Towards a Secure Framework for Regulating Artificial Intelligence Systems

Haroon Elahi, *Member, IEEE*, Nian Liu, Jiatong Chen, Fengwei Zhang, *Senior Member, IEEE*

***Abstract*—Regulating high-risk artificial intelligence (AI) systems is an urgent issue, yet technical infrastructure for their effective regulation remains scarce. In this paper, we address this gap by identifying key challenges in developing technical frameworks for AI systems' regulation and proposing conceptual, methodological, and practical solutions to address these challenges. In this regard, we introduce the concept of AI's operational qualification and propose the temporal self-replacement test, akin to certification tests for human operators, to examine the AI's operational qualification. We propose measuring AI's operational qualification across its operational properties critical for its regulatory fitness and introduce the operational qualification score as a pragmatic measure of AI's regulatory fitness. In addition, we design and develop a Secure Framework for AI Regulation (SFAIR), a tool for automatic, recurrent, and secure examination of an AI's operational qualification and attestation of its regulatory fitness, leveraging the proposed test and measure. Key strengths of SFAIR include its regulatory focus, flexibility in adapting to evolving regulatory requirements, and conformity to the secure-by-design principle. To achieve this, in addition to the aforementioned, we introduce a novel threat model for AI regulation frameworks. Considering the identified threats, we leverage randomization, masking, encryption-based schemes, and real-time monitoring to secure SFAIR operations. We also leverage AMD's Secure Encrypted Virtualization-Encrypted State (SEV-ES) for enhanced system security. We validate the efficacy of the temporal self-replacement test and the practical utility of SFAIR by demonstrating its capability to support regulatory authorities in automated, recurrent, and secure AI qualification examination and attestation of its regulatory fitness using an open-source, high-risk AI system. Finally, we make the source code of SFAIR publicly available.**

***Keywords*—High-risk AI, AI regulation, AI testing, qualification testing, secure-by-design, trustworthy AI.**

I. Introduction

RECENT advancement in artificial intelligence (AI) has exceeded earlier expectations, integrating AI systems into everyday life and positioning them to potentially replace human operators in various operational domains [1], [2]. However, AI systems remain fragile and susceptible to deception and corruption [3], [4]. Such unreliability and vulnerability raise significant concerns, particularly in high-risk AI systems (AI systems having the potential to harm human fundamental rights, health, safety, and the environment [5]), where the consequences of system failure can be severe [6]–[8]. Hence, there has been a growing outcry to address AI risks through effective regulatory measures [2], [9]. Accordingly, governments worldwide are responding by introducing relevant regulations [5], [10]–[12]. Yet, the technical infrastructure to enforce these regulations is currently missing [5], [9].

Although there exist some commercial AI monitoring systems [13]–[17], they are unfit for AI regulation, primarily due to a narrow focus and conflicts of interest. For example, AI regulation necessitates a recurrent examination to attest whether an AI system upholds its regulatory fitness [5], [9], i.e., whether it continues to be fit for operations within the given regulatory context post-deployment. However, existing solutions for post-deployment monitoring [13]–[16] primarily serve the in-house quality assurance needs of machine learning-based systems. Their focus remains either on monitoring the robustness of these systems to data and concept drift [15]–[17] or evaluating a subset of behavioral traits [13], [14], [16] required for determining regulatory fitness. Nevertheless, AI systems are not limited to those based on machine learning technologies [2], [5], and an AI system's regulatory fitness cannot be adequately examined through its partial evaluation. Likewise, the use of commercial AI monitoring tools for regulation introduces conflicts of interest. Additionally, while all major AI regulations [5], [10]–[12] underline the need to secure AI systems and their infrastructure, none of the existing solutions [13]–[16] consider security in their design.

The above limitations of the existing AI monitoring systems highlight the need to develop new frameworks tailored to AI regulation needs. However, developing these frameworks faces different challenges. Here, we identify five key challenges to developing effective post-deployment AI regulation frameworks and set associated design goals.

1) *Inadequate Methods:* Unlike conventional computing systems, AI systems' operational logic is not based on lines of code or linked to well-defined specifications. Consequently, the conventional testing methods depending on such traits of computing systems are unsuitable to examine AI systems [14], [18]–[20] and assess their regulatory fitness. Likewise, the drift-based methods used by existing AI monitoring frameworks [15]–[17] to evaluate the robustness of ML-based systems to changes in input data or concepts are ineffective in assessing the regulatory fitness of AI systems due to their limited scope [3]. Therefore, the first goal $(G1)$ is to design a

Haroon Elahi is with the Department of Computer Science and Engineering, Chalmers University of Technology and the University of Gothenburg, Sweden (Email: haroonelahi@ieee.org).

Nian Liu, Jiatong Chen, and Fengwei Zhang* (corresponding author) are with the Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, China, and Research Institute of Trustworthy Autonomous Systems, Southern University of Science and Technology, Shenzhen, China. (Email: 12132347@mail.sustech.edu.cn, chenjt2023@mail.sustech.edu.cn, zhangfw@sustech.edu.cn).

Manuscript received February 06, 2025; revised September 24, 2025.

methodological framework to examine AI systems and their regulatory fitness.

2) *Security Requirements:* Due to its strategic nature, risks associated with AI and its underlying infrastructure are high [21]. Therefore, it is unsurprising that all major AI regulations [5], [10]–[12] emphasize the need to identify novel threats to AI and its infrastructure and to develop relevant mitigation controls. Considering its role, we argue that AI governance infrastructure is equally critical and requires robust solutions. Therefore, the second goal $(G2)$ is to identify potential threats to AI regulation systems and secure our framework against relevant threats.
3) *Regulatory Evolution:* It is anticipated that the regulatory framework surrounding AI will continue to evolve in response to emerging AI capabilities and feedback [22]. Therefore, the third design goal $(G3)$ is to develop a flexible AI assessment framework capable of accommodating regulatory evolution.
4) *Conflicts of Interest:* Post-deployment monitoring for AI regulation is sensitive to trust issues. For instance, relying on commercial monitoring solutions from companies with a history of trust violations [23]–[26] creates conflicts of interest and undermines the principle of separation of powers. Therefore, the fourth goal $(G4)$ is eliminating regulatory authorities' dependence on commercial entities to avoid potential conflicts.
5) *Temporal Dimension:* Unlike conventional computing systems requiring a one-time certification, AI systems need recurrent evaluation to attest their regulatory fitness [5], [9], [27], introducing a temporal dimension to the AI regulation problem. This aspect is tied to AI's unpredictability [28] and its tendency to evolve and degrade over time [3]. Consequently, the fifth goal $(G5)$ is for the framework to account for AI's temporal evolution and enable regulatory authorities to automatically and recurrently assess AI's regulatory fitness.

A. Aim and Contributions:

This research aims to bridge gaps in AI regulation due to the absence of a secure technical infrastructure for examining and attesting the regulatory fitness of high-risk AI systems recurrently, automatically, and securely post-deployment. In this regard, we aim to identify the challenges to developing secure technical tools for evaluating these systems to examine and attest their regulatory fitness and address these challenges. While our preliminary investigations are led by a careful review of AI regulations in China [10], the EU [5], Great Britain [12], the United States [11], and related literature, we aim to introduce generalizable concepts and methods capable of serving the needs of any specific regulatory framework. The overall purpose of this research is to enable regulatory authorities to enforce AI regulations. In short, the paper makes the following contributions.

- *Theoretical and Methodological Contributions (Section IV):* We introduce the concept of AI's operational qualification II and propose the *temporal self-replacement test* (Section IV-B) to examine and attest these qualifications of AI systems post-deployment. Contrary to existing AI testing approaches relying on human discrimination, problem benchmarks, and peer confrontation [19], the *temporal self-replacement test* is self-discriminatory in nature and achieves its goal by recurrently testing properties of an AI system relevant to its operational qualification critical for its regulatory fitness. In addition, we introduce the *operational qualification score* to measure regulatory fitness quantitatively. A comparison of pre- and post-deployment *operational qualification scores* determines the preservation of the AI's regulatory fitness. The proposed test serves as a methodological framework. It allows for integrating additional properties and new testing methods to examine and attest AI qualifications corresponding to its emerging behaviors and relevant regulatory needs. We select the AI's relevant properties after reviewing major AI regulations [5], [10]–[12] and considering their impact, and provide their regulation-aligned definitions (Section IV-C).
- *Development of a Secure by Design Framework for AI Regulation (Section V):* We design and develop the *Secure Framework for AI Regulation* (SFAIR) to assist regulatory authorities in the secure, automatic, and recurrent examination of an AI's operational qualification and attestation of its regulatory fitness leveraging the *temporal self-replacement test*. SFAIR follows the secure-by-design principle [29]. In this regard, a novel threat model is defined (Section III), and security mechanisms, including randomization-based input selection, randomization of execution time, masking of test input to prevent information leakage, secure channels and cryptography, real-time monitoring, and hardware-based protection, are designed and integrated into SFAIR design to counter the identified threats.
- *Feasibility Demonstration (Section VI):* We demonstrate SFAIR's feasibility in assisting regulatory authorities in the secure, automatic, and recurrent post-deployment examination and attestation of AI qualifications through prototype implementation and experimental evaluation. In this regard, we examine an open-source, high-risk AI system [30] to demonstrate the efficacy and practical utility of *temporal self-replacement test* and SFAIR.
- *Provision of an Open-Source AI Regulation Tool:* We make the SFAIR prototype publicly available on GitHub under a free license to encourage the open-source community to further indulge in the development of post-deployment AI regulatory tools and to resolve the conflicts of interest issue arising from the use of commercial tools for regulatory purposes. It can be accessed from https://github.com/Compass-All/SFAIR.

B. Organization

The rest of the paper is organized as follows. Section II establishes background and reviews important related

works. Section III presents the assumptions and introduces a novel threat model. Section IV proposes the *temporal self-replacement test*, defines the *operational qualification score*, and offers the regulation-aligned definitions of the AI properties used in this work to determine its regulatory fitness. Section V presents the implementation details of SFAIR. Section VI evaluates the efficacy of the proposed test and framework. Section VII establishes the generality of the proposed concepts and methods, compares SFAIR with state-of-the-art, and discusses its computational complexity and practical challenges. Finally, Section VIII concludes the paper.

II. Background and Related Work

In this Section, we provide the background information and review important related works.

A. AI Regulation and its Requirements

AI regulations primarily set different mandatory requirements, prohibitions, and guidelines for AI systems. For instance, Chinese Interim Measures to regulate generative AI emphasize healthy development and standardized application of generative AI while safeguarding national security, public interests, and citizens' rights [10]. The European Union's AI Act [5] requires a thorough evaluation of high-risk AI systems pre and post-deployment. It also mandates identifying relevant threats and leveraging cybersecurity measures tailored to the security requirements of AI systems and their infrastructures. The British AI regulatory framework [12] demands responsible AI development, emphasizing safety, security, testing, data quality, and robustness throughout AI system development lifecycle stages. Meanwhile, the relevant regulations in the United States [11] focus on AI privacy, security, and fairness. However, since AI is an emerging field, the regulatory landscape for AI is expected to evolve as new needs emerge [22]. Consequently, relevant frameworks and solutions must be flexible to accommodate emerging regulatory requirements.

Furthermore, all major AI regulations require examining multiple aspects of AI systems. For instance, Chinese regulations [10] mandate that generative AI systems exhibit transparency, correctness, reliability, fairness, and responsibility. The EU's AI Act [5] imposes various requirements on high-risk AI systems, including high-quality data usage, documentation, traceability, transparency, correctness, human oversight, fairness, robustness, and cybersecurity measures. Similarly, the United States presidential order [11] sets various requirements for AI systems, including safety, fairness, data privacy, transparency, consideration of human alternatives, and fallback mechanisms. However, some regulatory requirements, such as using high-quality training data, protecting data privacy, traceability, documentation, notice and explanation mechanisms, and human oversight [5], [10], [11], pertain to pre-deployment quality assurance and design stages, and their presence should be confirmed pre-deployment. Moreover, no clear definitions for the relevant properties are provided in AI regulatory documents. Likewise, no regulation explains how to evaluate these AI properties and use them to determine regulatory fitness. In this work, we propose a methodological framework to evaluate relevant AI properties to examine the regulatory fitness post-deployment and provide the regulation-aligned definitions of the properties used for this purpose IV.

B. AI Evaluation

An effective AI regulation requires recurrent evaluation of AI systems to examine and ensure their regulatory fitness post-deployment [9], naturally requiring trustworthy methods and technical tools to automate this process. However, despite being a more than seven-decades-old field of research [31], AI evaluation is still in its infancy. Most existing AI testing approaches are adaptations of conventional testing methods [32]. Such methods can be of limited use [2]. For example, Hernandez [19] contends that adaptations of conventional methods can be used to test the conventionally developed components of a complex AI system. However, Aleti [33] cautions against the effectiveness of these methods, suggesting that their application in evaluating the trustworthiness of AI systems might provide users with a false sense of safety. Considering the nature and construction of AI systems, Hernandez [19] proposes that they should be tested as a black box based on their operational capabilities. In this regard, he identifies three approaches to AI evaluation: first, based on human discrimination, i.e., by comparing it against humans; second, using peer confrontation, i.e., whether it performs better than peer AI systems; and finally, using problem benchmarks, i.e., by testing whether it can solve a given set of problems. Similarly, Sifakis [2] suggests that AI systems should be evaluated as operational entities intending to replace humans in different operational contexts. We argue that such replacement requires evaluating AI systems for the operational abilities that human operators possess and rely on for effective performance in a given operational context.

Considering the AI regulation requirements, we propose using a *self-discriminatory* AI evaluation approach that repeatedly examines relevant AI operational abilities, necessitating the design of novel frameworks. In this regard, in line with the operational definition of AI, we propose a shift towards examining whether an AI system qualifies for use within a specific operational context and term it as an AI's *operational qualification*. Our notion of examining an AI's operational qualification parallels certifying human professionals who demonstrate their knowledge, skills, and expertise by passing carefully designed tests based on a subset of problems within their field before taking on an operational role. To bridge the methodological gaps, in this paper, we propose the *temporal self-replacement test* (Section IV) to examine and attest AI's operational qualification and introduce the *operational qualification score* as a pragmatic measure of regulatory fitness. Due to the regulatory focus of our work, we only examine the ability of an AI to correctly, fairly, efficiently, and robustly perform its operations. However, it is important to note that a broader examination can be conducted through carefully designed tests specific to an AI's operational domain. Regulatory bodies or their agents evaluate an AI system's operational qualifications before granting it a certification to operate in the wild, followed by recurrent post-deployment

examinations to ensure ongoing compliance reflected by the acceptable operational qualification score.

In addition, Anderljung et al. [9] point out the absence of technical tools for automatic AI regulation. In this regard, as we pointed out in Section I, current post-deployment AI monitoring tools [13]–[16] have various limitations in serving AI regulation requirements. We cover this gap by designing and developing a secure framework for AI regulation (Section V) that leverages the methods and measures proposed in this work. Following the secure-by-design principle [29], we design and integrate different cybersecurity measures to secure its operations.

III. Assumptions and Threat Model

A. Assumptions

We consider a competitive market where, before deployment, AI providers must demonstrate the required operational qualifications of their high-risk AI systems to the regulatory authorities (e.g., please refer to the EU AI Act [5]). The providers may develop AI systems specializing in an operational domain from the ground up or retrain general-purpose AI systems for this purpose. Upon successful examination, the regulatory authority may assign the AI system identity and issue a certificate to practice, allowing AI providers to deploy their systems in the wild. Additionally, considering the propensity of these systems to evolve, the regulatory authority recurrently re-examines them post-deployment to ensure they uphold their required operational qualifications.

B. Threat Model

The EU's AI Act [5] explicitly requisites identifying novel threats to AI systems and their infrastructure. Accordingly, this Subsection introduces a novel threat model identifying the threats, threat actors, and their capabilities that may affect an AI regulatory framework's confidentiality, integrity, or availability. Following the recommendations in [34], we keep our threat model practical and conservative.

1) Compromised AI: We consider high-risk AI systems that, despite undergoing stringent pre-deployment qualification tests under regulations [5], [10], may manifest post-deployment anomalies. They may develop unintended capabilities and exhibit undesired and unlawful operational behaviors due to changes in operational environments or adversarial manipulations [3], [35], which might allow these systems to interfere with regulatory frameworks' operations. Moreover, the ability of these systems to memorize their interactions may enable replay attacks using the memorized outputs to conceal their anomalous behavior [33].

2) AI Provider: AI provider develops, deploys, operates, and maintains the AI system. However, he depends on third-party service providers, such as cloud providers, to host the systems. He may try to disseminate unvetted features rapidly due to competitive pressures or malicious intent, bypassing thorough re-examination of the AI system by regulatory bodies [25], [36]. Moreover, the recurrent nature of qualification testing could incentivize a malicious provider to predict the timing of tests and employ substitutes between two assessments [37], [38]. The AI provider may also try to guess test inputs to spoof the regulatory framework by replaying previously generated outputs. However, while trying to evade regulations, the provider desires to retain the certificate to practice, necessitating preserving AI's identity and using covert methods to avoid regulatory apprehension.

3) Cloud Provider: The cloud provider has physical access to the AI systems' deployment infrastructure. Therefore, he may observe the deployed AI's behaviors. The cloud provider may also be a competitor to the AI provider (e.g., Amazon and Google offer hosting services and strive to lead the AI market [39]). A malicious cloud provider may want to disrupt qualification testing to damage the reputation of the AI provider [23], [24]. Under different circumstances, the cloud provider may try unauthorized access and AI system tampering. Furthermore, security misconfigurations by the cloud provider can allow infrastructure access to independent malicious entities [40].

4) Independent Malicious Entities: These entities (e.g., hackers and hacktivists) do not have physical access to the regulatory authority's infrastructure or the AI deployment infrastructure. However, they have developed capabilities to target critical infrastructures, and public digital infrastructures are their prioritized targets [41]. Consequently, they may hunt software vulnerabilities (e.g., memory-related vulnerabilities) [42], [43] in AI products and try to exploit them to disrupt the operations of the AI regulation framework to express their disregard for governments and their regulations or monitory reasons. In this work, we assume they are incapable of exploiting hardware vulnerabilities.

5) Out of Scope: Consistent with the confidential computing (CC) threat model assumptions [44], we do not consider side-channel attacks, sophisticated physical attacks, upstream hardware supply-chain attacks, and availability attacks.

IV. Temporal Self-Replacement Test

In this section, we propose the *temporal self-replacement test* to examine and attest an AI's operational qualification post-deployment and determine its regulatory fitness. However, considering the guidelines of the European Digital Identity (EUDI) Regulation [45] on attesting the qualifications of an individual, before examining an AI system's qualifications, its identification must be performed. Therefore, we first introduce a dynamic identity-based identification scheme, followed by the *temporal self-replacement test*.

A. Identification

Closely aligned with the EUDI Regulation [45], we define identification as the process of using data uniquely representing an AI system to verify its identity. Here, we provide the relevant definitions and procedural steps.

Definition 1. Let AI denote an artificial intelligence system. The operational identity of an AI, represented by its operational signatures ($Signature_{AI}$), comprises a set of abstractions of its k distinct operational behaviors demonstrated while processing k inputs ($\hat{I}$) within a specified context C.

Signatures generation is shown in Equation 1.

$$Signature_{AI} = f_{sig}[\hat{I}, C[AI](\hat{I})] \quad (1)$$

where f_{sig} represents the function that captures AI's k distinct behavioral abstractions, $\{\hat{b}_1, \hat{b}_2, \cdots, \hat{b}_k\}$, associated with each output $\hat{o} \in \hat{O}$ produced by the AI corresponding to $\hat{i} \in \hat{I}$, and appends them to the signature vector.

Identification Process:

1) Let $\hat{V}_{t_0} = \{\hat{v}_{1_{t_0}}, \hat{v}_{2_{t_0}}, \cdots, \hat{v}_{k_{t_0}}\}$ and $\hat{V}_{t_1} = \{\hat{v}_{1_{t_1}}, \hat{v}_{2_{t_1}}, \cdots, \hat{v}_{k_{t_1}}\}$ denote vectors containing normalized values representing behavioral abstractions corresponding to $\hat{O}_{AI_{t_0}}$ and $\hat{O}_{AI_{t_1}}$, where $\hat{O}_{AI_{t_0}}$ and $\hat{O}_{AI_{t_1}}$ are the outputs produced by the AI at times t_0 and t_1, respectively, corresponding to $\hat{I}$.
2) Measure the Jensen-Shannon Distance (JSD) between $\hat{V}_{t_0}$ and $\hat{V}_{t_1}$, which is the square root of the Jensen-Shannon divergence [46]. The JSD is given by: JSD($\hat{V}_{t_0}$, $\hat{V}_{t_1}$) = sqrt([KL($\hat{V}_{t_0}$,M) + KL($\hat{V}_{t_1}$,M)] / 2), where M is pointwise mean of $\hat{V}_{t_0}$ and $\hat{V}_{t_1}$ and KL is the Kullback-Leibler divergence. The JSD value lies between 0 and 1. A JSD value of 0 confirms the identity of the AI system, given that it is highly improbable for two distinct AI systems to generate exactly the same probability distributions.

B. Temporal Self-Replacement Test

We propose the *temporal self-replacement test* to examine and attest an AI system's operational qualifications. This test examines and attests whether or not a deployed AI system preserves its operational qualifications, validated through an operational qualification score measure.

Definition 2. Let AI be an artificial intelligence system that, in a given operational context C, for each distinct test input i, with an associated ground truth g, produces a distinct output o, with an associated statistical attribute s. The operational context, C, includes objectives, operational conditions, and task performance expectations. The test inputs and outputs have their respective domains $Dom(i) = I$ and $Dom(o) = O$. Further, AI's operational behavior is depicted by $o = C[AI](i)$. AI's operational qualification is evaluated by an oracle $H(i, o)$, which determines its success or failure in performing a task involving input $i \in I$ and the corresponding output $o \in O$, using pre-determined criteria.

Let $\hat{I}_{p_j} \in I$ be a sequence of k randomly selected adequate inputs, used to examine the AI's property $p_j \in P$, which is relevant while determining its operational qualification. In context C, if AI performs a task evaluating $p_j \in P$ at t_1 as effectively as at t_0 ($\forall \hat{i}_{p_j} \in \hat{I}_{p_j}, H(\hat{i}_{p_j}, C[AI_{t_1}](\hat{i}_{p_j})) \iff H(\hat{i}_{p_j}, C[AI_{t_0}](\hat{i}_{p_j}))$), AI's operational qualification relevant to p_j is equivalent at t_0 and t_1.

Test Procedure

1) *Identify AI's critical properties:* Identify a set of properties P of an AI demonstrating its operational qualifications critical for its regulatory fitness.
2) *Random input selection:* For each property $p_j \in P$, select a sequence of k random inputs ($\hat{I}_{p_j} \in I$) adequate for testing p_j.
3) *Assessment using success criterion:* Test $p_j \in P$ using $\hat{I}_{p_j}$ and evaluate the corresponding outputs at t_0 and t_1 according to the success criterion H_{p_j} (where H_{p_j} is either the ground truth-based i.e., H_δ or threshold-based i.e., H_θ).
Ground truth-based success criterion: Let $\hat{i} \in \hat{I}$ denote a test case with an associated ground truth value $\hat{g}$, used to evaluate AI within context C and let $\hat{o} \in \hat{O}$ be the corresponding output. The oracle H_δ determines the success score as follows:
$$Score_{H_\delta} = \delta^{\hat{\mathbf{o}}}_{\hat{\mathbf{g}}} = \begin{cases} 1 & \text{if } \hat{g} = \hat{o} \\ 0 & \text{if } \hat{g} \neq \hat{o} \end{cases}$$
where δ is the Kronecker delta function that returns 1 if the output of the test $\hat{o}$ matches the corresponding ground truth $\hat{g}$ and 0 otherwise.
Threshold-based success criterion: Let $\hat{i} \in \hat{I}$ denote a test input to evaluate AI within context C. Let $\hat{o} \in \hat{O}$ be the corresponding output, and let $\mu_{\hat{o}}$ be the measure associated with $\hat{o}$. Suppose τ is the threshold value. Then, the oracle H_θ determines the success score as follows:
$$Score_{H_\theta} = \mathbf{1}_{\{\mu_{\hat{o}} \leq \tau\}} = \begin{cases} 1 & \text{if } \mu_{\hat{o}} \leq \tau \\ 0 & \text{if } \mu_{\hat{o}} > \tau \end{cases}$$
where $\mathbf{1}_{\{\mu_{\hat{o}} \leq \tau\}}$ is an indicator function returning one if the corresponding output is less than or equal to the threshold (tau), and zero otherwise.
4) *Validate property preservation:* Let $\hat{V}_{t_0}$ and $\hat{V}_{t_1}$ be two binary vectors of equal length, each containing k elements such that $\forall\ \hat{v}_{x_{t_0}} \in \hat{V}_{t_0}$ and $\hat{v}_{x_{t_1}} \in \hat{V}_{t_1}$, where x is an index ranging from 1 to k, $\hat{v}_x = \delta^{\hat{o}_x}_{\hat{g}_x}$ or $\hat{v}_x = \mathbf{1}_{\{\mu_{\hat{o}_x} \leq \tau\}}$ corresponding to the scores of k tests condcuted to test a property P at t_0 and t_1. Measure the Hamming distance (d_H) between $\hat{V}_{t_0}$ and $\hat{V}_{t_1}$ using: $d_H(\hat{\mathbf{V}}_{\mathbf{t_0}}, \hat{\mathbf{V}}_{\mathbf{t_1}}) = \sum_{x=1}^{k} \mathbf{1}_{\{\hat{v}_{x_{t_0}} \neq \hat{v}_{x_{t_1}}\}}$, where k is the length of the binary vectors, $\hat{v}_{x_{t_0}}$ and $\hat{v}_{x_{t_1}}$ are the x-th elements of $\hat{\mathbf{V}}_{t_0}$ and $\hat{\mathbf{V}}_{t_1}$ respectively, and $\mathbf{1}_{\{\hat{v}_{x_{t_0}} \neq \hat{v}_{x_{t_1}}\}}$ is an indicator function that returns 1 if $\hat{v}_{x_{t_0}} \neq \hat{v}_{x_{t_1}}$ and 0 otherwise. The property $p_j \in P$ and associated operational qualification of the tested system are deemed preserved if $d_H \approx 0$.
5) *Attest the operational qualification:* Validate if the AI's operational qualification score at t_1 mirrors its corresponding score at t_0. The operational qualification score is calculated as follows: $\forall p_j \in P \hat{i}_{p_j} \in \hat{I}_{p_j}$, $\frac{1}{j}\left\{\sum_{x=1}^{j}\left(\frac{1}{k}\left(\sum_{\hat{i}_{p_j}=1}^{k} H_{p_j}(\hat{i}_{p_j}, C[AI_{p_x}](\hat{i}_{p_j})\right) \times 100\right)\right\}$, where $\hat{i}_{p_j} \in \hat{I}_{p_j}$ is the ith input suitable for testing the jth property and H_{p_j} is the corresponding success criterion. The operational qualification scores for the given AI should be approximately equal at t_0 and t_1.

C. The Tested Properties

The *temporal self-replacement test* validates whether an AI system upholds its operational qualifications reflected by properties such as operational accuracy, fairness, efficiency, and robustness over time. The *temporal self-replacement test* is flexible in its construction (generizable), and the tested properties can vary in accordance with the requirements of the specific regulatory framework applicable to the region where the AI system is operating. Here, for demonstration purposes, we select the properties based on a review of major regulatory documents [5], [10]–[12] and related literature [9], [20], [47]. Below, we elaborate on these properties:

1) Operational Accuracy: Article 4(2) of the Chinese Interim Generative AI Measures [10] sets out accuracy as a requirement for AI systems. However, the definition of accuracy is not provided in major AI regulations [5], [10]–[12]. Nevertheless, the British Data Protection Act 2018 [48] defines inaccurate as "incorrect or misleading as to any matter of fact". Consequently, we treat the operational accuracy of an AI as a measure of its success or failure in task completion, considering a pre-defined criterion, which serves as the fact. We define operational accuracy as follows.

Definition 3. The operational accuracy of an AI in successfully completing a task $\forall \hat{i} \in \hat{I}$ and within context C can be expressed as $H_\delta[\hat{i}, C[AI](\hat{i})]$, where H_δ measures the success or failure in task completion based on the closeness of $[\hat{i}, C[AI](\hat{i})]$ to the ground truth.

In the *temporal self-replacement test*, an AI's aggregate operational accuracy is measured by $\sum_{x=1}^{k} H_\delta[\hat{i}_x, C[AI](\hat{i}_x)]$.

2) Operational Fairness: Article 4(2) of the Chinese Interim Generative AI Measures [10], Article 10 and Article 77 of the EU AI Act [5], and Article 21 of the European Union Charter of Fundamental Rights [49] define fairness in terms of just treatment and non-discrimination. Accordingly, an AI is fair if its operations within a given context, C, are free of bias and discrimination (direct, indirect, or multiple), i.e., harmful effects of bias, towards any entities it interacts with.

Definition 4. Let d={direct discrimination, indirect discrimination, multiple discrimination}. Let $Z = \{z_1, z_2, \cdots, z_n\}$ be n 'different' entities. An AI is operationally fair if $\forall\ \hat{i} \in \hat{I}$, $z_{n-x} \in Z$, and $z_n \in Z$ it satisfies: $H_{\delta_{z_{n-x}}}[\hat{i}, C[AI](\hat{i})] = H_{\delta_{z_n}}[\hat{i}, C[AI](\hat{i})]$, where comparing the evaluation using H_δ determines whether AI's treatment of any z_{n-x} and z_n is equal, i.e., AI is free of discrimination.

We adopt the definitions of direct, indirect, and multiple discrimination provided by Fibbi et al. [50] and extend them. *Direct discrimination* is a straightforward form of discrimination where individuals (humans or AIs) or groups of individuals (humans or AIs) are discriminated against based on their characteristics or due to selfish motives. *Indirect discrimination* refers to situations where apparently neutral treatments produce disproportionate disadvantages for one category or individuals compared to others. Finally, *multiple discrimination* arises on multiple grounds.

3) Operational Efficiency: Efficiency measures the speed of an AI to execute a task [20]. Although major AI regulations [5], [10]–[12] do not require evaluating efficiency to determine the regulatory fitness of AI systems, it is crucial in many high-risk scenarios [47]. For example, in an emergency, how much time an autonomous vehicle takes to hand the driving control over to the human driver can determine the fate of the driver and the passengers [6]. We define AI's operational efficiency as follows.

Definition 5. Given $\hat{i} \in \hat{I}$, C, and $E_{\mu_{\hat{i}}}$, where $E_{\mu_{\hat{i}}}$ is the time needed to process an input $\hat{i}$. The operational efficiency of an AI system is its ability to generate an output ($\hat{o} \in \hat{O}$), where $\hat{o} = C[AI](\hat{i})$, within a desired time interval, θ. This relationship is represented as: $Efficiency_{AI} = H_\theta[\hat{i}, C[AI](\hat{i})]$ where H_θ verifies if the time taken to process $\hat{i}$ is less than the threshold value, i.e., $E_{\mu_i} \leq \theta$.

4) Operational Robustness: Consistent with the EU's AI Act [5], we treat operational robustness as the AI's ability to maintain consistent performance despite minor changes in input or its operational environment.

Definition 6. Let $i' = \hat{i} + \triangle_i$ be a test input with noise (naturally occurring) or perturbations (artificially induced), such that $dist(\hat{i}, i') \leq \varepsilon$ (ε represents the limit of human perceptibility). Let C be the operational context, and $C' = C + \Delta_C$, where Δ_C denotes minor changes in the operational environment. An AI is operationally robust if it satisfies: $H_\delta[\hat{i}, C[AI](\hat{i})] \approx H_\delta[i', C[AI](i')]$, $H_\delta[\hat{i}, C[AI](\hat{i})] \approx H_\delta[\hat{i}, C'[AI](\hat{i})]$, and $H_\delta[\hat{i}, C[AI](\hat{i})] \approx H_\delta[i', C'[AI](i')]$.

V. The Secure Framework for AI Regulation (SFAIR)

This section introduces the assumptions, ecosystem, and design of SFAIR, the framework that leverages the *temporal self-replacement test* (Section IV) to support regulatory authorities in the secure, automated, and recurrent examination of the AI systems post-deployment to attest their regulatory fitness.

A. Design Assumptions

The SFAIR operates based on three key assumptions. Firstly, it assumes that the regulatory body possesses the technical expertise to evaluate the operational qualification of the high-risk AI system. Secondly, it assumes that the regulatory body has thoroughly evaluated the AI's operational qualifications to assess its regulatory fitness before its deployment, leveraging evidence from both the AI provider and independent assessments and registering it at the deployment time. Thirdly, it assumes that the regulatory body maintains a repository containing a set of adequate test inputs to examine the AI's operational qualifications, along with a preserved copy of the AI that was approved for deployment for cross-referencing (a widely used approach in assessing AI systems [33], [51]). Additionally, the regulatory authority has access to the deployed AI requiring regulation, which tends to evolve due to reasons identified in Section III.

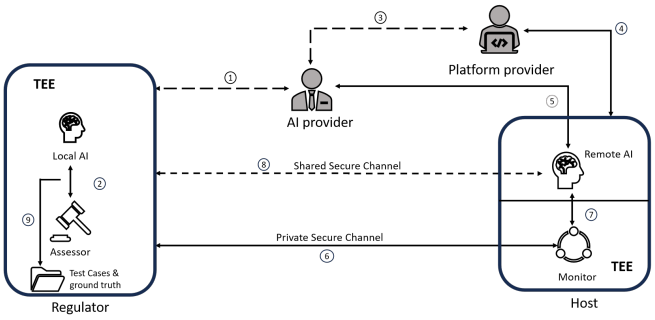


Fig. 1: *SFAIR Workflow:* ① *The AI provider submits the AI system, test data, and relevant documentation to the regulatory authority.* ② *The authority conducts preliminary qualification tests to validate if the system qualifies for autonomous operations and grants approval if the system passes the test.* ③ *The AI provider engages a deployment platform provider.* ④ *The deployment platform provider sets up the environment and resources for hosting the AI.* ⑤ *The AI provider deploys the AI on the designated platform.* ⑥ *The regulatory authority deploys a secure monitor within the deployment host and establishes a secure communication channel.* ⑦ *The regulator registers the deployed AI with the monitor for continuous monitoring.* ⑧ *The regulator uses secure channels to perform post-deployment qualification testing repeatedly.* ⑨ *Remotely conducted tests' scores are compared against local test scores to validate if the AI preserves its qualifications.*

B. SFAIR Ecosystem

Figure 1 provides an overview of the SFAIR ecosystem and depicts its workflow. This workflow within SFAIR systematically and securely tests an AI system's operational qualifications by evaluating the success scores it secures during initial qualification testing pre-deployment and the recurrent testing carried out by the regulatory authority post-deployment. The workflow includes two primary components: the assessor, operating on the regulator's machine and responsible for conducting the *self-replacement test* and analyses, and the monitor, functioning on the remote machine hosting the AI operating in the wild, responsible for its continuous monitoring and alerting the assessor if unauthorized changes in the AI or context are observed. Moreover, secure channels and trusted execution environments (TEE) are used for secure communication and system-level security, respectively. A complete introduction of all major SFAIR components and their design is provided in the following subsections.

C. SFAIR Design

This subsection introduces major SFAIR components and their design. These components are designed considering the challenges and the design goals identified in Section I considering AI regulation requirements.

1) The Deployment Module: The deployment module helps the regulatory authority deploy the preserved copy of the AI system. It includes setting up the operational environment needed to examine and attest the deployed copy of AI. Since SFAIR focuses on examining deployed systems, a web-based interface, the web server gateway interface, is used for this purpose. It specifies how the input will be received from the assessor and fed to the AI system. It also ensures that the input used to examine the AI system is per the expected format.

2) The Registration Module: The registration module is used by the regulatory authority or the AI provider (depending on the regulatory requirements) at the deployment time to register the AI_{t_1} with the monitor. It includes AI provider data and metadata of AI_{t_1} extracted using different OS functions (e.g., $os.path.getctime()$ and $os.stat()$) to note the information that can be later used to detect unauthorized changes. Furthermore, it notes the operational settings in which the system operates to validate that the information provided by the AI provider is correct and repeatedly verifies it in the future.

3) The Assessor: The assessor's primary responsibility is to identify an AI and perform the *temporal self-replacement test* to determine if it has maintained its regulatory fitness depicted by its operational qualifications. The assessor begins by identifying the deployed AI. The assessor has a *signature generator* module, which generates runtime operational signatures for AI_{t_0} and AI_{t_1}. Given inputs $\hat{I}$, a size parameter k, and the list of recent ten signatures R_{10}, it iteratively constructs signatures by appending the confidence score $s_{\hat{o}_j}$ associated with each output generated by the AI to the signature. If the generated signature is not already present in R_{10}, it adds the signature to the list of recent signatures, i.e., R_{10}, and returns it for identification. If a match with an existing signature is detected, the algorithm recursively retries the signature generation process until a unique signature is created. Authentication is performed following the process provided in Section IV-A

Once the AI's identity is confirmed, the assessor contacts the monitor to check for changes in AI_{t_1} and its operational context. It then examines the deployed AI using the *temporal self-replacement test*. The assessor curates a pool of adequate test inputs to examine the operational qualifications and regulatory fitness. SFAIR uses a *systematic scheme* for arranging these input files into directories named after the properties they test and the input classes that they belong to. For example, the inputs used for testing the robustness property are stored in the robustness directory, which has two sub-directories: *perturbed* and *benign*, each of which has, in turn, two sub-directories, *male* and *female*. The assessor indexes test inputs stored across these directors. The assessor uses a 'select_random_input()' function to select random files from the test suite. The user can specify the property name, the number of directories to walk, and the number of random test files randomly picked from each sub-directory.

As mentioned above, the assessor indexes all test inputs stored in the test suite. However, the index includes fully qualified names of the test inputs. Using the file names as given in the index can reveal sensitive information, such as the test input class or the test cache's directory structure. The task of the mask generator is to remove such information before the file is used to examine remotely deployed AI. The assessor uses a 'generate_mask' helper function for mask generation.

It takes the list of randomly selected test inputs $randInput(\hat{I})$ designated to test a given AI property in the given test run, $size(m)$ (the size of the mask - it has to be the same as the length of the input list), and the list of ten recent masks, $recent10(R_{10})$ to return a mask. A recursive approach ensures that the generated mask is distinct from the recent ten masks, R_{10}. The policy preventing the reuse of recently used ten masks is motivated by industrial best practices prohibiting password reuse [52].

As given in Section IV-B, the success criteria for an individual test comprise a threshold value or a ground truth. The assessor has a *ground truth generator* module that generates ground truth during each test run. This module takes advantage of the input file names in the index, which include the information about the property they test and the class to which they belong. Thus, for example, a random input selected from the 'women' folder must essentially be a picture of a female. This fact is used to generate the ground truth. This scheme helps solve the oracle problem in SFAIR, i.e., determining the success or failure of the test without human intervention [53].

Moreover, SFAIR randomizes the time across testing iterations to prevent timing-based attacks. The *test scheduler* module manages this randomization. The scheduler sets up the next test execution between the minimum time, (t_{min}), which is initialized to the current system time, and the maximum time, $MaxTime$, which is calculated as $t_{min} + t_{max}$. t_{max} is set by the SFAIR user. Using the $\texttt{Rand}(t_{min},\ t_{max})$ function, the scheduler sets a random time, t_{next}, the point at which the next test will be invoked. Meanwhile, the scheduler enters a perpetual loop to monitor the system time continuously. It checks if the current system time has reached or surpassed t_{next}. Upon reaching t_{next}, the scheduler invokes the 'test dispatcher,' initiating the qualification assessment process. After triggering a test, t_{next} is updated by recalculating a new random time.

The *test dispatcher* dispatches test inputs required for conducting the *temporal self-replacement test*. It receives requests from the test scheduler and organizes transmitting relevant input files over secure channels. The *result accumulator*, in turn, collects the outcomes stemming from testing properties and prepares them to forward to the *property preservation tester* that follows the process defined in Section IV-B, returning the operational qualification scores for AI_{t_0} and AI_{t_1} to determine whether the system has preserved its regulatory fitness. The results are logged in JSON files in the results directory and displayed on the screen for user-friendliness.

4) The Monitor: Deployed in a CVM on the machine hosting AI_{t_1}, the *monitor* runs as a service. It has an *age tracker* module that verifies the *age* of AI_{t_1} by comparing its metadata (created and modified times) observed at the current system time with that noted at the registration time. The second module is the *context tracker*, which uses system environment variables to track changes in the operational context of the AI_{t_1}, which comprises its runtime environment and hosting system. In our implementation, the monitor tracks changes in the operating system (OS) version, Python and its libraries, and Flask. The tracked changes are passed on to the *alert generator*, which is the third module of the monitor and collaborates with the *age tracker* and the *context tracker*. It generates *alarms* and logs relevant information in timestamped JSON files locally in the instances where the AI_{t_1} has been substituted or modifications have occurred to it or its operational context. The *alarm* serves as an alert to the assessor, prompting the regulatory authority to take adequate action. The *monitor* establishes a secure channel to communicate with the assessor using symmetric key cryptography.

5) Hardware-Based Security: SFAIR relies on Confidential Computing (CC) [54] for system-level security. CC harnesses hardware-based security functionalities to establish trusted execution environments (TEEs). SFAIR leverages AMD Secure Encrypted Virtualization - Encrypted State (SEV-ES) technology for this purpose [55]. AMD SEV uses memory encryption capabilities to offer encrypted virtual machines. Inline memory encryption/decryption is performed by the hardware AES engine located in the memory controller. One private key, unknown to any software on the central processing unit (CPU), is used for encryption/decryption per virtual machine to isolate guests and the hypervisor cryptographically. SEV-ES further enhances this protection by encrypting the guest register state using its encryption key. SFAIR setup involves the creation of two CVMs: one to protect the assessor and the preserved copy of AI, and another to protect the monitor.

The design decision to leverage SEV-ES was driven by the availability of the relevant hardware. The more recent AMD SEV-SNP (Secure Encrypted Virtualization - Secure Nested Paging) introduces additional security mechanisms to support additional VM use models and offers increased protection against various threats, including those arising from recently disclosed side channel attacks [56]. Thus, SFAIR implementations leveraging SEV-SNP can further improve system security.

VI. Experimental Evaluation and Results

In this Section, we provide the details of the experimental setup, evaluate the effectiveness of *temporal self-replacement test* and SFAIR in helping the regulatory authorities examine and attest an AI system's operational qualifications and regulatory fitness, conduct security evaluation to demonstrate its ability to counter threats identified in Section III, and measure the impact of using SEV-ES-based security on its overall performance.

A. Experimental Setup

The SFAIR prototype is implemented using 1,116 lines of Python 3.10.12 code. Moreover, scripts for AI system hosting and running tests are written in the Bourne Shell scripting language. We use an open-source code analysis tool, cloc [57], to count the lines of code written for the SFAIR prototype implementation. Table I provides a breakup of the lines of code according to different functionalities. The SFAIR *assessor* and *monitor* are deployed on two AMD Secure Encrypted Virtualization-Encrypted State (SEV-ES)-based CVMs [55]. The primary server features an AMD EPYC 9274F 24-core processor running the Linux kernel version 6.5.0. Each CVM is configured with 8GB of memory and a 100 GB hard disk and is managed through the open-source library libvirt [58]. Both

CVMs run Ubuntu 24.04 LTS with minimal configurations. For evaluation, we select a publicly available face and gender

TABLE I: Code Breakup

Functionality	Language	Lines of Code
Registeration	Python	52
Assessor	Python	568
Monitor	Python	102
AI Deployment	Python	76
AI and Test Launchers	Bourne Shell	23
Data Processing & Visualization	Python	318

detection [30] AI system. Such systems are integrated into use cases, such as identification, personalized healthcare, and security solutions [59], involve processing sensitive personal information, and can affect fundamental human rights and the well-being of users. Consequently, the AI systems for gender detection are classified as high-risk and are subject to regulation [5]. Therefore, the selected system is suitable for our demonstration. The operational qualifications of the system under examination (SUE) demonstrating its ability to accurately, fairly, and efficiently detect gender and its robustness to perturbations are validated using the 'Labeled Faces in the Wild (LFW)' dataset, a public benchmark for face verification [60]. Deployment of the SUE is facilitated using Gunicorn 22.0.0 [61], enabling remote access for the *assessor* via network requests. Data exchange between the assessor and monitor is secured using Transport Layer Security (TLS), with session signing performed within the CVMs.

B. Experimental Evaluation

In this subsection, we experimentally evaluate SFAIR and address the following research questions (RQs):

1) **RQ1:** How does SFAIR help regulatory authorities in recurrent examination and attestation of an AI system's operational qualifications post-deployment?
2) **RQ2:** Can SFAIR's security measures counter the threats identified in the threat model in Section III?
3) **RQ3:** How does deploying SFAIR in CVMs affect its performance?

The **RQ1** intends to validate whether SFAIR can achieve its operational goals of assisting the regulatory authorities in recurrently examining and attesting SUE's operational qualifications to determine its regulatory fitness. To address **RQ1**, we perform qualification testing of SUE using SFAIR. Since AI systems often perform poorly when exposed to unseen data, we use the LFW dataset [60], distinct from the SUE's training dataset, to examine and validate its operational qualifications. Furthermore, based on findings from [62], which suggest that one hundred benign inputs are sufficient to discover distinct AI behaviors with a confidence level of 92%, we examine all relevant operational properties using 100 inputs adequate for testing the corresponding property. We use images with diverse poses, illumination conditions, and emotional variations to examine the operational accuracy of SUE. Such diversity increases the task's difficulty level and is known to influence an AI system's performance significantly [63], and is, therefore, suitable to examine operational accuracy. Likewise, contrasting inputs can expose biases in an AI system's operational behavior [16]. Hence, we examine SUE's fairness by measuring the differences in its success in face recognition and gender detection tasks for different races. Similarly, 100 diverse inputs are used to examine the operational efficiency. Moreover, we use the Fast Gradient Sign Method (FGSM) [64] to introduce perturbations in the inputs used to examine SUE's robustness to minor changes in the input.

Additionally, we validate the SUE's robustness to changes in the operational context by altering the deployment platform versions and substituting the initially registered SUE version with another without re-registration. Furthermore, to demonstrate the ability of SFAIR to detect unauthorized changes in SUE, we retrain it using an extended dataset comprising original images and 5% additional images drawn from the LFW [60] dataset. Since SFAIR does not proceed if identity is not verified, we disable identity checking to simulate the scenarios where the AI providers may use augmentation-based techniques to change the AI qualifications while retaining its identity. We perform security evaluation for **RQ2**, which aims to theoretically and experimentally assess the effectiveness of SFAIR's security design. Lastly, **RQ3** considers the implications of deploying SFAIR in CVMs on its performance. After all, using cryptographic primitives has associated costs [65]. We compare SFAIR's execution time to examine the SUE's efficiency when deployed with and without SEV-ES features for 500 experiments each. Here, we provide the answers to these research questions.

RQ1: How does SFAIR help regulatory authorities in recurrent examination and attestation of an AI system's operational qualifications post-deployment?

Figures 2 and 3 present the plots for the *operational qualification scores* of successful and failed temporal self-replacement tests, depicting that SFAIR successfully evaluates and effectively measures whether the SUE preserves its operational qualifications pre and post-deployment. In the first case (Figure 2), where the SUE successfully passes 500 temporal self-replacement tests, the minimum operational qualification score is 77.50, the maximum is 82, the mean is 79.83, and the standard deviation is 0.92 pre and post-deployment.

Figure 3 shows plots representing the results of 500 temporal self-replacement tests where the SUE fails to pass the tests, which is suggested by the differences effectively detected by SFAIR among corresponding post and pre-deployment operational qualification scores measured across 500 runs of the test. Post-deployment statistics are as follows: minimum score: 87.75, maximum score: 92.75, mean score: 90.06, and standard deviation: 0.87. However, pre-deployment statistics, significantly differing from post-deployment statistics, are as follows: minimum score: 77.25, maximum score: 82.5, mean score: 79.72, and standard deviation: 1.01.

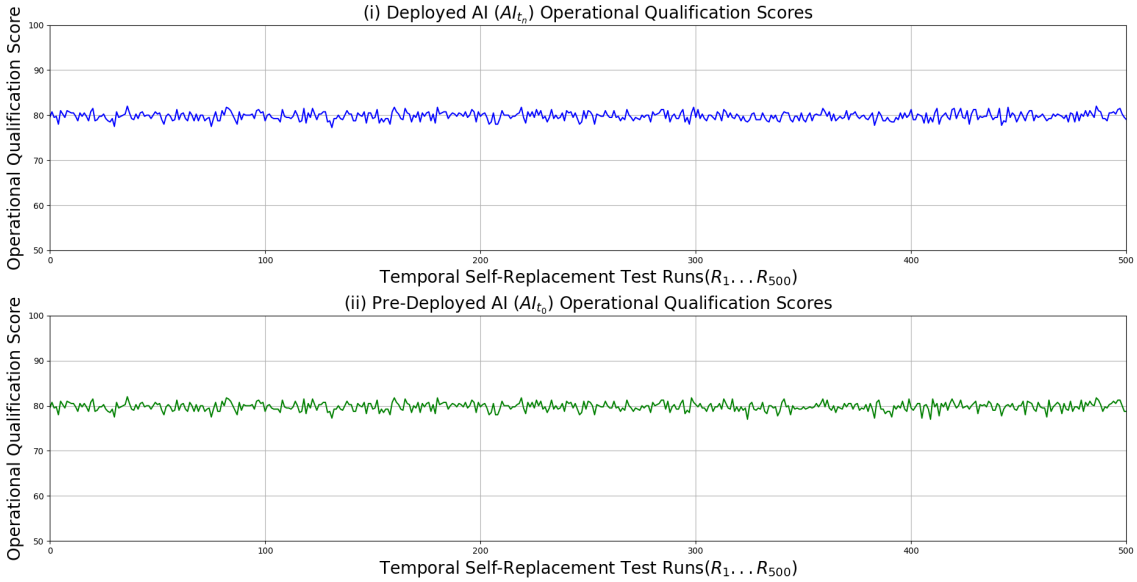


Fig. 2: *The SUE passes the temporal self-replacement tests. Plots in sub-figures (i) and (ii) show matching operational qualification scores of the tested AI system at t_n and t_0 across 500 runs of the temporal self-replacement test.*

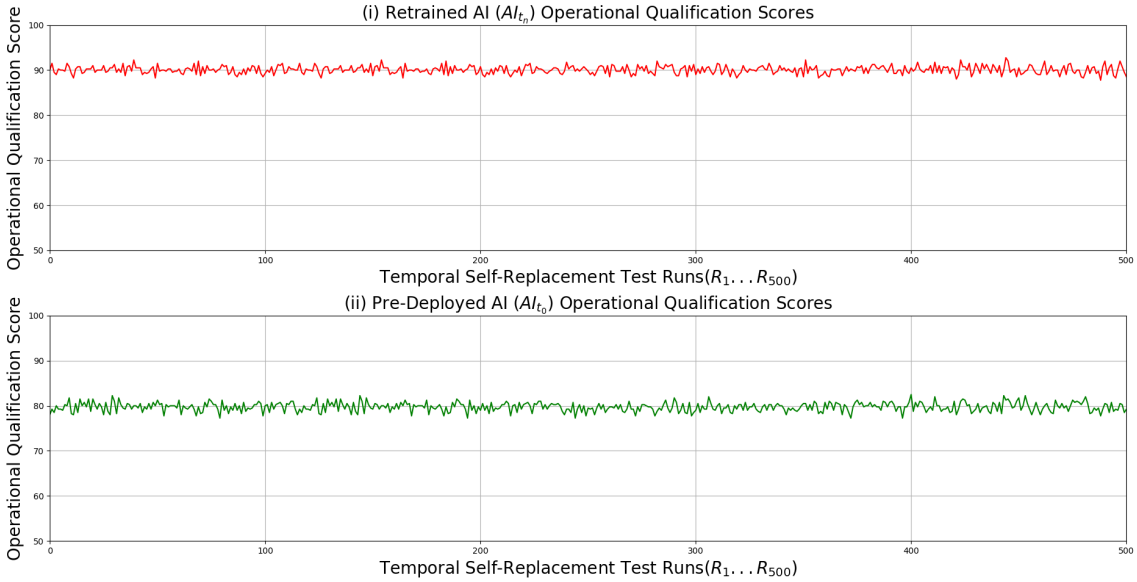


Fig. 3: *The SUE fails in the temporal self-replacement tests. Plots in sub-figures (i) and (ii) show the operational qualification scores for the examined AI system from 500 runs of temporal self-replacement test significantly differ pre and post-deployment.*

Answer to RQ1: Our evaluation of the face and gender detection AI system demonstrates SFAIR's potential to assist regulatory authorities in the recurrent examination of operational qualifications of AI systems and the attestation of their regulatory fitness through effective, automated, and recurrent measurements.

RQ2: Can SFAIR's security measures help counter the threats identified in the threat model in Section III?

SFAIR employs multiple mechanisms to mitigate the threats identified in Section III and illustrated in Figure 4. First, hardcoded identities are vulnerable to identity theft, which enables evading the identification check before the temporal self-replacement test. To counter this threat, SFAIR uses

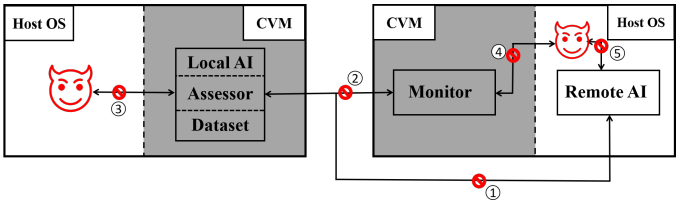


Fig. 4: *SFAIR Attacks:*① *Indicates guessing-based and replay attacks by the AI provider or third parties attempting to predict or reuse valid data for spoofing.* ② *Represents attacks by third parties trying to eavesdrop or intervene in the information exchanged between the monitor and assessor, compromising data integrity.* ③ *and* ④ *Denote attackers gaining control of the host OS to interfere with the data inside Confidential Virtual Machines (CVM), manipulate or corrupt data, or execute malicious code.* ⑤ *Indicates attacks that try to alter the post-deployment behavior of an AI, modifying its functions or actions maliciously.*

dynamically generated signatures consisting of one hundred, six-digit strings of values using the scheme proposed in Section IV-A to establish and verify the AI's identity. We verified the effectiveness of the identification scheme through more than 1500 experiments involving unaltered and altered SUE. Alterations were made through retraining the SUE. If an adversary wants to break this scheme, he must correctly guess the signatures. However, there are 10^6 possible values for each of the hundred strings constituting the signatures. Further, to successfully guess a signature, the adversary must correctly guess the sequence of one hundred six digit strings in a 10^{600} search space in a timely manner, which is infeasible. In addition, any advantage of the adversary in guessing-based attacks (III-B1,III-B2,III-B4) gained during an unsuccessful attack is nullified in the successive attempt due to the randomness of signatures used across different runs of the test.

Referring to ① in Figure 4, SFAIR blocks guessing-based and replay attacks on its integrity. A single run of the temporal self-replacement test involves input sequences comprising randomly selected test inputs. If each property test randomly selects k inputs from a cache of size n, the probability that an adversary correctly guesses the input vector of size k is nCk. Our experiments use 100 random inputs to examine each operational trait of the SUE. Given the availability of 500 test inputs per property in the test cache, the likelihood of correctly guessing a sequence of test input vectors for one property is $\frac{1}{2.0417\times10^{107}}$, demonstrating a very low probability. Considering the additional processing involved in recognizing the input files, the probability of a successful guess further decreases. The randomization of inputs across different runs and their pseudonymization through masking further help thwart the efforts of the malicious actors (III-B2, III-B4), aiming to disrupt the temporal self-replacement tests.

Referring to ② in Figure 4, SFAIR blocks attacks by third parties, eavesdropping, or intervening in the information exchanged between the monitor and assessor, and attempting to compromise confidentiality and data integrity. This is achieved by establishing a dedicated secure channel between the assessor and monitor using Advanced Encryption Standard (AES). The key size is 128, requiring an adversary to perform 2^{128} operations to break the encryption, which is infeasible for effective attacks [66]. This defense can be further strengthened by periodically replacing old keys with new ones. The repetitive information exchange between the assessor and monitor, and the timestamped local logs generated by the monitor, provide extra protection to the discoveries made by the monitor.

Referring to ③ and ④ in Figure 4, SFAIR provides protection against attackers gaining control of the host OS to interfere with its operations by manipulating or corrupting the data or interfering with the code in use. The SEV-ES technology used to fortify SFAIR encrypts CVM memory and makes it inaccessible to the hypervisor [55]. The encryption is performed using keys unique to each CVM. AMD Secure Processor (SP) manages these keys and re-generates them whenever a CVM is reset. Moreover, during global switches, when execution control is passed between the CVM and host, SEV-ES encrypts the memory and CPU register content when the CVM stops running. This prevents the hypervisor from reading sensitive state information of the CVM or affecting its control flow. In short, the memory image and initial CPU register state encryption blocks sophisticated memory scrapping-based replay attacks, cold boot attacks, and code injection attacks [44]. Finally, during initialization, AMD-SP cryptographically measures the correctness of the memory image and CPU register state. This step provides extra assurance that CVMs have not been tampered with. We rely on AMD's assurances for the correctness of these measures.

Referring to ⑤ in Figure 4, in attacks that try to alter the post-deployment behavior of an AI, modifying its functions or actions maliciously, SFAIR cannot prevent such attacks, but it can successfully detect them. To evaluate SFAIR's ability to detect unauthorized changes in the AI system, we retrained the model using an extended dataset, including additional images from the LFW [60] dataset. Figure 3 shows that SFAIR successfully detected the unauthorized changes in the SUE. Likewise, to test the monitor's effectiveness, we replaced the original model with an unregistered one and changed the version of the model hosting platform. The monitor detected these changes in each experiment and alerted the assessor.

Answer to RQ2: The security analysis and experimental evaluation suggest that SFAIR's security measures, designed considering the secure-by-design principle, can effectively counter the attacks identified in Section III.

RQ3: How does deploying SFAIR in CVMs affect its performance?

To address **RQ3**, we perform the Wilcoxon signed-rank test (WSRT) to evaluate the difference among the SFAIR execution times measured across 1000 experiments conducted with and

without enabling SEV-ES features (500 experiments each). The test returns W-value: 20464, RBC (rank-biserial correlation): 0.67323, and CLES (common language effect size): 0.789412, and p-value: $6.19x10^{-39}$, suggesting that in at least 78% cases, execution time is higher when SEV-ES features are enabled. Further, Table II presents the minimum (Min), the maximum (Max), the mean execution time values, and the standard deviation (SD). These statistics suggest a modest difference in execution times between running SFAIR with and without hardware-based protection.

TABLE II: Execution Time

SEV-ES Features	Min	Max	Mean	SD
Enabled	92.06	95.63	93.46	0.60
Disabled	90.83	95.98	92.71	0.82

Answer to RQ3: The experimental results and subsequent statistical analysis reveal that introducing hardware-based protection in SFAIR incurs modest performance overhead.

VII. Discussion

In this section, we first establish the generality of our approach and methods. Then we benchmark SFAIR against state-of-the-art AI monitoring frameworks and discuss its computational complexity and practical challenges.

A. Generalizability of our Approach and Methods

While various AI regulatory frameworks introduced by different countries are inherently limited by their geographic jurisdiction, they exhibit significant commonalities (please see Section II). Motivated by these shared principles, we propose a generalized methodology to serve the needs of assessing the regulatory fitness of high-risk AI systems. Accordingly, the workflow of the proposed approach is inspired by the European Digital Identity Regulation [45] and the European AI Act [5]. However, the core assessment requirements to determine the regulatory fitness of an AI system are synthesized from a careful review of AI regulations in China [10], the EU [5], Great Britain [12], the United States [11], and related literature. However, it is important to note that, in their construction, the approaches, concepts, and methods proposed in this work are agnostic to a single regulatory framework or underlying technology of AI systems and are systematically designed to meet the assessment requirements of diverse regulations, while considering the regulatory evolution and a broad range of AI systems.

For example, the concept of operational qualification of high-risk AI systems proposed in this research, used to determine their regulatory fitness, is applicable to diverse AI systems, including machine learning-based and autonomous systems, deployed in different operational settings and across different jurisdictions. Likewise, the temporal self-replacement test proposed in this work (Section IV) is designed to serve as a generic methodological framework to examine the critical operational behaviors of high-risk AI systems and determine their regulatory fitness. The test is flexible in its construction and is designed to accommodate the emerging regulatory requirements. In this regard, we demonstrate the effectiveness of the temporal self-replacement test in assessing and attesting the regulatory fitness of a gender and face detection high-risk AI system using four properties: operational accuracy, operational fairness, operational efficiency, and operational robustness (Section IV). However, in practice, a different set of properties, critical for determining the regulatory behaviors of AI systems in a given jurisdiction, can be used to assess the regulatory fitness of a given high-risk AI system. The selection of such properties has to be necessarily aligned with the regulatory requirements for the jurisdiction where the AI system operates.

Finally, the threat model defined in this paper (Section III) is both practical and designed to guide future research on securing AI regulatory infrastructures. Likewise, the security mechanisms we designed to secure SFAIR (Section V) are directly applicable to real-world implementations.

TABLE III: Comparison with State of the Art

Framework	Focus	Security Concern	Design Flexibility	Conflict of Interest	Target AI[1]	Covered Properties	Solves Oracle Problem
SageMaker Clarify [13]	QA	✗	✗	✓	ML	Bias & Explanations	✗
SageMaker Model Monitor [15]	QA	✗	✗	✓	ML	Data & Concept Drift	✗
ComplAI [16]	Explanation	✗	✗	✓	Supervised ML	Explainability, Drift Susceptibility, Robustness, Fairness, Trust	✗
AITEST [14]	QA	✗	✓	✓	ML	Accuracy, Fairness, Robustness	✗
VertexAI [17]	QA	✗	✗	✓	ML	Feature Skew & Data Drift	✗
SFAIR	Regulation	Secure-by-design	✓	✗	ML & AS	Identity, Operational Accuracy, Operational Fairness, Operational Efficiency, Operational Robustness, Operational Qualification	✓

[1]QA=Quality Assurance,ML=Machine Learning-Based Systems, AS= Autonomous Systems, ✓=Presence, ✗= Absence

B. Comparison of SFAIR with State of the Art

Table III illustrates that SFAIR distinguishes itself from existing post-deployment AI monitoring frameworks [13]–[17] in several key aspects. First, SFAIR meets stringent regulatory requirements, including a security-focused design and offering flexibility to accommodate regulatory changes. It intends to help avoid conflicts of interest and has been designed to regulate a broader spectrum of high-risk AI systems, including machine learning-based and autonomous systems. Furthermore, compared to existing AI monitoring frameworks, SFAIR demonstrates greater generality in examining and attesting the high-risk AI system's operational behaviors, which have been selected carefully after reviewing the major AI regulations [5], [10]–[12]. SFAIR examines high-risk AI systems by conducting the *temporal self-replacement test* and leverages its measurements to calculate the operational qualification score, which is used as a pragmatic measure of a given system's regulatory fitness. Conforming to the requirements of pragmatic measures [67], it is important for all stakeholders, inexpensive, actionable, and sensitive to change. Accordingly, changes in the operational qualification score of a high-risk AI system can guide the decisions of regulatory authorities, AI providers, and AI users. Likewise, our experiments demonstrate that the operational qualification score is inexpensive to calculate (please refer to Table II), can guide stakeholder actions, e.g., approval/disapproval and improvement of the system, and is sensitive to change (please refer to Figures 2 and 3).

Plots in Figures 5 and 6 demonstrate the explainability, sensitivity to change, and diagnosability of SFAIR. By diagnosability, we mean its ability to suggest the causes of low operational qualification scores or changes in post-deployment operational behaviors. For example, the plots in Figure 6 reflect SUE's consistently superior operational behavior compared to pre-deployment tests, warning that unauthorized changes to the AI, such as retraining, might have been made. Additionally, SFAIR allows using alternative testing methods or integrating tests to examine and attest new AI qualifications emerging due to AI evolution and relevant regulatory requirements. It is because SFAIR is developed on top of a systematically designed methodological framework, the temporal self-replacement test, which allows testing any number of relevant operational behaviors to examine an AI's operational qualifications and determine its regulatory fitness.

Likewise, contrary to the existing AI monitoring frameworks that offer no security guarantees, SFAIR is a working example of the secure-by-design principle, which advocates integrating security measures into the blueprint of critical systems [29]. Finally, leveraging the temporal self-replacement test, SFAIR is the only framework that solves the oracle problem, removing the need for human intervention to determine if a test has passed or failed [53]. We also want to point out that we deliberately avoided discussing various commercial AI monitoring solutions since they have shortcomings similar to those listed in Table III. In particular, their use in AI regulation introduces conflicts of interest.

C. Computational Complexity

The overall per-test time complexity of SFAIR is $O(PI(D+N))$, which is primarily dominated by the network and I/O operations. Here, P is the number of properties, I is the number of input files per property, N is the network processing overhead, D is the number of datasets (subfolders), and F is the number of files per subfolder. The space complexity for storing results and intermediate data is $O(PI)$.

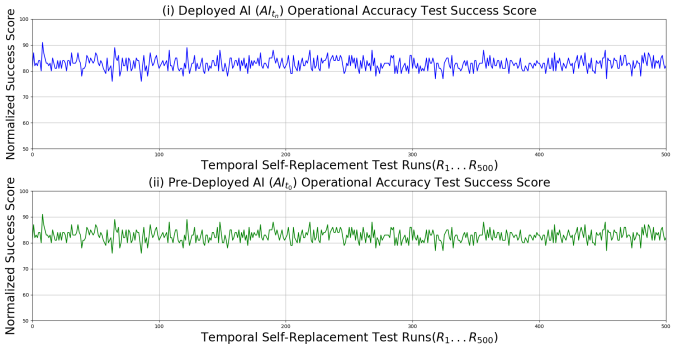
SFAIR operations are inherently network-intensive. The cost of network requests, at $O(IN)$ per property, can be the primary performance bottleneck. While this work does not prioritize optimization, techniques such as batch and parallel processing could improve system efficiency. Overall performance is directly impacted by external factors, including network latency, model inference time, filesystem performance during input selection, and the number of tests run simultaneously.

D. Challenges

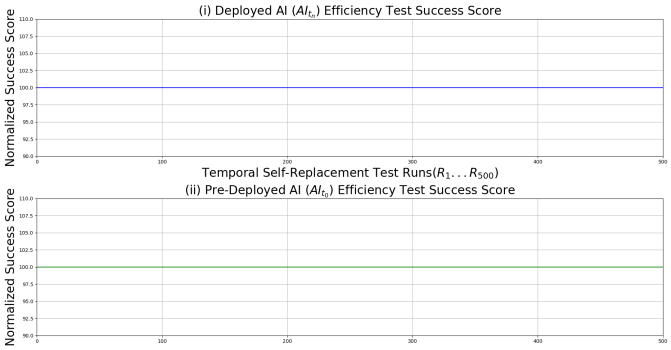
SFAIR has some practical challenges. For example, maintaining a copy of the high-risk AI system mirroring its originally approved version is a challenging task. However, this design choice is motivated by widely used cross-referencing-based approaches [33] and is essential for regulatory oversight and security requirements. Such approaches have previously been used to detect intellectual property infringement and vulnerability propagation by comparing the performance of two AI systems on the same set of inputs [51], [68]. One of the solutions to facilitate this process can be standardizing the runtime environment requirements. Nevertheless, AI systems with significant resource demands may require alternative approaches guided by the *temporal self-replacement test*. For example, initial testing followed by periodic evaluations and benchmarking using baseline values can provide a viable alternative. However, it can be vulnerable to replay and guessing-based attacks, given the possible curiosity of involved parties and advancing capabilities of AI systems [33]. This particularly applies to large language models like ChatGPT. Finally, SFAIR is not a silver bullet. We propose conducting regular post-deployment audits alongside recurrent qualification testing. SFAIR supports such audits by logging all its discoveries.

VIII. Conclusions

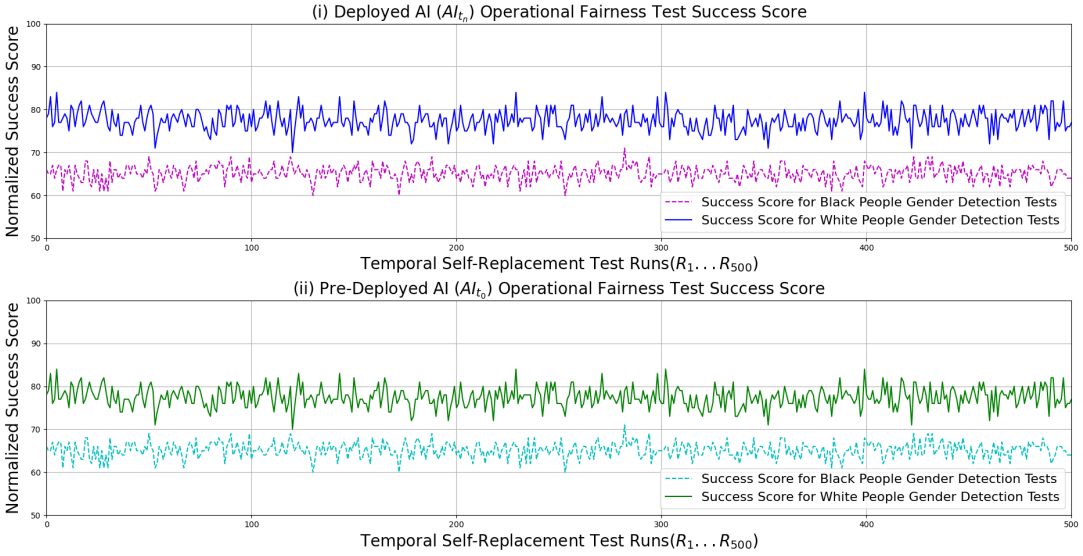
Governments worldwide are introducing regulatory measures to regulate high-risk AI systems. Yet, the technical infrastructure to enforce these regulations is currently missing. In this paper, we identified key challenges in developing technical frameworks for AI systems' regulation and proposed conceptual, methodological, and practical solutions to address these challenges. We introduced the concepts of AI's operational qualification and proposed the 'temporal self-replacement test' for pre- and post-deployment assessment of an AI's operational qualification. We introduced the operational qualification score as a pragmatic measure of an AI system's regulatory fitness. In addition, we designed and developed SFAIR, a tool for automatic, recurrent, and secure examination of an AI's operational qualification and attestation of its regulatory fitness, leveraging



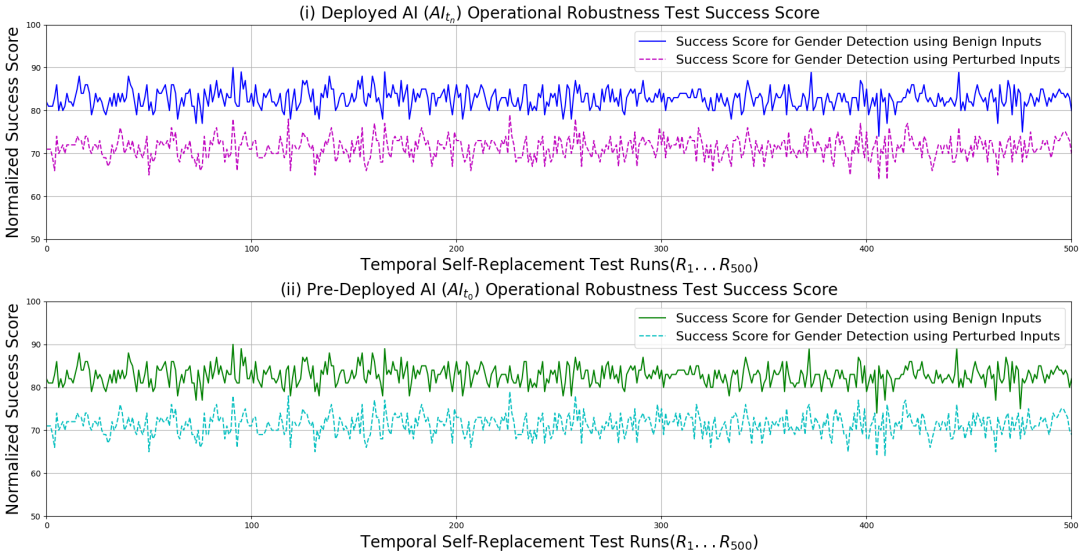
(a) *Operational accuracy*



(b) *Operational efficiency*

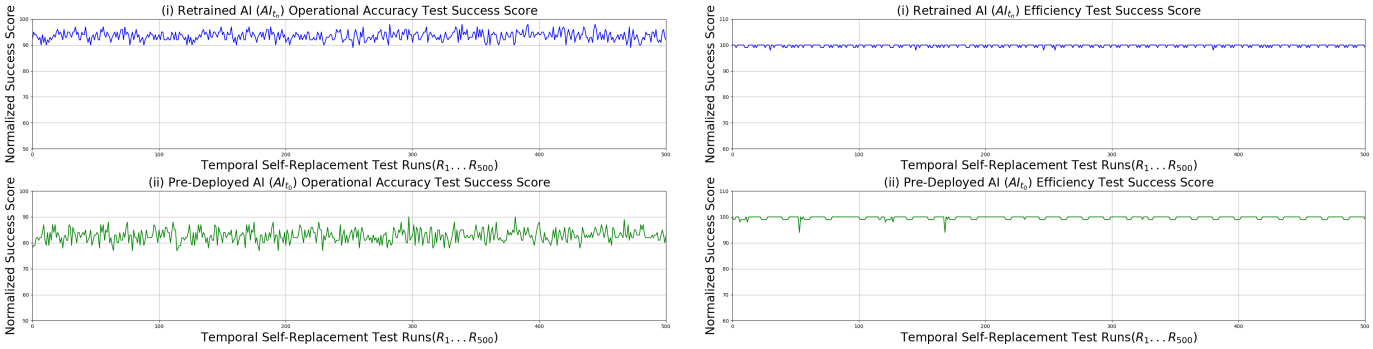


(c) *Operational fairness*

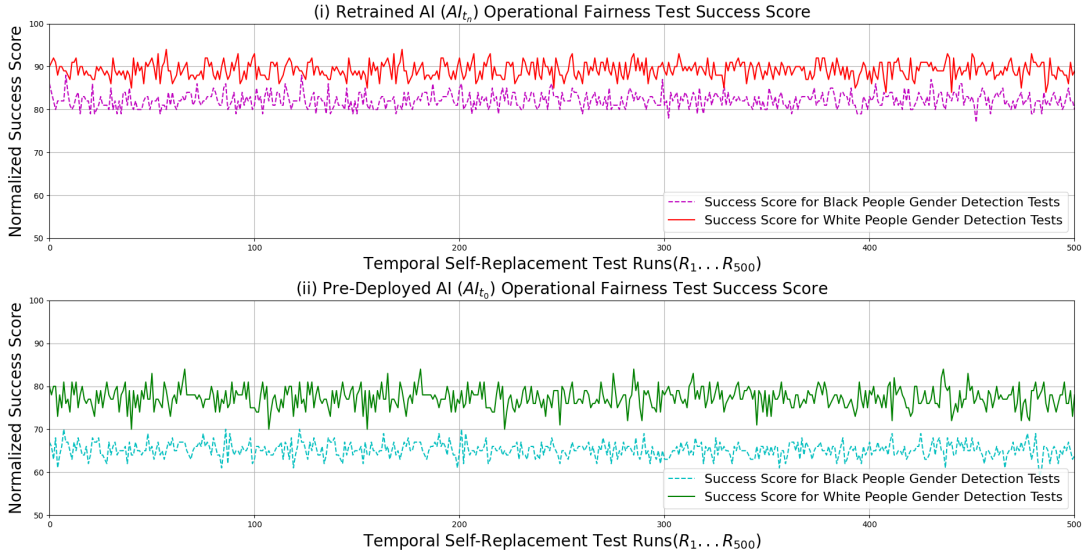


(d) *Operational robustness*

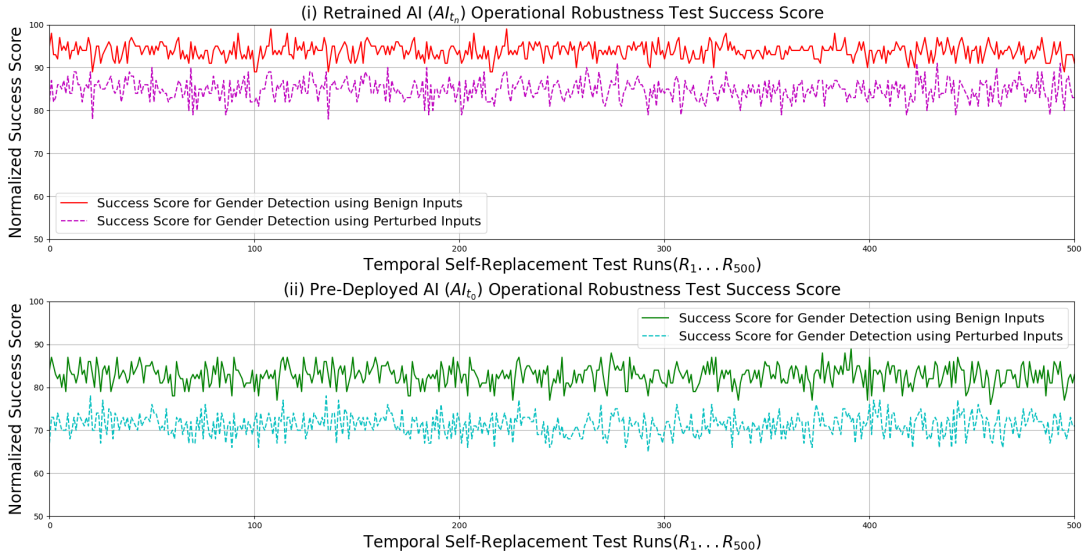
Fig. 5: *Identical plots for pre and post-deployment success scores for individual properties examined under successful temporal self-replacement tests explain how the system under test has preserved its tested operational qualifications post-deployment. In addition, plots in 5c and 5d show consistently low scores for operational fairness and operational robustness.*



(a) *Operational accuracy* (b) *Operational efficiency*



(c) *Operational fairness*



(d) *Operational robustness*

Fig. 6: *Plots for individual properties tested under failed temporal self-replacement tests show clear mismatches among relevant success scores pre and post-deployment and reflect on the sensitivity to change and diagnosability of the proposed temporal self-replacement test.*

the proposed test and measure. SFAIR's design is driven by regulatory requirements, and it is a working example of the secure-by-design principle. We introduced a new threat model for AI regulatory tools and mitigated the identified threats in SFAIR's design. We validated the efficacy of the temporal self-replacement test and the practical utility of SFAIR by demonstrating its capability to support regulatory authorities in automated, recurrent, and secure AI qualification examination and attestation of its regulatory fitness using an open-source, high-risk AI system. Finally, we make the SFAIR prototype publicly available to address the conflicts of interest.

REFERENCES

[1] D. Lo, "Trustworthy and synergistic artificial intelligence for software engineering: Vision and roadmaps," in *2023 IEEE/ACM International Conference on Software Engineering: Future of Software Engineering (ICSE-FoSE)*. IEEE, May 2023.

[2] J. Sifakis, "Testing system intelligence," *arXiv preprint arXiv:2305.11472*, May 2023.

[3] D. Vela, A. Sharp, R. Zhang, T. Nguyen, A. Hoang, and O. S. Pianykh, "Temporal quality degradation in AI models," *Scientific Reports*, vol. 12, no. 1, Jul. 2022.

[4] A. Kuppa and N.-A. Le-Khac, "Adversarial XAI methods in cybersecurity," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 4924–4938, 2021.

[5] The European Parliament and the Council of the European Union. (2024, Jun.) Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence and amending Regulations (EC) No 300/2008, (EU) No 167/2013, (EU) No 168/2013, (EU) 2018/858, (EU) 2018/1139 and (EU) 2019/2144 and Directives 2014/90/EU, (EU) 2016/797 and (EU) 2020/1828 (Artificial Intelligence Act) (Text with EEA relevance). [Online]. Available: http://data.europa.eu/eli/reg/2024/1689/oj

[6] S. R. Steven Posada, Gregory Magno. (2022, Jun.) Subject vehicle crashes with in-road or roadside first responders. Accessed on May 21, 2024. [Online]. Available: https://static.nhtsa.gov/odi/inv/2022/INOA-EA22002-3184.PDF

[7] K. Stacey and D. Milano. (2023, Nov.) UK, US, EU and China sign declaration of AI's 'catastrophic' danger. The Guardian. Accessed on Dec 19, 2023. [Online]. Available: https://www.theguardian.com/technology/2023/nov/01/uk-us-eu-and-china-sign-declaration-of-ais-catastrophic-danger

[8] E. Drage and K. Mackereth, "Does AI debias recruitment? race, gender, and AI's "eradication of difference"," *Philosophy & Technology*, vol. 35, no. 4, Oct. 2022.

[9] M. Anderljung, J. Barnhart, A. Korinek, J. Leung, C. O'Keefe, J. Whittlestone, S. Avin, M. Brundage, J. Bullock, D. Cass-Beggs, B. Chang, T. Collins, T. Fist, G. Hadfield, A. Hayes, L. Ho, S. Hooker, E. Horvitz, N. Kolt, J. Schuett, Y. Shavit, D. Siddarth, R. Trager, and K. Wolf, "Frontier AI regulation: Managing emerging risks to public safety," *arXiv preprint arXiv: 2307.03718*, Jul. 2023.

[10] J. H. Chirstopher Ferguson. (2023, Aug.) China's new rules for generative AI: An emerging regulatory framework. Accessed Dec 1, 2023. [Online]. Available: https://www.fasken.com/en/knowledge/2023/08/chinas-new-rules-for-generative-ai

[11] White House. (2020, Nov.) Guidance for regulation of artificial intelligence applications in memorandom for the heads of executive departments and agencies. Executive Office of the President, Office of Management and Budget. Accessed Dec 1, 2023. [Online]. Available: https://www.whitehouse.gov/wp-content/uploads/2020/11/M-21-06.pdf

[12] The Secretary of State for Science, Innovation and Technology, *A Pro-innovative Approach to AI Regulation*, Department for Science, Innovation & Technology, Ed. HH Associates Ltd., 2023.

[13] M. Hardt, X. Chen, X. Cheng, M. Donini, J. Gelman, S. Gollaprolu, J. He, P. Larroy, X. Liu, N. McCarthy, A. Rathi, S. Rees, A. Siva, E. Tsai, K. Vasist, P. Yilmaz, M. B. Zafar, S. Das, K. Haas, T. Hill, and K. Kenthapadi, "Amazon SageMaker Clarify: Machine learning bias detection and explainability in the cloud," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, ser. KDD '21. ACM, Aug. 2021.

[14] A. Aggarwal, S. Shaikh, S. Hans, S. Haldar, R. Ananthanarayanan, and D. Saha, "Testing framework for black-box AI models," in *2021 IEEE/ACM 43rd International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*. IEEE, May 2021.

[15] D. Nigenda, Z. Karnin, M. B. Zafar, R. Ramesha, A. Tan, M. Donini, and K. Kenthapadi, "Amazon SageMaker Model Monitor: A system for real-time insights into deployed machine learning models," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD '22. ACM, Aug. 2022.

[16] A. De, S. S. Gudipudi, S. Panchanan, and M. S. Desarkar, "ComplAI: Framework for multi-factor assessment of black-box supervised machine learning models," in *Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing*, ser. SAC '23. ACM, Mar. 2023.

[17] Google. (2024, Apr.) Introduction to Vertex AI model monitoring. Google LLC. Accessed Dec 2, 2023. [Online]. Available: https://cloud.google.com/vertex-ai/docs/model-monitoring/overview

[18] S. Amershi, A. Begel, C. Bird, R. DeLine, H. Gall, E. Kamar, N. Nagappan, B. Nushi, and T. Zimmermann, "Software engineering for machine learning: A case study," in *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*. IEEE, May 2019.

[19] J. Hernández-Orallo, "Evaluation in artificial intelligence: from task-oriented to ability-oriented measurement," *Artificial Intelligence Review*, vol. 48, no. 3, pp. 397–447, Aug. 2016.

[20] J. M. Zhang, M. Harman, L. Ma, and Y. Liu, "Machine learning testing: Survey, landscapes and horizons," *IEEE Transactions on Software Engineering*, vol. 48, no. 1, pp. 1–36, Jan 2022.

[21] Department for Science, Innovation & Technology, United Kingdom Government, "Cyber security risks to artificial intelligence," 2024, accessed: [2024-10-12]. [Online]. Available: https://www.gov.uk/government/publications/research-on-the-cyber-security-of-ai/cyber-security-risks-to-artificial-intelligence

[22] A. Paleyes, R.-G. Urma, and N. D. Lawrence, "Challenges in deploying machine learning: A survey of case studies," *ACM Computing Surveys*, vol. 55, no. 6, pp. 1–29, dec 2022.

[23] D. McCabe. (2024, Jan.) Federal Trade Commission launches inquiry into A.I. deals by tech giants. Accessed May 2, 2024. [Online]. Available: https://www.nytimes.com/2024/01/25/technology/ftc-ai-microsoft-amazon-google.html

[24] V. Graham. (2023, Sep.) FTC sues Amazon for illegally maintaining monopoly power. Accessed May 2, 2024. [Online]. Available: https://www.ftc.gov/news-events/news/press-releases/2023/09/ftc-sues-amazon-illegally-maintaining-monopoly-power

[25] D. Milmo. (2024, May) OpenAI putting 'shiny products' above safety, says departing researcher. Accessed May 18, 2024. [Online]. Available: https://tinyurl.com/y9mtt4vx

[26] J. Constine. (2019, Jan.) Facebook pays teens to install VPN that spies on them. Accessed Dec 2, 2023. [Online]. Available: https://techcrunch.com/2019/01/29/facebook-project-atlas/

[27] J. Mökander, M. Axente, F. Casolari, and L. Floridi, "Conformity assessments and post-market monitoring: a guide to the role of auditing in the proposed European AI regulation," *Minds and Machines*, vol. 32, no. 2, pp. 241–268, Jun. 2022.

[28] S. Hong, P. Frigo, Y. Kaya, C. Giuffrida, and T. Dumitras, "Terminal brain damage: Exposing the graceless degradation in deep neural networks under hardware fault attacks," in *28th USENIX Security Symposium (USENIX Security 19)*. Santa Clara, CA: USENIX Association, Aug. 2019, pp. 497–514.

[29] CISA, NSA, FBI, ACSC, CCCS, CERT NZ, NCSC-UK, BSI, NCSC-NL, NCSC-NO, NUKIB, NICD, KISA, NISC-JP, JPCERT/CC, CSA, and CSIRTAMERICAS, "Shifting the balance of cybersecurity risk: Principles and approaches for secure by design software," Cybersecurity & Infrastructure Security Agency, Tech. Rep., Oct. 2023. [Online]. Available: https://www.cisa.gov/sites/default/files/2023-10/SecureByDesign_1025_508c.pdf

[30] Arun Ponnusamy. (2023) Gender detection (from scratch) using deep learning with Keras and CVLib. GitHub. Accessed Jan 12, 2024. [Online]. Available: https://github.com/arunponnusamy/cvlib/releases/download/v0.2.0/gender_detection.model

[31] A. M. Turing, "Computing machinery and intelligence," *Mind*, vol. 59, no. 236, pp. 433–460, 1950.

[32] M. Zhang, Y. Zhang, L. Zhang, C. Liu, and S. Khurshid, "DeepRoad: GAN-based metamorphic testing and input validation framework for autonomous driving systems," in *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering*, ser. ASE '18. ACM, Sep. 2018.

[33] A. Aleti, "Software testing of generative AI systems: Challenges and opportunities," in *2023 IEEE/ACM International Conference on Software Engineering: Future of Software Engineering (ICSE-FoSE)*. IEEE, May 2023.

[34] K. Grosse, L. Bieringer, T. R. Besold, and A. M. Alahi, "Towards more practical threat models in artificial intelligence security," in *33rd USENIX Security Symposium (USENIX Security 24)*, Aug. 2024, pp. 1–18, USENIX Security '24 Fall Accepted Paper.

[35] G. Deng, Y. Liu, Y. Li, K. Wang, Y. Zhang, Z. Li, H. Wang, T. Zhang, and Y. Liu, "Masterkey: Automated jailbreaking of large language model chatbots," in *Proceedings 2024 Network and Distributed System Security Symposium*, ser. NDSS 2024. Internet Society, 2024.

[36] D. Coldewey. (2023, Dec.) Google's best gemini demo was faked. Accessed on December 18, 2023. [Online]. Available: https://techcrunch.com/2023/12/07/googles-best-gemini-demo-was-faked/

[37] S. Josefsson. (2023, Apr.) A security device threat model: The substitution attack. Accessed on December 18, 2023. [Online]. Available: https://blog.josefsson.org/2023/04/27/a-security-device-threat-model-the-substitution-attack/

[38] P. Kruus and D. Challener, "On reporting of the time of attestation measurements," in *MILCOM 2015-2015 IEEE Military Communications Conference*. IEEE, 2015, pp. 354–359.

[39] T. Bove. (2023, Apr.) Amazon is taking on Google and Microsoft with its own generative A.I. as CEO Andy Jassy says it will be a 'big deal' for the company. Fortune.com. Accessed Jun 20, 2024. [Online]. Available: https://fortune.com/2023/04/13/amazon-joins-ai-race-google-microsoft-cloud-services-aws/

[40] I. Shevrin and O. Margalit, "Detecting Multi-Step IAM attacks in AWS environments via model checking," in *32nd USENIX Security Symposium (USENIX Security 23)*, Aug. 2023, pp. 6025–6042.

[41] European Union Agency for Cybersecurity, *ENISA threat landscape 2024: July 2023 to June 2024.*, I. Lella, M. Theocharidou, E. Magonara, A. Malatras, R. S. Naydenov, C. Ciobanu, and G. Chatzichristos, Eds. European Union Agency for Cybersecurity (ENISA), 2024.

[42] C. Cimpanu. (2019, Feb.) Microsoft: 70 percent of all security bugs are memory safety issues. Accessed on December 18, 2023. [Online]. Available: https://www.zdnet.com/article/microsoft-70-percent-of-all-security-bugs-are-memory-safety-issues/

[43] Lightspin. (2021, Dec) Lightspin research team discovers cross-account access vulnerability on AWS SageMaker jupiter notebook instance. Accessed Dec 15, 2023. [Online]. Available: https://www.prnewswire.com/news-releases/

[44] Confidential Computing Consortium, "A technical analysis of confidential computing," Confidential Computing Consortium, Tech. Rep. V1.3, Nov. 2022. [Online]. Available: https://confidentialcomputing.io/wp-content/uploads/sites/10/2023/03/CCC-A-Technical-Analysis-of-Confidential-Computing-v1.3_unlocked.pdf

[45] The European Parliament and the Council of the European Union, "Regulation (EU) 2024/1183 of the EUROPEAN PARLIAMENT and of THE COUNCIL," *Official Journal of the European Union*, Jun. 2024, amending Regulation (EU) No 910/2014 as regards establishing the European Digital Identity Framework. [Online]. Available: http://data.europa.eu/eli/reg/2024/1183/oj

[46] J. Lin, "Divergence measures based on the shannon entropy," *IEEE Transactions on Information theory*, vol. 37, no. 1, pp. 145–151, 1991.

[47] R. Dodhiawala, N. S. Sridharan, P. Raulefs, and C. Pickering, "Real-time AI systems: a definition and an architecture," in *Proceedings of the 11th International Joint Conference on Artificial Intelligence - Volume 1*, ser. IJCAI'89. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1989, p. 256–261.

[48] Parliament of the United Kingdom. (2018) The Data Protection Act 2018. Accessed May 18, 2024. [Online]. Available: https://www.legislation.gov.uk/ukpga/2018/12/contents/enacted

[49] European Union Agency for Fundamental Rights. (2024) EU Charter of Fundamental Rights. Accessed May 18, 2024. [Online]. Available: https://fra.europa.eu/en/eu-charter/article/21-non-discrimination

[50] R. Fibbi, A. H. Midtbøen, and P. Simon, *Concepts of Discrimination*. Springer International Publishing, 2021, pp. 13–20.

[51] Y. Li, Z. Zhang, B. Liu, Z. Yang, and Y. Liu, "ModelDiff: testing-based DNN similarity comparison for model reuse detection," in *Proceedings of the 30th ACM SIGSOFT International Symposium on Software Testing and Analysis*. ACM, Jul 2021.

[52] V. Pamnani. (2017, Apr.) Enforce password history. Accessed January 15, 2024. [Online]. Available: https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-10/security/threat-protection/security-policy-settings/enforce-password-history

[53] E. T. Barr, M. Harman, P. McMinn, M. Shahbaz, and S. Yoo, "The oracle problem in software testing: A survey," *IEEE Transactions on Software Engineering*, vol. 41, no. 5, pp. 507–525, May 2015.

[54] D. P. Mulligan, G. Petri, N. Spinale, G. Stockwell, and H. J. M. Vincent, "Confidential computing—a brave new world," in *2021 International Symposium on Secure and Private Execution Environment Design (SEED)*. IEEE, Sep. 2021.

[55] D. Kaplan, "Protecting VM register state with SEV-ES," Advanced Micro Devices, Tech. Rep., Feb. 2017.

[56] Advanced Micro Devices, Inc. (2025) AMD Secure Encrypted Virtualization (SEV). Accessed Sep 13, 2025. [Online]. Available: https://www.amd.com/en/developer/sev.html

[57] A. Danial, "cloc: v1.92," Dec. 2021, Accessed Oct 10, 2024. [Online]. Available: https://doi.org/10.5281/zenodo.5760077

[58] D. Veillard. Libvirt virtualization api. Accessed May 2, 2024. [Online]. Available: https://libvirt.org/

[59] Visage Technologies. Gender recognition software. Accessed Nov 10, 2024. [Online]. Available: https://visagetechnologies.com/gender-detection/

[60] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," University of Massachusetts, Amherst, Tech. Rep. 07-49, October 2007.

[61] B. Chesneau, K. Kapustin, R. Leeds, B. Peksag, J. Madden, and B. Randall. (2023) Gunicorn. Accessed Jan 20, 2024. [Online]. Available: https://gunicorn.org/

[62] T. Maho, T. Furon, and E. L. Merrer, "Fingerprinting classifiers with benign inputs," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 5459 – 5472, Aug. 2023.

[63] K. Seshadri and M. Savvides, "Towards a unified framework for pose, expression, and occlusion tolerant automatic facial alignment," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 10, pp. 2110–2122, Oct. 2016.

[64] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, Dec. 2014.

[65] A. Alwarafy, K. A. Al-Thelaya, M. Abdallah, J. Schneider, and M. Hamdi, "A survey on security and privacy issues in edge-computing-assisted internet of things," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4004–4022, Mar. 2021.

[66] N. Mouha and M. Dworkin, *Review of the advanced encryption standard*. Gaithersburg, MD: National Institute of Standards and Technology, Jul. 2021, no. NIST.IR.8319.

[67] R. E. Glasgow and W. T. Riley, "Pragmatic measures," *American Journal of Preventive Medicine*, vol. 45, no. 2, pp. 237–243, Aug. 2013.

[68] Z. Deng, G. Meng, K. Chen, T. Liu, L. Xiang, and C. Chen, "Differential testing of cross deep learning framework APIs: Revealing inconsistencies and vulnerabilities," in *32nd USENIX Security Symposium (USENIX Security 23)*. Anaheim, CA: USENIX Association, Aug. 2023, pp. 7393–7410.

Haroon Elahi (Member, IEEE) is with the Chalmers University of Technology and the University of Gothenburg, Sweden. Previously, he was a visiting research scholar at the Department of Computer Science and Engineering, Southern University of Technology, China, where his research focused on AI regulation. He completed his doctoral degree at the School of Computer Science, Guangzhou University, China. After that, he held a postdoctoral position at Umeå University, Sweden. His research interests include identifying and solving privacy, security, and trust issues of emerging technologies in the rapidly changing privacy and security threat landscapes. He has published over twenty articles and conference papers at reputed venues, including IEEE TCSS, IEEE IoT Journal, IEEE TVT, Information Sciences, and Neurocomputing.

Nian Liu received MS degree from the Department of Computer Science and Engineering, Southern University of Technology, China in 2024. Currently, she is a Research Assistant at the Computer and Systems Security Lab (COMPASS), Department of Computer Science and Engineering, Southern University of Technology, China. Her research interests include systems security, with a focus on confidential computing.

Jiatong Chen is a master's student at the Department of Computer Science and Engineering, Southern University of Science and Technology, China and affiliated with the Computer and Systems Security Lab (COMPASS). His research interests include system security, with a focus on confidential computing, trustworthy execution, and hardware-assisted security.

Fengwei Zhang (Senior Member, IEEE) received the Ph.D. degree in computer science from George Mason University. He is currently an Associate Professor with the Department of Computer Science and Engineering, Southern University of Science and Technology (SUSTech). His research interests include systems security, with a focus on trustworthy execution, hardware-assisted security, debugging transparency, transportation security, and plausible deniability encryption.