secure AI development approaches contribute secure development principles and specific security measures but lack the combination of lifecycle security orchestration, threat intelligence feedback across layers, quantitative threat transformation, resilience analysis, and SSI.

The distinction is therefore not that SAIDL replaces NIST AI RMF, MITRE ATLAS, or OWASP guidance, but that it provides an architectural mechanism for coordinating security controls and quantitatively assessing their combined contribution across the AI lifecycle. NIST AI RMF provides a comprehensive structure for managing AI risks through functions such as governance, mapping, measurement, and management. The merit of this framework is that it creates a common language for risk management and establishes an organizational process. However, it is a risk management framework rather than a technical architecture that defines how security controls should operate at each stage of the AI lifecycle. Specifically, it does not provide a mechanism for propagating runtime threats through the data, model, training, deployment, and recovery layers or a single index for measuring the cumulative effect of security layers.

Limitations of Existing AI Security Lifecycle Models: Current AI security lifecycle models offer valuable guidance regarding risk management, governance, threat detection, and security-control selection; however, several shortcomings remain. First, security controls are usually presented as advice or recommendations rather than as an integrated approach that connects different phases of the AI lifecycle. Second, current approaches lack mechanisms for propagating information about detected threats from runtime monitoring and incident handling back to data preparation, model creation, and training. Third, the quantitative assessment of the influence of different security layers on reducing threat presence is not adequately considered. Fourth, many current approaches focus on specific types of attacks, application domains, or phases of the AI lifecycle. Fifth, adaptive security measures that can use newly discovered threats at runtime are not adequately covered. These limitations have motivated the development of the SAIDL framework, which encompasses security controls throughout the lifecycle by incorporating inter-layer threat intelligence feedback and providing a mathematical model for measuring threat reduction, residual threats, security contributions, and resilience using the SSI.

For additional positioning of the suggested framework, Table 1 presents SAIDL vs. representative state-of-the-art approaches based on the following criteria: security objective, life cycle span, threat intelligence feedback between layers, inter-layer cooperation, quantitative evaluation of security measures, and adaptive security features. From the comparison, it can be seen that existing approaches possess relevant capabilities, although each one addresses certain security aspects in AI. SAIDL is considered as an integrative approach that integrates life-cycle security measures along with inter-layer feedback and quantitative assessment of security. As a whole, the aforementioned literature indicates considerable advancements made in the field of adversarial ML, AI governance, LLMs security, and safe AI development. However, there is no unity in the research strands discussed with regard to lifecycle integration, inter-layer coordination, adaptive feedback, and quantifiable assessment of security. It is due to this deficiency that the SAIDL approach has been proposed.

Table 1: Comparative positioning of SAIDL with existing AI security frameworks.

Framework/ Approach	Primary Focus	Lifecycle Coverage	Threat Intelligence/ Feedback	Inter-Layer Coordination	Quantitative Security Assessment	Adaptive Security	Main Limitation
NIST AI RMF	AI risk management and governance	Broad lifecycle	Limited	Limited	Risk-oriented assessment	Limited	Primarily provides risk-management guidance rather than an integrated operational security architecture
MITRE ATLAS	AI/ML attack tactics and techniques	Threat-oriented	Threat-informed	Not a lifecycle orchestration mechanism	No unified security index	Limited	Primarily a knowledge base for understanding and mapping adversarial techniques
OWASP LLM Security Guidance	LLM/application security	Primarily LLM application lifecycle	Limited	Limited	No unified lifecycle security index	Limited	Focused mainly on LLM-specific security risks
Conventional AI Security Pipeline	Security controls across development/deployment	Partial	Limited	Limited	Application-dependent	Limited	Security controls may remain fragmented across lifecycle stages
Secure AI Development Methodologies	Secure development practices	Development-oriented	Varies	Varies	Varies	Varies	Often emphasizes secure development practices rather than continuous lifecycle-wide security orchestration
Proposed SAIDL	Lifecycle-wide AI security orchestration	Data → Model → Training → Deployment → Runtime → Response/Recovery	Yes	Yes	Threat reduction, resilience, SSI	Yes	Requires further validation through real-world operational deployments

3 Proposed Methodology

The following section details the proposed SAIDL, a multi-layered security framework designed to protect AI systems from AI-enabled cyberattacks and adversarial machine learning-based cyber-attacks. In its operation, the SAIDL uses a secure by design approach through integration of cybersecurity controls into all stages of the operational lifecycle of an AI system. As compared to conventional security frameworks, the SAIDL provides end-to-end security from data acquisition all the way to recovery from incidents. Six main layers of the security framework include: Data Security Layer, Model Security Layer, Training Security Layer, Deployment Security Layer, Runtime Monitoring Layer, and Incident Response & Recovery Layer.

Collectively, these layers establish a defense-in-depth security architecture that protects AI systems against adversarial attacks, prompt injection, model theft, data poisoning, backdoor insertion, and AI-enabled cyberattacks.

3.1 Proposed SAIDL Architecture

The SAIDL approach is developed as a lifecycle-oriented architecture intended to support security across the AI system lifecycle. While each of the layers serves its own security purposes, they also interact among themselves in terms of exchanging threat intelligence and security metrics.

The SAIDL approach incorporates a looped security architecture within the framework, whereby threat intelligence is consistently provided from the runtime monitoring stage back to the preceding stages of the lifecycle. Fig. 1 shows how the SAIDL security approach is created. SAIDL can be represented by the operation of four modules: inputs, security operations, outputs, and feedback. The key inputs for SAIDL include AI training and validation data, model artifacts and parameters, training configurations, deployment configurations, runtime events, security alerts, and incidents data. Such inputs are sequentially and collaboratively processed by the six security layers. The Data Security Layer is responsible for validation and protection of data, the Model Security Layer for protection of model artifacts and parameters, the Training Security Layer for application of security mechanisms for training and adversarial robustness, the Deployment Security Layer for the protection of deployed AI service and its interfaces, the Runtime Monitoring Layer for detection of abnormal behavior and attacks, and the Incident Response & Recovery Layer for containment, recovery, and restoration actions. The outputs include validated data, protected model artifacts, security-enhanced models, secure deployment configurations, runtime threat alerts, mitigation actions, and restored states of the system. Information about security events and metrics produced during runtime and incident response is fed into the Threat Intelligence Feedback Engine that distributes appropriate information to the required earlier layers.

3.1.1 Data Security Layer

Data Security Layer represents the fundamental building block of the SAIDL approach. As a result of the fact that the majority of AML attacks occur at the level of collecting and preprocessing data, the necessity of providing security guarantees on the level of data prior to AI model training becomes evident. Such a layer performs data validation and source identification, as well as anomalies and any possible signs of poisoning attempts introduced by an attacker. In order to filter out any suspicious data, statistics-based tests are used. Also, encryption technologies may be utilized to maintain data confidentiality. In doing so, the Data Security Layer provides a basis for further operations on the basis of which other building blocks of the SAIDL concept may operate, thus preventing potential attacks at the very beginning of the process. Thus, the Data Security Layer reduces the likelihood that compromised, manipulated, or unauthorized data will propagate into subsequent model-development and training stages.

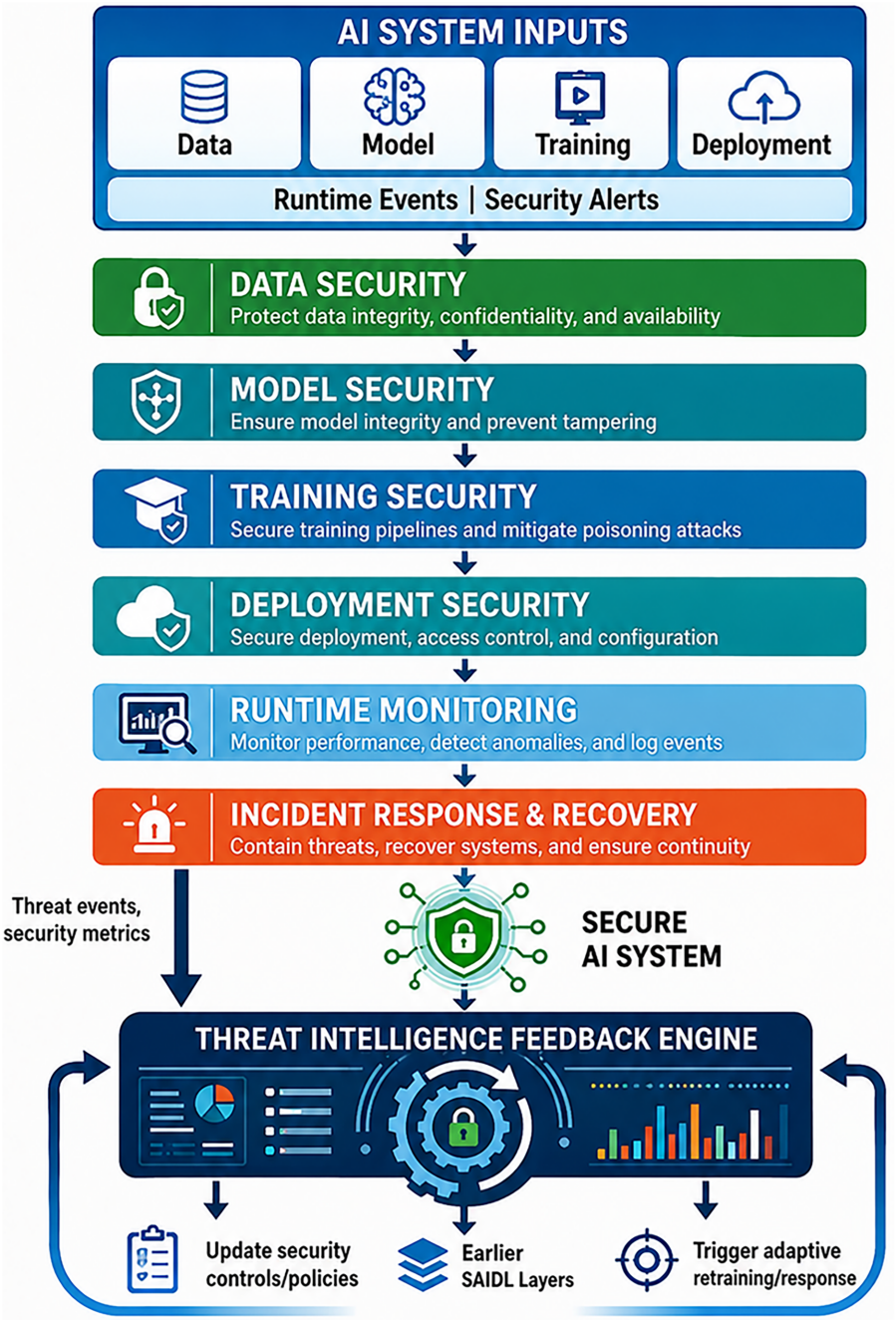


Figure 1: Proposed SAIDL architecture.

3.1.2 Model Security Layer

Model Security Layer is created to ensure the confidentiality, integrity, and ownership of trained AI models. With the development of modern AI technology, there are more cases of attacks from malicious actors who want to steal AI models and their parameters and use them for purposes not meant by their creators, as well as change the parameters to influence decision-making processes in a specific manner. Thus, to address these challenges, the framework uses differential privacy, model protection based on cryptographic principles, model watermarking, and integrity verification. First, the former makes membership inference

or model inversion harder, while digital watermarking allows verifying the ownership of the AI model when someone tries to copy its code or design. At the same time, cryptographic hashing and integrity verification help monitor the model parameters' stability and changes. Thus, the Model Security Layer reduces the risk of model theft, unauthorized modification, tampering, and compromise of model integrity before and during deployment.

3.1.3 Training Security Layer

The Training Security Layer is concerned with providing protection for the learning of the model from any adversarial manipulations and unapproved access. In the training process of machine learning, attackers may try to manipulate the training behavior of the model using poisoned training data, attacks against training parameters, or exploiting computing devices used in training. The measures that can be used to protect machine learning systems during their training phase include adversarial training, secure training environment, role-based access control, and audit logging. The adversarial training strategy involves including adversarial samples in the training process to make the trained models resistant to potential evasions. The secure training environment prevents attacks targeting computing devices, and the access controls prohibit malicious access to training processes. Thus, the Training Security Layer improves model robustness against adversarial manipulation and reduces the likelihood that poisoned or malicious training information will be incorporated into the resulting model.

Adaptive Adversarial Retraining: Adversarial training does not happen continuously but rather is carried out adaptively in case of multiple adversarial attacks being detected or the threat score exceeding a pre-set threshold of retraining (τ_t). New adversarial samples, injection attempts, cases of data poisoning, and others are all collected by the Threat Intelligence Feedback Engine and added to the training dataset. The new dataset is used for retraining or fine-tuning of the AI model in order to improve its robustness against novel attacks without increasing retraining costs unnecessarily.

3.1.4 Deployment Security Layer

After successful development of the model, the Deployment Security Layer ensures that the AI systems continue to be secured even when in use. The AI models that have been deployed through cloud infrastructure, application programming interface (API), edge devices, or enterprise applications are regularly subject to attempts by unauthorized parties to gain access and use. This layer includes several types of security measures like API gateway security, authentication, authorization, secure containerization, and zero-trust communication. The secure containers isolate the AI services from any outside applications, and at the same time, authentication and authorization ensure that the access is allowed only by authorized entities. In addition, encrypted communication prevents any interference with inference requests and outputs. Thus, the Deployment Security Layer reduces the exposure of AI services and model interfaces to unauthorized access, exploitation, and model extraction attempts.

3.1.5 Runtime Monitoring Layer

Runtime Monitoring is a layer that enables real-time monitoring of the AI system behavior while it is running in practice. Considering that the nature of AI-based cyber-attacks is highly volatile and unpredictable, static controls are not enough to uncover new security threats. For that reason, anomaly detection, behavioral analysis, monitoring in real-time, and the integration of threat intelligence sources have been employed to discover attacks at the point of their occurrence. Machine learning-enabled monitoring engines continuously compare the existing system behaviors against the baseline one in order to detect any

abnormalities in system functioning, which can hint at either an adversarial attack, the use case abuse of the model, or prompt injections. In addition to that, threat intelligence feeds have been added in order to gain situational awareness regarding emerging attack techniques. The runtime monitoring results are further utilized to generate adaptive retraining datasets and update security policies through the Threat Intelligence Feedback Engine, thereby improving resilience against previously unseen AI-enabled cyberattacks. Thus, the Runtime Monitoring Layer provides continuous detection of abnormal behaviour and emerging attacks and generates threat information that can be used by the response and feedback mechanisms.

3.1.6 Incident Response and Recovery Layer

Incident Response and Recovery Layer is the last defense component of the SAIDL approach and is designed to ensure the resilience of the operations after any security breaches. The detection of a threat activates the mechanisms that automatically initiate the isolation process in order to stop the spreading of an attack on the system components. After that, a recovery procedure is conducted to restore trusted copies of the models saved in the storage while keeping the operations up and running. Besides, the digital forensic analysis is conducted to investigate the sources of the attack and to evaluate its impact on the system operations. The security reports and incident logs are also provided by this layer. Thus, the Incident Response & Recovery Layer limits the impact of successful attacks through containment, recovery, and restoration of trusted AI system states.

Decision Making Criteria for Rollback: The suggested SAIDL scheme triggers the rollback process whenever the accumulated threat score produced by the Threat Intelligence Feedback Engine surpasses a certain predetermined threshold (τ_r). Rollback also takes place when the integrity verification mechanism identifies any modifications that have been made to the existing model without proper authorization. Moreover, the decision for rollback also becomes necessary when the Runtime Monitoring Layer detects sustained attacks with high severity, policy violations, and deterioration in the performance of the model that crosses a tolerable threshold.

3.1.7 Threat-to-Layer Mapping in SAIDL

In order to provide a clear depiction on how the suggested SAIDL framework prevents AI-driven attacks in the entire cycle, Table 2 shows the association between each kind of major attack and the relevant security layer that takes care of detecting, preventing, mitigating, and recovering from such attacks. It is clear from the table that an approach towards layered security is employed within the framework.

Table 2: Mapping of AI-enabled cyber threats to SAIDL security layers.

AI Security Threat	Primary Security Layer	Security Mechanism
Data Poisoning	Data Security Layer (L1)	Data validation, anomaly detection, provenance verification
Backdoor Injection	Data Security Layer (L1), Training Security Layer (L3)	Data inspection, adversarial training
Model Theft	Model Security Layer (L2)	Model encryption, watermarking, access protection
Membership Inference	Model Security Layer (L2)	Differential privacy

(Continued)

Table 2 (continued)

AI Security Threat	Primary Security Layer	Security Mechanism
Model Inversion	Model Security Layer (L2)	Differential privacy, secure model access
Adversarial Evasion	Training Security Layer (L3)	Adversarial training, robustness enhancement
Unauthorized Model Access	Deployment Security Layer (L4)	Authentication and authorization
API Exploitation	Deployment Security Layer (L4)	API gateway protection, secure communication
Prompt Injection	Runtime Monitoring Layer (L5)	Prompt monitoring, anomaly detection
LLM Jailbreak Attacks	Runtime Monitoring Layer (L5)	Behavioral analytics, policy enforcement
AI Service Abuse	Runtime Monitoring Layer (L5)	Continuous monitoring and threat intelligence
System Compromise	Incident Response Layer (L6)	Containment and recovery procedures
Persistent Threat Activity	Incident Response Layer (L6)	Forensic analysis and remediation

The threat-to-layer mapping clearly confirms the fact that several layers of security can be used to reduce the risk posed by a particular attack type. The above approach helps develop an architecture of defense in depth, enhances the ability of the AI system to withstand potential cyber-attacks.

3.1.8 Inter-Layer Coordination and Threat Intelligence Feedback

While each layer fulfills a different role during the SAIDL process, the proposed architecture is built to allow collaboration between the layers through information exchange. Security events produced by the layer do not act independently; rather, they are transmitted to the next layers by means of the Threat Intelligence Feedback Engine. As such, collaborative security actions and adaptability are possible at all times for the entirety of the AI development lifecycle. To illustrate, when the Data Security Layer identifies unusual distribution in data or data-poisoning attempts, the threat intelligence is sent to the Training Security Layer to reinforce adversarial training and data validation policy for the training phase. In a similar fashion, should the Runtime Monitoring Layer identify prompt injection attacks, unusual inference behavior, and model misuse, the information will be sent to the Incident Response and Recovery Layer for further actions to take place, including containment, investigation, and recovery operations. At the same time, the threat intelligence will be sent back to the Data Security, Model Security, and Training Security Layers for future adjustments to be made accordingly. This ongoing sharing of threat intelligence helps turn the suggested SAIDL into not just a system of individual security controls but an integrated security architecture that can continually improve its defense capabilities against constantly evolving AI-powered cyber threats.

3.2 Mathematical Formulation of the SAIDL

SAIDL as presented in this paper is mathematically defined as a process of security transformation whereby attacks enabled by AI are reduced through six security layers. Each layer offers a certain level of reduction in the attack impact, thus contributing to the overall security of the AI environment. A Threat Intelligence Feedback Engine evaluates the accumulated threat score from the sum of the alerts from all the security layers. The threat score is then passed on to the specific layer using certain policies for update. If the accumulated threat score goes beyond a certain threshold value, then certain adaptive mitigation measures are taken, which include retraining with the newly acquired attacks, updating anomaly detectors, enhancing access control policies, and securing deployment.

3.2.1 Initial Threat State

Let the initial threat state be defined as:

$$T_0 = \sum_{j=1}^{M} \omega_j A_j$$

where:

- T_0 = aggregate threat level before protection
- A_j = severity of attack category j
- ω_j = importance weight of attack category j
- M = total number of attack categories

The types of attacks are classified into: data poisoning attacks, model theft attacks, prompt injection attacks, adversarial evasion attacks, backdoor attacks, membership inference attacks, and model inversion attacks. Value of T_0 is the total cyber threat pressure exerted on the AI system.

3.2.2 Layer Mitigation Function

The mitigation capability of the security layer i is defined as:

$$\mu_i = \frac{\sum_{k=1}^{N_i} \alpha_k C_{ik}}{N_i}$$

where:

- μ_i = mitigation capability of the layer i
- C_{ik} = effectiveness of control k in layer i
- α_k = importance coefficient
- N_i = number of security controls in layer i

Examples of controls include poisoning detection, differential privacy, adversarial training, API authentication, anomaly detection, and recovery orchestration.

The mitigation capability is normalized as:

$$0 \leq \mu_i \leq 1$$

where:

$$\mu_i = 1$$

indicates complete protection.

Mitigation Capability Estimation: Mitigation Capability Coefficient for each security layer is based on how effective the applied security controls are. E_{ik} indicates effectiveness of the security control k for layer i in percentage terms as the reduction in impact of successful attacks in the experiment.

The mitigation capability of layer i is computed as

$$\mu_i = \frac{\sum_{k=1}^{N_i} \alpha_k E_{ik}}{\sum_{k=1}^{N_i} \alpha_k}$$

where:

- μ_i = mitigation capability of security layer i;
- E_{ik} = estimated effectiveness of control k in layer i;
- α_k = weighting coefficient representing the relative importance of control k;
- N_i = total number of security controls deployed within layer i.

The effectiveness values were estimated from representative attack scenarios and published benchmark studies. For each representative attack scenario, the attack success rate before and after applying the corresponding security control was used to estimate the control effectiveness, based on simulation-derived or published quantitative evidence.

The effectiveness of each security control is calculated as

$$E_{ik} = \frac{ASR_{\text{before}} - ASR_{\text{after}}}{ASR_{\text{before}}}$$

where:

- ASR_{before} = attack success rate before applying the mitigation mechanism;
- ASR_{after} = attack success rate after applying the mitigation mechanism.

The resulting effectiveness value E_{ik} ranges from 0 to 1, where a value closer to 1 indicates stronger attack mitigation performance. Consequently, the layer mitigation coefficient μ_i represents the weighted average protection capability provided by all security controls within the corresponding security layer.

This methodology ensures that mitigation capability values are derived from representative attack scenarios, published quantitative security evaluations, and the analytical formulation of the proposed SAIDL framework rather than from subjective or manually assigned estimates.

3.2.3 Security State Transition Model

The residual threat after passing through the layer i is given by:

$$T_i = T_{i-1}(1 - \mu_i)$$

where:

- T_i = residual threat after layer i
- T_{i-1} = threat entering layer i
- μ_i = mitigation capability of layer i

For the six-layer architecture:

$$T_1 = T_0(1 - \mu_1)$$

$$T_2 = T_1(1 - \mu_2)$$
$$T_3 = T_2(1 - \mu_3)$$
$$T_4 = T_3(1 - \mu_4)$$
$$T_5 = T_4(1 - \mu_5)$$
$$T_6 = T_5(1 - \mu_6)$$

Thus, the final residual threat becomes:

$$T_{Final} = T_0 \prod_{i=1}^{6}(1 - \mu_i)$$

This formulation demonstrates how threats are progressively reduced throughout the lifecycle.

3.2.4 Layer Contribution Analysis

The security contribution of each layer is:

$$SC_i = T_{i-1} - T_i$$

where:

- SC_i = security contribution of layer i

The total threat reduction is:

$$SC_{Total} = \sum_{i=1}^{6} SC_i$$

A higher value indicates greater protection effectiveness.

3.2.5 Security Resilience Function

The resilience score generated by layer i is defined as:

$$R_i = \gamma_1 P_i + \gamma_2 D_i + \gamma_3 RS_i + \gamma_4 RC_i$$

where:

- R_i = resilience score of layer i
- P_i = prevention capability
- D_i = detection capability
- RS_i = response capability
- RC_i = recovery capability
- $\gamma_1, \gamma_2, \gamma_3, \gamma_4$ = weighting factors

subject to:

$$\gamma_1 + \gamma_2 + \gamma_3 + \gamma_4 = 1$$

The overall lifecycle resilience becomes:

$$R_{SAIDL} = \frac{1}{6}\sum_{i=1}^{6} R_i$$

3.2.6 Adaptive Threat Intelligence Feedback

The Runtime Monitoring Layer and Incident Response Layer continuously generate threat intelligence.

The feedback signal is defined as:

$$F = \lambda_1 D_A + \lambda_2 D_N + \lambda_3 D_P$$

where:

- F = threat intelligence feedback score
- D_A = detected attack events
- D_N = detected anomalies
- D_P = detected policy violations
- $\lambda_1, \lambda_2, \lambda_3$ = weighting coefficients

The adaptive update rule is:

$$\mu_i^{new} = \mu_i + \eta F$$

where:

- η = learning coefficient

This equation models the adaptive nature of the proposed framework.

Let Q denote the cumulative threat score generated by aggregating threat events across all security layers. The framework defines two operational thresholds:

- **Retraining Threshold** (τ_t): If $Q \geq \tau_t$, adaptive adversarial retraining is initiated.
- **Rollback Threshold** (τ_r): If $Q \geq \tau_r$, or if model integrity verification fails, the framework initiates rollback to the latest trusted model version.

These thresholds provide clear operational rules for adaptive retraining and model rollback within the SAIDL framework.

3.2.7 Secure AI Security Index

The final security state of the AI system is expressed as:

$$SSI = \frac{SC_{Total} + R_{SAIDL} + F}{1 + T_{Final}}$$

where:

- SSI = Secure AI Security Index
- SC_{Total} = cumulative threat reduction
- R_{SAIDL} = lifecycle resilience score
- F = adaptive feedback capability
- T_{Final} = final residual threat

The SSI evaluates the security performance of the AI system holistically, taking into consideration the interplay of three dimensions of evaluation: the security contribution (SC_{Total}), lifecycle resilience (R_{SAIDL}), and adaptive threat intelligence capability (F). The security contribution refers to the total threat reduction achieved in the course of implementing the six-level SAIDL approach. The score assigned for the lifetime resilience factor refers to the level of protection against cyber-attacks offered throughout the entire lifecycle of an AI system, including the detection of attacks and recovery from them. The adaptive threat intelligence factor is associated with the constant learning and security adaptation capabilities in relation to previously experienced threat incidents. Thus, the inclusion of the final residual threat level (T_{Final}) in the denominator correctly assesses the negative effects of having higher residual risk levels. As a result, a higher SSI score denotes better security performance, higher resilience, and adaptive response capabilities.

Methodological Justification: The chosen security metrics comprise complementary dimensions of AI security: Attack Detection Rate (ADR), False Positive Rate (FPR), Threat Reduction Rate (TRR), Security Layer Contribution (SCS), and Adversarial Robustness Score (ARS). The SSI is comprised of these dimensions together with adaptive threat intelligence and residual threat to produce a general indicator of security performance over the entire lifecycle. The metric definition is based on the sequential security state transformation, where each security layer decreases the residual threat.

Parameter Validation: The weights for the parameters are normalized such that they correspond to the significance of the respective security dimension. In this study, there was no formal validation using an expert panel. The formulation's robustness was tested through sensitivity and ablation studies. The sensitivity test indicated that the value of SSI was 0.94 when the average mitigation capability was 0.95 compared to the initial values of 0.71 and 0.60, respectively. Changes in resilience weights did not change SSI by more than 5%. The ablation study focused on the contribution of individual layers in the SAIDL framework.

Parameter Normalization and SSI Calculation: The parameters used in the SAIDL mathematical model were normalized to keep consistency among all the security layers and dimensions of evaluation. Values of control effectiveness were acquired through representative attacks and relevant research from cybersecurity and AI security literature based on the relative reduction in attack effectiveness before and after mitigation. Mitigation coefficients were then utilized to compute residual threat, security contribution, lifecycle resilience, adaptive threat intelligence, and finally the SSI. Sensitivity analysis was done by varying the average mitigation capability value from 0.60 to 0.95, with all other parameters kept constant. The calculated SSI values ranged from 0.71 to 0.94. As SAIDL is assessed in terms of a generic lifecycle security architecture, not an operational realization of an AI security system, the parameter values should be used only for analytical and comparative purposes at the framework level. For the framework-level analytical evaluation, these parameters were treated as normalized model coefficients, while the sensitivity analysis was used to examine the effect of changes in mitigation capability and resilience weighting on the resulting SSI. The reported SSI values should therefore be interpreted as analytical indicators of the proposed framework rather than as experimentally measured values from a specific operational AI system.

4 Experimental Setup and Evaluation Parameters

Public benchmark datasets have been chosen for simulation. This section details the experimental setup for evaluation of the proposed SAIDL framework. The evaluation is centered on measuring the ability of the proposed framework to provide protection against AI-enabled cyberattacks, minimize threat propagation, and increase resilience and security performance in the process. Public cybersecurity and AML benchmark datasets are used in order to enable reproducibility and transparency. For the purpose of analyzing and evaluating the effectiveness of the proposed SAIDL framework, public cybersecurity and benchmark datasets are employed as the source for creating simulations that are based on various threats that the framework

should guard against. The purpose of the evaluation is not to demonstrate real-world results from an operational framework implementation but to showcase the expected security performance gains.

Scope of Evaluation: The scope of evaluation of the SAIDL framework is based on assessing its suitability as a generic lifecycle security architecture and not through its implementation in a particular AI-based system. To perform this evaluation, different attack scenarios for AI were generated from the publicly available cybersecurity and AML benchmark datasets like CIC-IDS2017, CIC-DDoS2019, NSL-KDD, ART, and PromptBench. These benchmark datasets constitute different attack surfaces, such as network intrusion detection attacks, AML attacks, and LLM prompt injection attacks.

Evaluation Process: For each threat scenario, the following before and after evaluation process was done to mitigate the scenario. Initially, the attack scenario was evaluated without the SAIDL security controls to determine the initial state of the attack. After that, the attack scenario was mitigated using the corresponding SAIDL security mechanisms and then the attack scenario was re-evaluated to determine the decrease in the threat scenario's effectiveness. After determining the threat reduction and layer mitigation capabilities of the SAIDL mechanisms, the values were incorporated into the mathematical formula for SAIDL to find out the contribution towards security, lifecycle resilience, adaptive threat intelligence capability, and the SSI. Various cyberattack scenarios were evaluated using publicly available datasets, as illustrated in Table 3.

Table 3: Cyberattack scenarios using publicly available datasets.

Dataset	Assessment Purpose
CIC-IDS2017 [26]	Threat detection, runtime monitoring, and incident response evaluation
CIC-DDoS2019 [27]	Threat propagation and resilience assessment
NSL-KDD [28]	Benchmark intrusion detection evaluation
ART Adversarial Datasets [29]	Adversarial robustness and AML resilience assessment
PromptBench [30]	Prompt injection detection and LLM security evaluation

The different benchmark datasets correspond to each element of the proposed threat model. CIC-IDS2017, CIC-DDoS2019, and NSL-KDD benchmarks were used for assessing the ability of detecting network-based threats and runtime monitoring. AML attacks, which include adversarial evasion and data poisoning, were evaluated using the ART benchmark. For assessing prompt injection attacks on large language models applications, we used PromptBench. Overall, these benchmark datasets cover representative attack surfaces within the proposed SAIDL framework. Attack scenarios used for evaluation include five representative examples: (i) data poisoning attacks that poison training datasets, (ii) adversarial evasion attacks created by applying FGSM, PGD, and DeepFool methodologies, (iii) model theft attacks to steal the model deployed, (iv) prompt injection attacks to poison large language model-based systems, and (v) backdoor attacks through which hidden triggers initiate any malicious activity while making inference.

Reproducibility and Execution Details: The benchmark datasets were used to develop representative attack scenarios corresponding to the security layers in the proposed SAIDL framework. CIC-IDS2017, CIC-DDoS2019, and NSL-KDD datasets were used to represent network threats and runtime monitoring, whereas the ART datasets were used to represent adversarial machine learning attacks, including data poisoning and adversarial evasion. PromptBench was used to represent LLM prompt-injection attacks. The evaluation followed a before-and-after approach, in which each representative attack scenario was first evaluated without the corresponding SAIDL security control to determine the initial threat state and then evaluated after applying the appropriate security mechanism to determine the resulting reduction in threat.

These threat-reduction values were used as inputs to the mathematical model presented in Section 3.2 to compute the security contribution, resilience, adaptive threat intelligence capability, and SSI.

Dataset and Implementation Scope: The benchmark datasets were used as publicly available sources for constructing representative framework-level attack scenarios rather than for training and testing a single AI model. Consequently, the evaluation does not define a model-specific training/test partition or model-dependent preprocessing pipeline. No software, hardware, or random-seed configuration is treated as a universal experimental requirement because the evaluation is based on the analytical SAIDL formulation and representative simulation scenarios rather than an end-to-end implementation of a particular AI system.

Since SAIDL is analyzed as a generic lifecycle security architecture rather than as an implementation of a particular AI system, no single model architecture, model-specific train/test split, or hardware configuration is prescribed. The benchmark datasets are used to generate representative simulation cases rather than to train and evaluate a single machine learning model. Therefore, the reported results should be interpreted as analytical and simulation-based metrics rather than measurements obtained from a real-world AI deployment.

4.1 Threat Model and Attack Scenario Configuration

The evaluation of the proposed SAIDL framework was conducted using a representative AI threat model that includes potential attacks across various stages of the AI lifecycle. The threat model posits various adversaries with capabilities ranging from external actors seeking to exploit AI services to malicious users tampering with the training or inference process. The goal of the adversary is to undermine the security of the AI through attacks on various security layers. Five representative AI threats were used for evaluation:

- Data Poisoning: It was assumed that malicious samples were injected into the training set to influence model behavior. This threat was evaluated by using representative poisoning attacks from AML benchmark datasets.
- Adversarial Evasion: Adversarial samples were created using representative evasion attacks such as Fast Gradient Sign Method (FGSM), Projected Gradient Descent (PGD) and DeepFool attacks.
- Model Theft: The model theft attacks were conducted by considering that multiple API calls would be made to recreate the behavior of the model and predict the decision boundary.
- Prompt Injection: The prompt injection attack was assessed using PromptBench and publicly available scenarios of prompt manipulation attacks that try to circumvent the safety mechanism or cause change in the behavior of the LLM-based model.
- Backdoor Attacks: The backdoor attacks were simulated by creating hidden trigger patterns in the training phase and then checking if the model generates harmful output while these triggers appear in the inference phase.

Attack Scenarios and Control Configurations: The five representative attacks have been defined as simulation scenarios at the framework level rather than as operational attacks on a single AI model. Data poisoning has been modeled through malicious sample injection into the training dataset, adversarial evasion through representative FGSM, PGD, and DeepFool attacks, model theft through repeated API-query-based attempts to emulate model behavior, prompt injection through representative PromptBench and publicly available prompt-manipulation scenarios, and backdoor attacks through hidden trigger patterns introduced during training and evaluated during inference. For each scenario, the corresponding SAIDL security control was evaluated before and after mitigation, and the resulting change in threat effectiveness was used as an input to the mathematical model. Since the study considers SAIDL as a generic security architecture spanning the AI development lifecycle rather than as an implementation for a specific operational AI model,

attack-specific parameters such as poisoning ratios, perturbation budgets, attack iterations, model-query budgets, trigger specifications, and model-specific control configurations were not defined as universal experimental parameters.

Attack-to-Control Traceability: The attack scenarios were evaluated against the corresponding SAIDL security controls identified in the threat-to-layer mapping in Table 2. Data poisoning was associated primarily with data validation, anomaly detection, and provenance verification; adversarial evasion with adversarial training and robustness enhancement; model theft with model protection and access controls; prompt injection with prompt monitoring and anomaly detection; and backdoor attacks with data inspection and adversarial training. The before-and-after threat-reduction values associated with these control mappings were used as inputs to the mitigation formulation in Section 3.2. Because the evaluation is framework-level rather than an operational attack campaign against a specific AI model, attack-specific numerical parameters were not defined as universal experimental settings.

The SAIDL framework was evaluated based on its capability to detect, mitigate, and recover from the identified attacks using ADR, FPR, TRR, SCS, ARS, and SSI.

Dataset and Attack Scenario Configuration

For the purposes of this evaluation, publicly available benchmark datasets were used to generate representative attack scenarios to evaluate the proposed SAIDL architecture. These benchmark datasets include CIC-IDS2017, CIC-DDoS2019, NSL-KDD, ART, and PromptBench datasets. CIC-IDS2017, CIC-DDoS2019, and NSL-KDD datasets are used to represent network-based threats and runtime monitoring, while ART and PromptBench datasets are used to represent adversarial machine learning and LLM prompt-injection attacks, respectively.

The five representative attack types considered in the evaluation are data poisoning, adversarial evasion, model theft, prompt injection, and backdoor attacks. For adversarial evasion scenarios, FGSM, PGD, and DeepFool attack types are considered. For prompt injection, the PromptBench dataset is used to represent representative attacks. The representative attack scenarios are evaluated before and after the deployment of relevant SAIDL security control mechanisms. Threat reduction and mitigation values obtained from these evaluations were used as inputs to the mathematical formulation outlined in Section 3.2.

Since the current research assesses SAIDL as a generic lifecycle security framework and not as an end-to-end AI system, the benchmark datasets are employed to create realistic simulation cases and not for making claims about their performance in real-life deployment situations. Accordingly, numerical results derived from the SAIDL mathematical formulation are interpreted as analytical or simulation-based indicators of framework performance. Where specific attack or security-control parameters were not independently implemented and measured in an operational AI system, the corresponding results are not interpreted as direct experimental measurements.

4.2 Baseline Architectures for Comparison

The suggested SAIDL framework was evaluated relative to four representative architectures:

- Conventional AI Security Pipeline
- NIST AI Risk Management Framework (AI RMF)
- MITRE ATLAS-Based Security Model
- OWASP LLM Security Framework

The comparison enables assessment of lifecycle-wide security improvements introduced by the proposed architecture.

4.3 Evaluation Parameters

To comprehensively evaluate framework performance, multiple security, robustness, and resilience metrics are considered.

4.3.1 Attack Detection Rate (ADR)

ADR measures the proportion of successfully detected attacks.

$$ADR = \frac{TP}{TP + FN}$$

where, TP = True Positives and FN = False Negatives.

4.3.2 False Positive Rate (FPR)

FPR quantifies the proportion of benign events incorrectly classified as attacks.

$$FPR = \frac{FP}{FP + TN}$$

where, FP = False Positives and TN = True Negatives.

4.3.3 Threat Reduction Ratio (TRR)

TRR measures the effectiveness of the proposed framework in reducing threat levels across the lifecycle.

$$TRR = \frac{T_0 - T_{Final}}{T_0} \times 100$$

where, T_0 = Initial threat state and T_{Final} = Residual threat state after mitigation.

4.3.4 Security Contribution Score (SCS)

SCS evaluates the cumulative protection provided by all six security layers.

$$SCS = \sum_{i=1}^{6} SC_i$$

where SC_i = Security contribution of layer i.

4.3.5 AI Resilience Score (ARS)

ARS measures the four resilience dimensions of the proposed framework.

$$ARS = \frac{1}{6} \sum_{i=1}^{6} R_i$$

where R_i = Resilience contribution of layer i.

4.3.6 SAIDL Security Index (SSI)

SSI is the final performance indicator used for evaluating the overall effectiveness of the proposed framework.

$$SSI = \frac{SC_{Total} + R_{SAIDL} + F}{1 + T_{Final}}$$

where, SC_{Total} = Total security contribution of all layers and R_{SAIDL} = Overall resilience score of the SAIDL framework F = Framework effectiveness factor and T_{Final} = Residual threat state. A higher SSI value indicates stronger protection, greater resilience, and improved trustworthiness of the AI system.

4.3.7 Interpretation of SSI Values

Table 4 presents the interpretation of SSI values used for assessing overall AI security maturity and resilience.

Table 4: Interpretation of SSI values.

SSI Range	Security Level
SSI < 0.40	Vulnerable
0.40 ≤ SSI < 0.60	Moderate
0.60 ≤ SSI < 0.80	Strong
SSI ≥ 0.80	Highly Secure

The SSI categories provide a practical interpretation of overall AI security posture. Lower values indicate higher residual risk and weaker resilience, whereas higher values reflect stronger lifecycle-wide protection and adaptive security capability.

4.4 Performance Evaluation Strategy

For testing the capabilities of the framework for performing under unfavorable conditions, the framework was tested under increasing levels of attack intensity. Comparison analysis was carried out against all baseline frameworks based on the evaluation metrics stated earlier. The resulting analytical and simulation-based values were used to evaluate the potential contribution of the framework to attack mitigation, threat reduction, resilience, and overall security.

5 Framework Performance Evaluation and Analysis of Results

The following section presents the performance analysis of the proposed SAIDL framework against a range of attack scenarios involving AI-assisted cyber threats. The analysis aims to evaluate the capability of the framework to detect cyber threats, defend against attacks, prevent the spread of threats, enhance resilience, and improve the security of intelligent systems. For this evaluation, several attack scenarios were selected from publicly available datasets, including CIC-IDS2017, CIC-DDoS2019, NSL-KDD, the ART adversarial dataset, and PromptBench. The inclusion of ART and PromptBench extends the evaluation beyond conventional network-security datasets and provides coverage of modern AI-specific threats, including AML and LLM prompt-injection vulnerabilities.

5.1 Security Metrics Estimation and Result Provenance

The proposed SAIDL framework is evaluated as an architectural approach to cybersecurity assurance, and the reported results are based on representative simulation scenarios, published cybersecurity and AI-security studies, and the mathematical model of the SAIDL framework developed in Section 3.2. To maintain transparency in reporting, the numerical results are distinguished according to their source. Representative

attack scenarios constructed using publicly available benchmark datasets serve as the basis for assessing the considered threats and security controls. Where control-effectiveness values are obtained from published cybersecurity and AI-security studies, the corresponding references are provided to support these values. The obtained mitigation factors were used in further calculations within the mathematical framework presented in Section 3.2 to evaluate residual threats, security contributions, resilience, and the SSI. Thus, all results should be viewed as indicators of performance evaluation for the considered framework, but not metrics measured in a real-world scenario.

Accordingly, the results reported in this study should not be interpreted as independently replicated measurements from an end-to-end operational implementation of SAIDL. Simulation-derived results represent the outcomes of the considered representative attack scenarios, literature-derived values represent effectiveness information obtained from published studies, and analytical results represent quantities calculated using the proposed mathematical model.

5.2 Threat Detection Performance Analysis

Detection of threats becomes an important functionality within the designed framework, since timely detection of any malicious activities ensures that such threats do not spread to other lifecycle phases. Runtime Monitoring Layer constantly monitors network traffic, model behavior, prompt communications, and operation log files for any suspicious activity. The numerical results presented in the Table 5 is interpreted according to the result provenance described in Section 5.1.

Table 5: Threat detection performance of the proposed SAIDL framework.

Attack Type	Precision (%)	Recall (%)	F1-Score (%)	Detection Rate (%)
Data Poisoning	95.8	94.6	95.2	94.8
Adversarial Evasion	96.4	95.7	96.0	95.9
Model Theft	94.9	93.8	94.3	94.1
Prompt Injection	97.3	96.1	96.7	96.4
Backdoor Attack	95.2	94.5	94.8	94.7
Average	95.9	94.9	95.4	95.2

From Fig. 2, it is seen that there is a higher detection rate in the case of injection attacks, which is 96.4%, because of the use of behavioral analysis and prompt monitoring systems incorporated into the Runtime Monitoring Layer. Likewise, the detection rate in case of poisoning and backdoor attacks is effective because of the use of anomaly detection and data validation techniques at an early stage in the lifecycle. The 95.2% detection rate indicates the potential effectiveness of the proposed framework under the considered evaluation scenarios.

5.3 Attack Mitigation Effectiveness Analysis

In addition to identifying attacks, the proposed framework aims to reduce the effectiveness of adversarial attacks through the implementation of multiple security mechanisms. The experiments evaluate the effectiveness of the framework against different types of attacks by incorporating security measures across all six security layers.

From Fig. 3 below, it is apparent that there is a marked drop in attack success in all the different attack categories once the proposed framework is implemented. The adversarial evasion attacks have been highly affected, with their success dropping from 63.1% to 12.8%. This drop is attributed to the adversarial training

methods adopted in the Training Security Layer. On the other hand, model theft attacks are minimized using differential privacy, encryption, and model watermarking within the Model Security Layer.

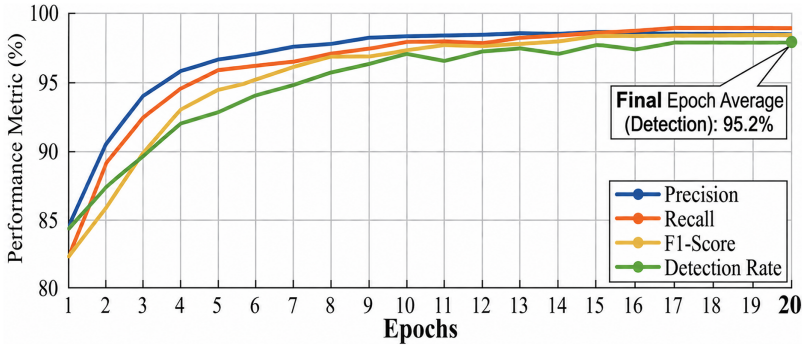


Figure 2: Performance metric evolution.

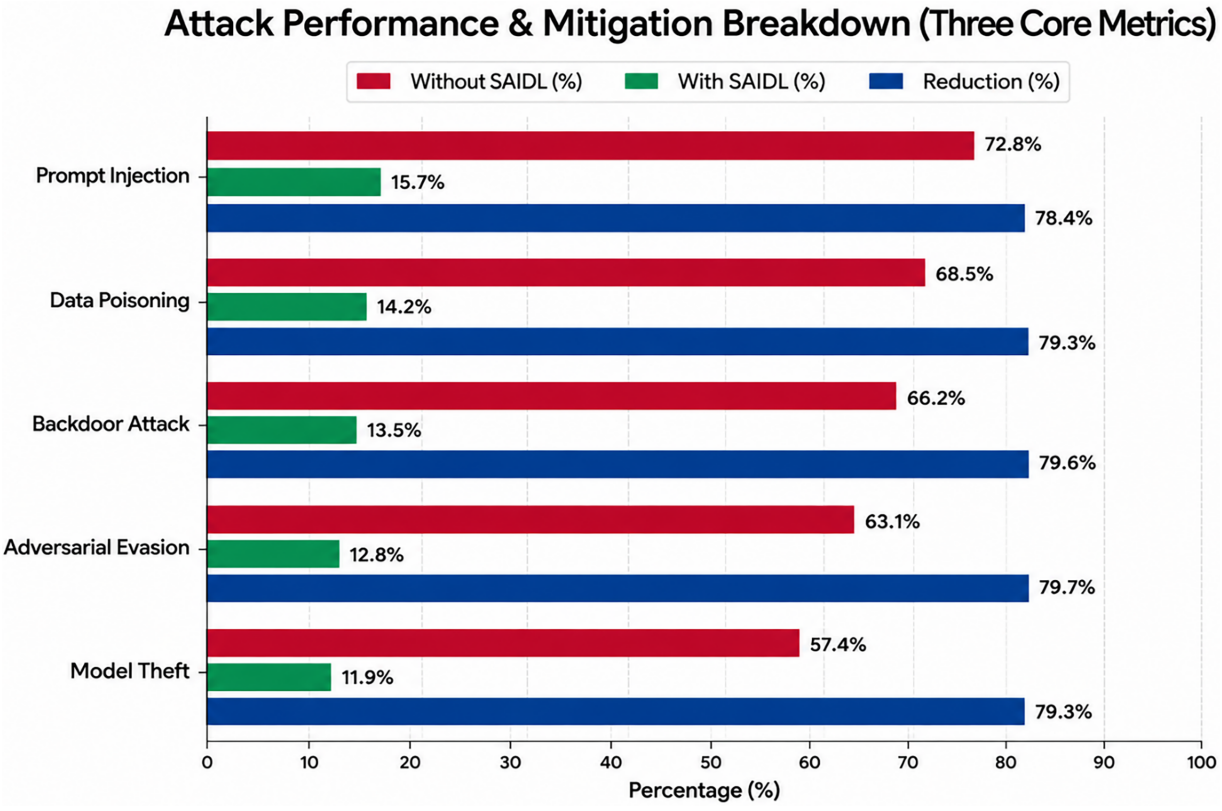


Figure 3: Attack success rate before and after SAIDL protection.

5.4 Layer-Wise Security Contribution Analysis

The contribution made by each layer of the security layers is analyzed to measure its significance towards the overall performance of protection capability. The Security Contribution Score (SCS) is determined by applying the formula discussed in Section 3.

From Fig. 4, the Runtime Monitoring Layer contributes the most towards security, with its score standing at 0.93. This illustrates that cybersecurity threat detection through real-time monitoring and

behavior analysis is very essential to improving overall security levels. Training Security Layer also makes a great contribution due to adversarial learning strategies to boost robustness. However, the Deployment Security Layer contributes the least among others; nevertheless, it is equally important to ensuring security when deployed.

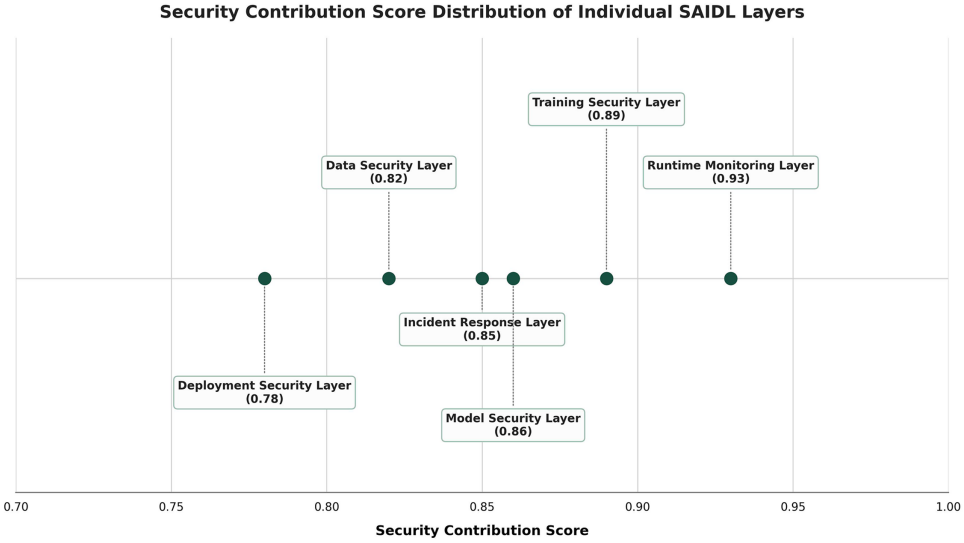


Figure 4: Security contribution score of individual SAIDL layers.

5.5 Threat Propagation Reduction Analysis

A key aim of this framework involves disrupting the propagation of threats across the entire lifecycle of AI services. The efficiency of the security layer across the lifecycle has been tested through the probability of threat propagation across two consecutive layers.

From Table 6, the results show that there is considerable improvement in threat propagation in all lifecycle stages. The largest reduction in attacks is observed when transitioning from the Data Security Layer to the Model Security Layer, primarily due to the poisoning detection and data validation processes. This result demonstrates the effectiveness of the defense-in-depth approach employed in the proposed framework.

Table 6: Threat propagation probability across lifecycle stages.

Lifecycle Transition	Without Protection	With SAIDL
Data → Model	0.72	0.21
Model → Training	0.69	0.19
Training → Deployment	0.64	0.17
Deployment → Runtime	0.58	0.15
Runtime → Incident Response	0.49	0.12

5.6 Sensitivity Analysis of SAIDL Parameters

The SAIDL framework has been subjected to sensitivity analysis to investigate the effects of different levels of security-control effectiveness and resilience on its performance. This analysis aims at assessing the effects that changes in the mitigation capability and resilience weighting factors have on T_{Final} and SSI. Description of Impact of Mitigation Capability on SSI. The SSI calculation process in this study involves

estimating residual threats using the mitigation-based threat model and integrating the residual threat with resilience measures in the framework as described in Section 4. To test the sensitivity of the model parameters, an average mitigation capability coefficient (μ_i) of six security layers has been varied between 0.60 and 0.95 while keeping other parameters constant, as shown in Table 7.

Table 7: Impact of mitigation capability on SSI.

Average Mitigation Capability (μ_i)	SSI
0.60	0.71
0.70	0.78
0.80	0.84
0.90	0.91
0.95	0.94

From the analysis of the results above, there is a clear positive correlation between the level of effectiveness of the mitigation measures and the total performance of the security systems. An increase in the average value of the mitigation measure from 0.60 to 0.95 results in an improvement of the SSI value from 0.71 to 0.94, giving a 32.4% increase. In assessing the robustness of the proposed framework, the weight factors for resilience of (γ_1, γ_2, γ_3, γ_4) representing the ability of the system to prevent, detect, respond, and recover were adjusted separately subject to the normalization condition.

$$\sum_{i=1}^{4} \gamma_i = 1$$

However, the recorded SSI scores showed only slight variation, with differences remaining within 5% across all configurations. This demonstrates that the presented framework does not overly rely on any one specific dimension of resilience. In turn, security performance depends on the cumulative impact of all four resilience dimensions: prevention, detection, reaction, and recovery. To quantify parameter influence, the normalized sensitivity coefficient was computed as

$$S_p = \frac{\Delta SSI/SSI}{\Delta p/p}$$

where θ represents the investigated parameter. (either μ_i or γ_i). From the results of the sensitivity analysis, we can conclude that the presented SSI score exhibits moderate sensitivity to mitigation capability but is relatively insensitive to small changes in the resilience weight parameters. In summary, from the presented sensitivity analysis results, we can conclude that the SAIDL framework proposed in our work is robust when using different parameter configurations. The increasing tendency of the SSI score with an increase in mitigation capability and insensitivity to slight perturbations in the resilience weights confirms that.

5.7 Ablation Analysis of SAIDL Layers

In order to examine the impact of individual layers on the overall security, ablation experiments were carried out with the removal of layers one-by-one while analyzing security metrics performance. According to the results of this analysis, it can be concluded that all six layers make a positive contribution to the security posture of the overall framework. Among the considered security layers, Runtime Monitoring (L5) and Incident Response (L6) have resulted in the most significant performance drop when disabled, which reflects their important role in the detection of attacks and subsequent mitigation. The Data Security (L1), Model

Security (L2), and Training Security (L3) layers contribute considerably to the improvement of attack success rate during the initial phases of the AI lifecycle. An additional layer of Deployment Security (L4) has been added for protecting the application from attacks using the approach of authentication and authorization. The above conclusions about the contribution of individual layers can be drawn from Fig. 5, showing the results of ablation experiments.

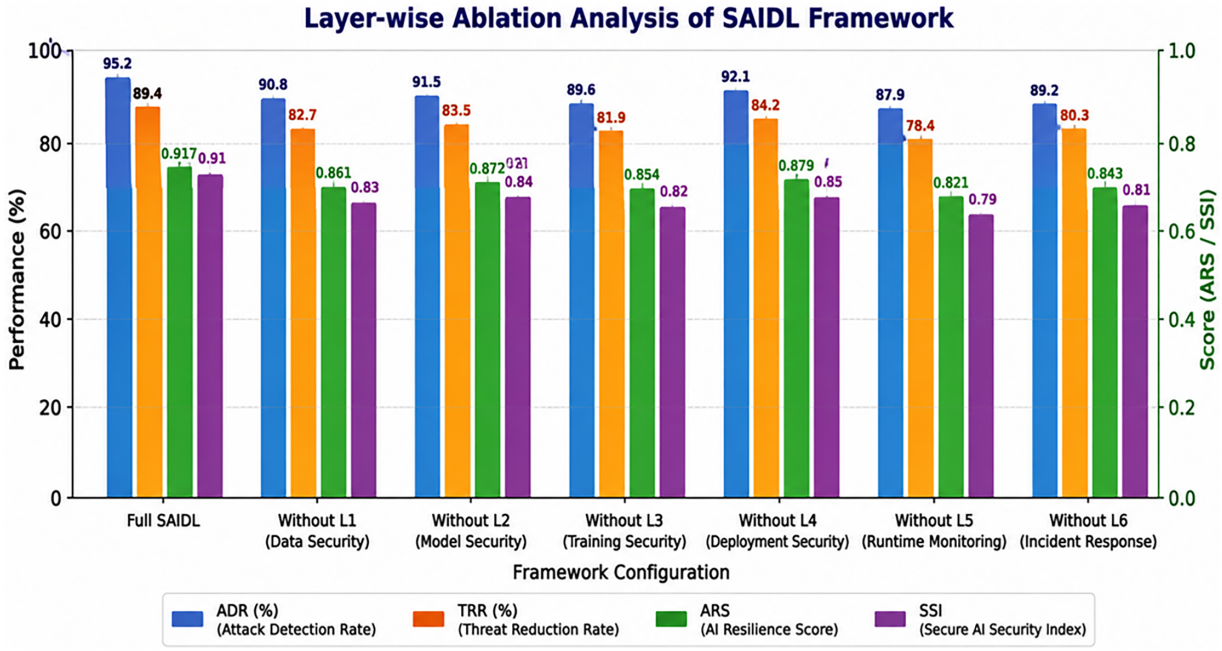


Figure 5: Ablation analysis of SAIDL layers.

Resilience refers to the capability of an AI system to safeguard itself against any potential cybersecurity threats by preventing them, detecting them, responding to them, and then recovering from them successfully. The developed framework is designed to integrate resilience capabilities across multiple lifecycle stages, with the objective of supporting operational continuity under security threats.

5.8 AI Resilience Assessment

From Fig. 6, the framework scores a total AI Resilience Score of 0.917. The capability of detection ranks number one owing to the presence of effective anomaly detection and runtime monitoring systems. Meanwhile, the capability of recovery ranks second thanks to rollback and recovery processes embedded in the Incident Response Layer.

Based on this comparison, the unique characteristic of SAIDL is that it encompasses the entire AI life cycle about security. The NIST AI RMF, the MITRE ATLAS, and the OWASP LLM security framework offer risk management, threat-driven, and application-level security, respectively. However, the SAIDL model capitalizes on the above security frameworks by arranging the relevant controls in a layered lifecycle and also connecting runtime threat monitoring with security updates. Moreover, the suggested mathematical model and the SSI give a single quantified approach to measure the impact of security. These baseline frameworks were selected because they represent complementary approaches to AI security, including conventional lifecycle security, AI risk management, threat-informed security, and LLM-specific security guidance.

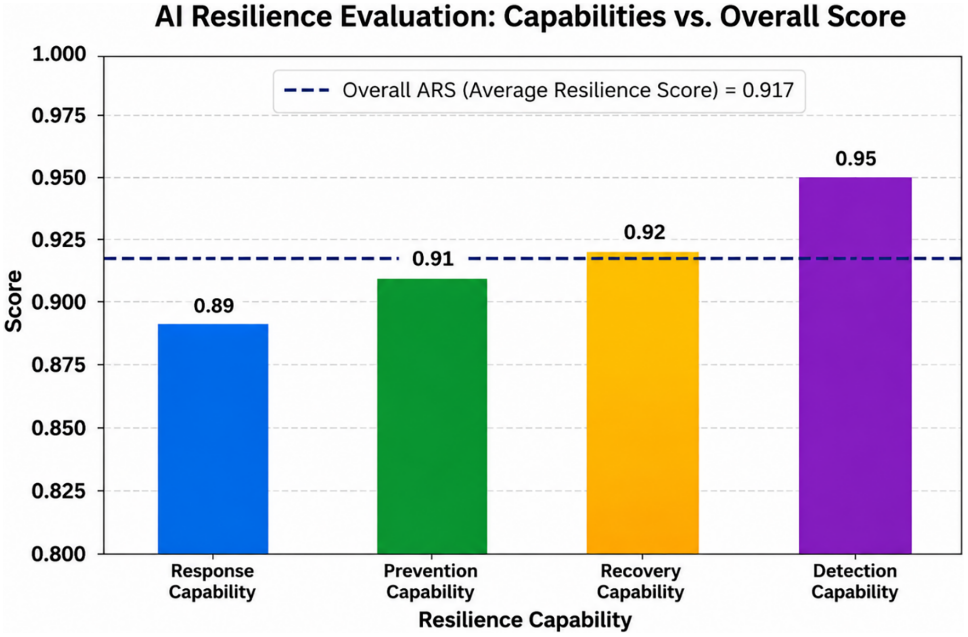


Figure 6: AI resilience assessment.

Statistical Consideration: As the current assessment is both an analysis and a simulation, and the effectiveness figures in the study are derived from representative benchmark data and literature, the current SSI and framework-level security measures are viewed as comparative assessments and not independent replicated experimental results. Therefore, no claims are made about the statistical significance of the current framework-level results. The sensitivity and ablation studies are utilized in order to analyze the robustness and impact of the suggested model parameters and security levels.

5.9 Comparative Framework Analysis

To examine the applicability of the proposed framework, the analysis was made between the existing AI security frameworks. From Table 8, the comparison is done according to the coverage of lifecycle protection, detection of threats, resilience capabilities, and adaptive security feedback mechanisms provided by the frameworks. Low, Medium, and High indicate how well the framework supports this category based on the available literature.

Table 8: Conceptual positioning of SAIDL relative to existing AI security frameworks.

Framework	Lifecycle Coverage	Threat Detection	Resilience Support	Adaptive Feedback
Conventional AI Pipeline	Low	Medium	Low	No
NIST AI RMF	Medium	Medium	Medium	No
MITRE ATLAS	Medium	High	Low	No
OWASP LLM Security	Low	High	Low	No
SAIDL (Proposed)	High	High	High	Yes

According to Table 8, the proposed SAIDL framework provides broader explicit coverage across the selected architectural dimensions than the other approaches. While existing frameworks primarily focus on governance, risk management, threat knowledge, or application-level security guidance, SAIDL integrates security controls across the AI lifecycle. In addition, SAIDL incorporates adaptive threat-intelligence feedback, quantitative security modelling, and security assessment within a unified architectural framework. The contribution of SAIDL is therefore not the development of a new attack-detection algorithm, but the integration and orchestration of security controls across the AI lifecycle together with adaptive feedback and quantitative security assessment.

Because NIST AI RMF, MITRE ATLAS, and OWASP LLM Security serve different purposes and are not executable end-to-end defense systems, Table 8 is presented as a conceptual positioning analysis rather than an experimental comparative evaluation of security performance. The qualitative descriptors (Low, Medium, and High) indicate the extent to which each approach explicitly addresses the architectural dimensions listed in the table, based on its documented scope, objectives, and security coverage. These descriptors are not measured performance scores and should not be interpreted as evidence of superiority of one framework over another. Instead, the comparison illustrates differences in architectural orientation and lifecycle coverage.

5.10 SSI Evaluation

The SSI can be used as an indicator of the efficiency of the overall framework by taking into account aspects such as threat mitigation, resilience, and feedback.

Fig. 7 shows that the proposed SAIDL configuration achieves the highest calculated SSI value among the configurations considered in the analytical evaluation. The reason behind such a significant increase in SSI can be attributed to the inclusion of lifecycle-based protection, threat adaptation, and resilience improvement within the framework.

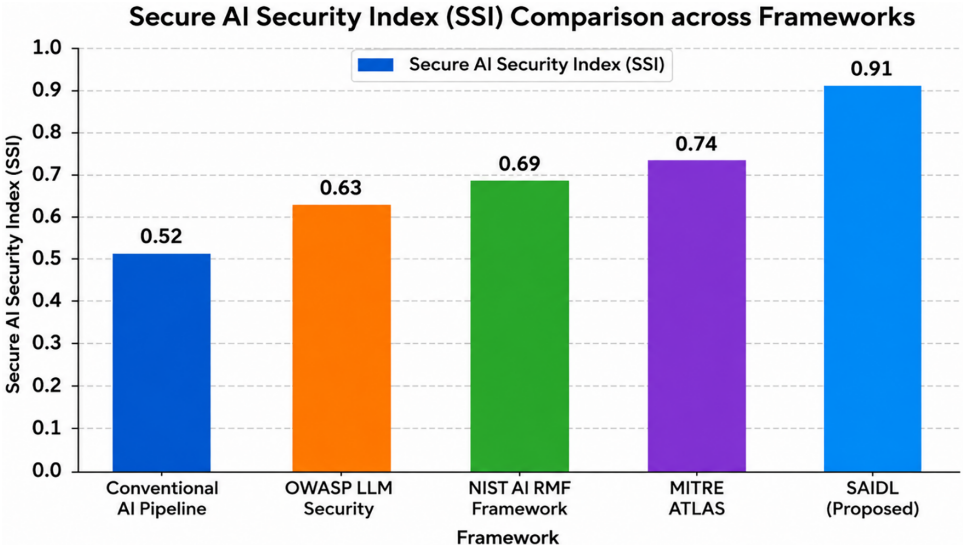


Figure 7: SSI evaluation.

6 Discussion, Practical Applications, and Recommendations for Future Research

With the growing use of AI in critical infrastructure, there is an increasing need for security architectures that can effectively address potential threats in AI environments. Unlike conventional software systems, AI environments are vulnerable to security threats throughout the entire lifecycle, including data collection,

model training, deployment, and operation. Consequently, individual security mechanisms may be insufficient to combat AI-enabled cybersecurity threats. The proposed SAIDL framework addresses this need through a lifecycle-based security architecture.

6.1 Discussion and Potential Practical Application

Implementation Feasibility: The proposed SAIDL architecture is meant to work like a security layer around the usual AI development and deployment environments, not as a full replacement for the underlying AI models, or infrastructure. The Data Security and Training Security Layers can be hooked into current data-validation, model-training, and adversarial-testing pipelines. The proposed SAIDL architecture is designed as a conceptual security layer that could be integrated with existing AI development and deployment environments. The Data Security and Training Security Layers could be incorporated into data-validation, model-training, and adversarial-testing pipelines, while Model Security could be integrated with model repositories and artifact-management processes. Deployment Security could operate alongside existing API gateways, access-control mechanisms, containerized services, and cloud or edge infrastructure. Runtime Monitoring could use existing logging and security-monitoring mechanisms, while the Incident Response and Recovery Layer could be connected with established security-response processes. Runtime Monitoring can take in application logs, model outputs, API activity, plus the security events that are already being produced by other monitoring systems. The Incident Response & Recovery Layer can be attached to what you already have in security operations, and incident-management processes. Meanwhile, the Threat Intelligence Feedback Engine acts as the coordination mechanism so that security events seen during operation can be used to revise detection rules, adjust security policies, and even guide retraining decisions. So overall SAIDL can be rolled out step by step, letting organizations bring in single layers as they go, while still staying compatible with existing cybersecurity infrastructure.

Based on the analytical and simulation-based evaluation, the proposed SAIDL framework indicates potential improvements in AI security through the integration of security processes across the AI lifecycle. The framework is designed to reduce exposure to attacks, limit threat propagation, and strengthen resilience through coordinated security controls. The Runtime Monitoring Layer and Threat Intelligence Feedback Engine provide mechanisms for detecting emerging threats and communicating relevant security information across lifecycle stages. These observations represent analytical and simulation-based indications of the potential contribution of SAIDL and require validation through operational implementation.

The proposed architecture could potentially be applied in areas where enhanced AI cybersecurity is required, such as healthcare, financial services, smart-city applications, automation processes, transport, and military applications. Furthermore, the flexible architecture could support organizations in considering the integration of individual security layers with existing cybersecurity systems, subject to domain-specific validation. Unlike traditional layered approaches in which security measures may operate independently, the SAIDL architecture provides a mechanism for communication between layers through the exchange of threat information. Such communication could allow security lessons learned during the execution phase to inform and refine subsequent stages of the AI lifecycle.

To further clarify the position of SAIDL relative to established AI security and governance approaches, Table 9 compares SAIDL with NIST AI RMF, MITRE ATLAS, OWASP AI/LLM security guidance, and ISO/IEC AI standards in terms of scope, methodology, security coverage, and implementation approach.

Table 9 illustrates the differences between SAIDL and other AI security and governance methods by considering such aspects as scope, methodology, security coverage, and implementation approach. The NIST AI RMF framework offers an extensive framework for risk management, MITRE ATLAS is concerned

with adversarial AI/ML techniques, while OWASP offers application-based guidance on security. ISO/IEC standards offer a common standardization foundation for AI governance, risk management, and security. On the other hand, SAIDL is focused on security controls orchestration throughout the full AI lifecycle, with threat intelligence being passed from one layer to another.

Table 9: Comparison of SAIDL with existing AI security and governance frameworks.

Framework	Scope	Methodology	Security Coverage	Implementation Approach
NIST AI RMF	AI risk management and governance	Govern, Map, Measure, Manage	Broad AI risks	Risk-management guidance
MITRE ATLAS	AI/ML adversarial threats	Threats, tactics and techniques	Attack-oriented	Threat-informed security
OWASP AI/LLM Guidance	AI/LLM application security	Vulnerability and risk guidance	Mainly AI/LLM application threats	Security recommendations
ISO/IEC AI Standards	AI governance, risk and security	Standardized management and governance practices	Governance, risk, security and lifecycle aspects	Organizational standards and controls
Proposed SAIDL	Lifecycle-wide AI security	Six security layers with threat-intelligence feedback	Data, model, training, deployment, runtime and recovery	Integrated and adaptive security architecture

AI-Specific Threats and SAIDL Lifecycle Coverage

Specific threats that emerge for AI systems may target any of the stages in the AI lifecycle and hence should be protected collectively in SAIDL. Data poisoning threats are mainly covered by Data Security (L1) through the validation of data, detection of anomalies and provenance of the data. Training security layer (L3) covers the threat to model training. Threats such as model extraction can be mitigated by Model Security (L2) through the use of measures such as model encryption, watermarking, and access control measures that are supported by the Deployment Security (L4). Adversarial attacks can be managed mainly by Training Security (L3) through adversarial training and robustness improvement. Runtime Monitoring (L5) can monitor abnormal model inference. Threats of prompt injection attacks can be managed by Runtime Monitoring (L5) through prompt monitoring, anomaly detection, and behavioral analysis. Model inversion attacks and privacy attacks may be countered using Model Security (L2), specifically via differential privacy and secure model access. Dangers related to the supply chain such as datasets, pretrained models, and other AI elements from third parties can be countered at various layers, namely L1–L4, through provenance validation, integrity validation of models, controlled access, secure deployment, and auditing. The security information about any new threats is distributed by the Threat Intelligence Feedback Engine to the appropriate layers.

Table 10 highlights how the new AI-specific threat categories are managed throughout the SAIDL lifecycle. In contrast to traditional cybersecurity management approaches which handle each type of attack

separately, SAIDL manages its security controls throughout different phases of its lifecycle. While data poisoning and supply-chain security threats are largely dealt with prior to and during the training phase via data and model validation, adversarial attacks are handled via robust training and runtime monitoring. On the other hand, model extraction attacks and privacy attacks are managed through model protection and secure access control. Lastly, prompt injection attacks are monitored at the runtime phase along with incident response.

Table 10: Mapping of emerging AI-specific threats to SAIDL lifecycle security layers.

AI-Specific Threat	Primary SAIDL Layer (s)	Key Security Mechanisms
Data poisoning	L1–L3	Data validation, anomaly detection, provenance verification, secure training
Model extraction	L2–L4	Model encryption, watermarking, authentication, access and API protection
Adversarial examples	L3–L5	Adversarial training, robustness enhancement, runtime anomaly detection
Prompt injection	L5–L6	Prompt monitoring, anomaly detection, behavioral analysis, containment
Model inversion and privacy attacks	L2	Differential privacy, secure model access
AI supply-chain risks	L1–L4	Data/model provenance, integrity verification, access control, secure deployment

Practical Applicability

SAIDL can be added to DevSecOps pipelines without changing the AI systems that are already in place. The Data and Training Security Layers can help with checking data, keeping training safe, and testing for attacks. The Model and Deployment Security Layers can work with places where models are stored, systems that control access, APIs with mechanisms to verify users, and cloud or edge setups. The Runtime Monitoring Layer can use existing logs and security tools. The Incident Response and Recovery Layer can link with the security teams that are already in place. The main problems when implementing SAIDL are the need for additional computing power, integration with legacy systems, and ensuring that personnel working with AI, software, and security collaborate effectively. SAIDL can therefore be implemented step by step based on the company's security requirements.

Table 11 shows how SAIDL can be added to DevSecOps processes by using security and AI tools that are already available. The framework introduces additional monitoring and security tasks; however, its layered design makes it easier for companies to adopt the components they need, step by step, based on their operational requirements and security goals.

6.2 Limitations and Future Research Directions

Though advantageous, SAIDL has a number of limitations. First, the scalability of the six-layer structure might be problematic for large-scale AI systems involving large volumes of data, distributed models, and runtime monitoring. Second, the reliability and relevance of threat intelligence influence the effectiveness of

the Threat Intelligence Feedback Engine. Third, the automatic assessment of AI system security is difficult because the security metrics and threat scores should be adjusted to the particular context. Finally, the current evaluation is performed under simulated attack conditions and using public benchmark datasets; thus, the presented results do not yet represent performance under real-life conditions. Further research will focus on the empirical evaluation of SAIDL through real-world deployment and larger-scale experiments. The next issues to investigate includes the automatic assessment of AI security, adaptive threat intelligence mechanisms, and scalability in the context of federated learning, edge AI, IoT networks, agentic AI, and multi-agent systems. The issues of explainable security decisions and self-defending AI systems will also be explored.

Table 11: Practical integration of SAIDL into DevSecOps.

SAIDL Layer	DevSecOps Integration	Main Mechanism
L1–L2	Data/model management	Validation, provenance, integrity
L3	Secure ML pipeline	Adversarial testing, secure training
L4	Deployment pipeline	Authentication, API protection
L5	Security monitoring	Anomaly and behavioral detection
L6	Incident response	Containment and recovery

Finally, the current evaluation is performed under simulated attack conditions using public benchmark datasets and does not constitute validation of an operational end-to-end SAIDL implementation. Consequently, the presented results should be interpreted as analytical and simulation-based indicators rather than evidence of real-world deployment performance. Further research is required to implement SAIDL in representative AI systems and evaluate its effectiveness under operational conditions.

7 Conclusion

Application of AI technologies in critical infrastructures has resulted in new threats, including AML attacks, data poisoning, model theft, prompt injection, and backdoor attacks. Existing security measures may not adequately address the diverse threats associated with these attack types. This research proposes SAIDL as a multi-layer architecture that integrates security controls across all stages of AI system development. The framework consists of six layers: Data Security, Model Security, Training Security, Deployment Security, Runtime Monitoring, and Incident Response & Recovery. This architecture is designed to establish a defense-in-depth security architecture for mitigating AI-powered cyberattacks. Moreover, the framework incorporates a Threat Intelligence Feedback Engine to enable adaptive security improvements. A mathematical framework and an SSI are introduced to assess the framework's potential contribution to threat mitigation, resilience, and overall security performance. In this way, SAIDL provides a conceptual foundation for integrating security controls across the AI lifecycle. The framework has potential relevance to critical domains such as healthcare, finance, smart cities, industrial automation, and autonomous systems. Future research will focus on the real-world implementation of the framework and its extension to emerging AI paradigms, including federated learning, edge AI, and autonomous multi-agent systems. The major contribution of this paper is not the design of a stand-alone machine learning approach but an integrated architectural approach. This architectural approach is made possible by the combination of lifecycle-based security orchestration, adaptive threat intelligence, mathematical security modelling, and quantitative security assessment. Although the proposed framework has been evaluated using multiple benchmark datasets representing different AI attack scenarios, it has not yet been validated through deployment within

a single operational AI application. Future work will include implementing SAIDL in domain-specific AI systems, such as LLM-based conversational agents, medical image classification systems, and intelligent industrial control systems, to evaluate application-specific security performance and attack surfaces.

Acknowledgement: The authors would like to thank the Papua New Guinea University of Technology (PNGUoT) for providing the academic environment and research support that facilitated the completion of this work.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: Dhaya Ramakrishnan: Conceptualization, methodology, framework design, mathematical formulation, investigation, writing—original draft preparation, visualization, and manuscript revision. Kanthavel Radhakrishnan: Supervision, validation, technical review, methodology refinement, and manuscript editing. Adline Freeda Russel: Literature review, data curation, manuscript review, and proofreading. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The datasets used in this study are publicly available benchmark datasets, including CIC-IDS2017, CIC-DDoS2019, NSL-KDD, ART Adversarial Robustness Toolbox datasets, and PromptBench. Additional information supporting the findings of this study is available from the corresponding author upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

Abbreviation	Full Form
ADR	Attack Detection Rate
AI	Artificial Intelligence
AI RMF	Artificial Intelligence Risk Management Framework
AML	Adversarial Machine Learning
API	Application Programming Interface
ARS	AI Resilience Score
ART	Adversarial Robustness Toolbox
CIC	Canadian Institute for Cybersecurity
DDoS	Distributed Denial-of-Service
FPR	False Positive Rate
FGSM	Fast Gradient Sign Method
LLM	Large Language Model
MITRE ATLAS	MITRE Adversarial Threat Landscape for Artificial-Intelligence Systems
NIST AI RMF	National Institute of Standards and Technology Artificial Intelligence Risk Management Framework
NSL-KDD	Network Security Laboratory–Knowledge Discovery in Databases
OWASP LLM Security	Open Worldwide Application Security Project Large Language Model Security
PGD	Projected Gradient Descent
SAIDL	Secure AI Development Lifecycle
SCS	Security Contribution Score
SIEM	Security Information and Event Management
SSI	Secure AI Security Index
TRR	Threat Reduction Ratio

References

1. Vassilev A, Oprea A, Fordyce A, Anderson H, Davies X, Hamin M. Adversarial machine learning: a taxonomy and terminology of attacks and mitigations. Gaithersburg, MD, USA: National Institute of Standards and Technology; 2025.
2. Qiao Y, Sathyanarayana NB, Shi C, He Z, Wang T, Hou T. A survey on adversarial machine learning: attacks, defenses, real-world applications, and future research directions. Neurocomputing. 2026;671:132670. doi:10.1016/j.neucom.2026.132670.
3. Siva Kumar RS, Nystrom M, Lambert J, Marshall A, Goertzel M, Comissoneru A, et al. Adversarial machine learning-industry perspectives. In: Proceedings of the 2020 IEEE Security and Privacy Workshops (SPW); 2020 May 21; San Francisco, CA, USA. p. 69–75. doi:10.1109/spw50608.2020.00028.
4. Kressel J, Perrella R, Reed E, Naik N, Sidhu J, Hu Q, et al. SAFE-AI: a framework for securing AI-enabled systems. McLean, VA, USA: MITRE; 2025 [cited 2026 Jan 1]. Available from: https://atlas.mitre.org/pdf-files/SAFEAI_Full_Report.pdf.
5. Seghid N, Iqbal F, Al-Room K, MacDermott Á. Emerging threats in AI: a detailed review of misuses and risks across modern AI technologies. Front Commun Netw. 2026;6:1727425. doi:10.3389/frcmn.2025.1727425.
6. Mohamed N. Artificial intelligence and machine learning in cybersecurity: a deep dive into state-of-the-art techniques and future paradigms. Knowl Inf Syst. 2025;67(8):6969–7055. doi:10.1007/s10115-025-02429-y.
7. Goldblum M, Tsipras D, Xie C, Chen X, Schwarzschild A, Song D, et al. Dataset security for machine learning: data poisoning, backdoor attacks, and defenses. IEEE Trans Pattern Anal Mach Intell. 2023;45(2):1563–80. doi:10.1109/TPAMI.2022.3162397.
8. Sidhpurwala H, Mollett G, Fox E, Bestavros M, Chen H. Building trust: foundations of security, safety, and transparency in AI. AI Mag. 2025;46(2):e70005. doi:10.1002/aaai.70005.
9. Zhang X, Zhang C, Li T, Huang Y, Jia X, Hu M, et al. JailGuard: a universal detection framework for prompt-based attacks on LLM systems. ACM Trans Softw Eng Methodol. 2026;35(1):1–40. doi:10.1145/3724393.
10. Huang J, Chen L. Defense against adversarial attacks via textual embeddings based on semantic associative field. Neural Comput Appl. 2024;36(1):289–301. doi:10.1007/s00521-023-08946-7.
11. Szadeczky T, Bederna Z. Risk, regulation, and governance: evaluating artificial intelligence across diverse application scenarios. Secur J. 2025;38(1):35. doi:10.1057/s41284-025-00495-z.
12. Megas K, Cuthill B, Dotter M, Garris M, Khemani I, Patrick B, et al. Cybersecurity framework profile for artificial intelligence (cyber AI profile): NIST community profile. Gaithersburg, MD, USA: National Institute of Standards and Technology; 2025. doi:10.6028/NIST.IR.8596.iprd.
13. He X, Xu G, Han X, Wang Q, Zhao L, Shen C, et al. Artificial intelligence security and privacy: a survey. Sci China Inf Sci. 2025;68(8):181101. doi:10.1007/s11432-025-4388-5.
14. Yerlikaya FA, Bahtiyar Ş. A survey of adversarial attacks on machine learning. Neurocomputing. 2026;670:132573. doi:10.1016/j.neucom.2025.132573.
15. Jedrzejewski FV, Thode L, Fischbach J, Gorschek T, Mendez D, Lavesson N. Adversarial machine learning in industry: a systematic literature review. Comput Secur. 2024;145(5):103988. doi:10.1016/j.cose.2024.103988.
16. Nakıp M, Gelenbe E. Online self-supervised deep learning for intrusion detection systems. IEEE Trans Inf Forensics Secur. 2024;19(8):5668–83. doi:10.1109/TIFS.2024.3402148.
17. He Z, Davila D, Bi S, Wang T, Hou T. Machine learning for cybersecurity: a survey of applications, adversarial challenges, and future research directions. Electronics. 2025;14(23):4563. doi:10.3390/electronics14234563.
18. Alotaibi A, Rassam MA. Adversarial machine learning attacks against intrusion detection systems: a survey on strategies and defense. Future Internet. 2023;15(2):62. doi:10.3390/fi15020062.
19. Zhang W, Kong X, Dewitt C, Braunl T, Hong JB. A study on prompt injection attack against LLM-integrated mobile robotic systems. In: Proceedings of the 2024 IEEE 35th International Symposium on Software Reliability Engineering Workshops (ISSREW); 2024 Oct 28–31; Tsukuba, Japan. p. 361–8. doi:10.1109/ISSREW63542.2024.00103.
20. Greshake K, Abdelnabi S, Mishra S, Endres C, Holz T, Fritz M. Not what you've signed up for: compromising real-world LLM-integrated applications with indirect prompt injection. In: Proceedings of the 16th ACM Workshop

on Artificial Intelligence and Security; 2023 Nov 26–30; Copenhagen, Denmark. p. 79–90. doi:10.1145/3605764.3623985.
21. Li Z, Peng B, He P, Yan X. Evaluating the instruction-following robustness of large language models to prompt injection. In: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing; 2024 Nov 12–16; Miami, FL, USA. p. 557–68. doi:10.18653/v1/2024.emnlp-main.33.
22. Yi J, Xie Y, Zhu B, Kiciman E, Sun G, Xie X, et al. Benchmarking and defending against indirect prompt injection attacks on large language models. In: Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining; 2025 Aug 3–7; Toronto, ON, Canada. p. 1809–20. doi:10.1145/3690624.3709179.
23. Jaffal NO, Alkhanafseh M, Mohaisen D. Large language models in cybersecurity: a survey of applications, vulnerabilities, and defense techniques. AI. 2025;6(9):216. doi:10.3390/ai6090216.
24. Tekeste B, Al-Hussaeni K, Fung BCM, Alawadhi I, Fachkha C. Adversarial machine learning: a 20-year survey of attacks, defenses, and standards. IEEE Access. 2026;14(3):69778–812. doi:10.1109/ACCESS.2026.3690620.
25. Chakraborty R, Hughes HK. Securing AI from adversarial attacks in the current landscape. Richardson, TX, USA: Infosys Knowledge Institute; 2024 [cited 2026 Jan 1]. Available from: https://www.infosys.com/iki/perspectives/securing-ai-adversarial-attacks.html.
26. Shafi M, Lashkari AH, Roudsari AH. Toward generating a large scale intrusion detection dataset and intruders behavioral profiling using network and transportation layers traffic flow analyzer (NTLFlowLyzer). J Netw Syst Manag. 2025;33(2):44. doi:10.1007/s10922-025-09917-0.
27. Ortet Lopes I, Zou D, Ruambo FA, Akbar S, Yuan B. Towards effective detection of recent DDoS attacks: a deep learning approach. Secur Commun Netw. 2021;2021:5710028. doi:10.1155/2021/5710028.
28. Bamber SS, Katkuri AVR, Sharma S, Angurala M. A hybrid CNN-LSTM approach for intelligent cyber intrusion detection system. Comput Secur. 2025;148:104146. doi:10.1016/j.cose.2024.104146.
29. Trusted-AI. Adversarial robustness toolbox (ART): Python library for machine learning security. 2025 [cited 2026 Jan 1]. Available from: https://github.com/Trusted-AI/adversarial-robustness-toolbox.
30. Zhu K, Wang J, Zhou J, Wang Z, Chen H, Wang Y, et al. PromptRobust: towards evaluating the robustness of large language models on adversarial prompts. In: Proceedings of the 1st ACM Workshop on Large AI Systems and Models with Privacy and Safety Analysis; 2024 Oct 14–18; Salt Lake City, UT, USA. p. 57–68. doi:10.1145/3689217.3690621.