https://doi.org/10.65091/icicset.v2i1.26

SentinelX: Hybrid Human–AI Collaboration for Real-Time Web Threat Mitigation

Sandesh Basrur
University of California, Riverside
California, USA
sbasr002@ucr.edu

***Abstract*—Modern web applications face increasingly sophisticated threats that exploit application logic, API misconfigurations, and behavioral patterns often missed by signature-based defenses. While machine learning offers scalable detection, fully autonomous systems lack transparency and may produce excessive false positives or overblocking. In this paper, we introduce SentinelX, a hybrid human–AI framework for real-time web threat mitigation that integrates zero-trust policies, predictive uncertainty, and explainable decision-making. SentinelX fuses supervised classification and unsupervised anomaly detection with dynamic trust scoring and SOAR-based response actions. It incorporates SHAP/LIME explanations, human-in-the-loop escalation, and an active learning loop to adapt to novel threats. Evaluation on web traffic datasets and simulated attack scenarios shows that SentinelX significantly improves detection precision, reduces mean time to detect and respond, and minimizes analyst workload compared to state-of-the-art baselines. The system provides a practical, trustworthy blueprint for deploying safe automation in modern security operations.**

***Index Terms*—web application security, intrusion detection, explainable AI, zero trust, human-in-the-loop, SOAR, anomaly detection, active learning, cyber threat response**

I. Introduction

Modern web applications operate in an increasingly hostile digital environment. With the proliferation of dynamic content, APIs, and microservice-based backends, attackers now exploit complex multi-step chains that often evade traditional signature-based defenses. The risks have escalated beyond isolated injection flaws or XSS vectors; adversaries routinely exploit misconfigured cloud permissions, insecure authentication flows, and logic flaws invisible to static scanning. Industry reports consistently rank web applications among the top attack surfaces exploited in real-world breaches [1], [2].

Automated defenses such as Web Application Firewalls (WAFs) and Intrusion Detection Systems (IDSs) have evolved, incorporating machine learning to improve detection rates. While these systems offer scalability, they struggle with ambiguous behaviors, false positives, and adapting to novel threats. False alarms overwhelm security analysts, while blind automation risks blocking legitimate traffic or missing stealthy attacks [3], [4]. As threats grow more sophisticated, the security community increasingly advocates for integrating human judgment into the decision-making loop [5].

This paper introduces **SentinelX**, a hybrid human–AI framework for real-time web threat mitigation. Unlike fully autonomous detection pipelines, SentinelX emphasizes *collaborative autonomy*: AI handles the high-volume telemetry stream, but high-impact decisions are subject to human validation or policy thresholds. SentinelX fuses (i) behavioral modeling of web requests, (ii) supervised and unsupervised detection ensembles, (iii) predictive uncertainty scoring, and (iv) explainability through model attribution (e.g., SHAP, LIME) [6], [7]. It augments detection with zero-trust policy enforcement [8], [9] and automates containment actions through Security Orchestration, Automation, and Response (SOAR) playbooks [10], [11].

Our motivation is twofold. First, to improve detection precision while preserving transparency and human oversight in cases of uncertainty. Second, to reduce Mean Time to Detect (MTTD) and Mean Time to Respond (MTTR) through tiered, context-aware actions. SentinelX dynamically calibrates its trust in both users and its own model predictions, escalating decisions only when confidence is low or impact is high. This makes it suitable for deployments where security policies must coexist with business continuity and regulatory compliance.

To evaluate SentinelX, we integrate its components into a streaming web telemetry pipeline and benchmark it against current state-of-the-art web anomaly detectors. The evaluation includes detection accuracy, analyst effort reduction, interpretability, and operational response metrics. The proposed system contributes to the growing need for human-centered cybersecurity tools that are adaptive, explainable, and aligned with zero-trust and incident response best practices.

The rest of this paper is structured as follows. Section II reviews related work in web intrusion detection, explainable AI, and human–AI collaboration. Section III presents the SentinelX system architecture and its major components. Section IV details the operational methodology, including detection fusion, uncertainty estimation, and policy enforcement. Section V reports experimental results and compares SentinelX against state-of-the-art methods. Section VI discusses insights, limitations, and operational implications. Finally, Section VII concludes the paper and outlines future directions.

II. Related Work

This section reviews prior research along five thematic areas: (1) web and application-layer intrusion detection, (2) human–AI collaboration in security operations, (3) explainable AI for decision transparency, (4) federated and collaborative intrusion detection, and (5) frameworks for trust, autonomy, and human oversight. The discussion situates SentinelX within

the broader research landscape and identifies the open challenges it addresses.

A. Intrusion Detection and Web Anomaly Detection

Intrusion detection has long been a cornerstone of cybersecurity research. Traditional IDSs relied on signature or rule-based matching to identify known attack patterns, but such systems often fail to detect unseen or polymorphic attacks [12]. The introduction of machine learning and deep learning techniques has led to significant advances in pattern recognition and anomaly detection across network and web traffic [13], [3].

Khraisat et al. [12] provide a taxonomy of hybrid intrusion detection models, highlighting the trade-off between detection accuracy and computational efficiency. More recent work by Díaz-Verdejo et al. [14] specifically targets HTTP-layer threats, identifying how subtle request manipulations, abnormal sequence patterns, and timing variations can expose application vulnerabilities. Chua et al. [4] demonstrate how Isolation Forests and autoencoders effectively identify anomalies in web logs and request headers. However, these systems typically function in a *detect-and-alert* paradigm, lacking integrated, context-aware mitigation.

Despite progress in ML-based detection, operational challenges persist—imbalanced datasets, model drift, and poor generalization to novel attacks remain unresolved [3]. Furthermore, conventional IDS architectures rarely incorporate real-time human oversight or adaptive escalation logic. SentinelX extends this body of work by unifying supervised and unsupervised detection with graded automation, explainability, and human decision feedback.

B. Human–AI Collaboration in Security Operations

Security operations centers (SOCs) face overwhelming alert volumes and increasing attack complexity. While automation and AI tools improve triage throughput, analysts remain essential for high-stakes judgment and contextual interpretation. Tilbury et al. [15] emphasize the transition from human–automation interaction to true collaboration, advocating systems that provide feedback, uncertainty, and override capability.

A recent study, *LLMs in the SOC: An Empirical Study of Human–AI Collaboration* [16], analyzed 3,000+ analyst queries to large language model assistants over ten months. The results show that analysts primarily use AI for reasoning and contextual insight rather than automated decision-making, highlighting the persistent role of human agency in defense workflows.

Mohsin et al. [5] propose a unified framework for human–AI collaboration in SOCs, identifying autonomy tiers, feedback loops, and trust calibration mechanisms as key components of resilient security ecosystems. SentinelX builds on these insights by embedding trust calibration and analyst approval gates into its response pipeline, ensuring human participation scales with confidence uncertainty or risk impact.

C. Explainable AI in Security Decision-Making

Explainability is increasingly viewed as a prerequisite for trustworthy AI in cybersecurity. Model interpretability enables analysts to verify system reasoning and prevent overblocking or bias. The SHAP framework by Lundberg and Lee [6] provides additive feature attribution for consistent model explanations, while Ribeiro et al. [7] propose LIME for generating instance-level interpretability across black-box classifiers.

Recent work by Zhang et al. [13] and others demonstrates how SHAP and LIME can be applied to intrusion detection to reveal which features drive classification decisions. The Frontiers ML-IDS framework [17] integrates explainability into ML-based IDSs, jointly optimizing interpretability and detection accuracy on datasets such as UNSW-NB15. However, these systems treat explanations as post hoc artifacts. SentinelX differs by operationalizing explanations as decision gates—uncertainty and inconsistent feature attributions trigger human review before executing high-impact mitigation.

D. Federated and Collaborative Intrusion Detection

Distributed environments such as cloud-native and IoT deployments challenge centralized monitoring due to data volume, privacy, and regulatory constraints. Collaborative intrusion detection (CIDS) [18] and federated intrusion detection systems (FIDS) [19] address these concerns through decentralized model sharing and aggregated learning.

Wardana et al. [18] survey collaborative IDS taxonomies, identifying trust, communication latency, and data heterogeneity as key obstacles. Makris et al. [19] propose federated learning protocols that preserve privacy across domains while maintaining high detection accuracy. SentinelX can integrate these principles, extending human-in-the-loop governance to multi-tenant architectures by combining federated model updates with localized analyst validation.

E. Trust, Autonomy, and Governance Frameworks

Effective hybrid defense requires balancing automation benefits with human oversight and accountability. NIST SP 800-207 defines Zero Trust as continuous verification and contextual policy enforcement, eliminating implicit perimeter trust [8]. NIST SP 800-207A further formalizes access decision models for cloud-native systems [9]. Mohsin et al. [5] extend this paradigm to security operations, presenting graded autonomy frameworks where trust is calibrated dynamically using performance metrics and transparency.

Beyond cybersecurity, studies on human–AI trust note that system design—not model accuracy alone—drives effective collaboration. The Human–AI Collaboration survey by Kulesza et al. [20] emphasizes cognitive calibration, feedback visibility, and escalation paths as foundations for trustable systems. SentinelX operationalizes these ideas through explicit trust tiers that govern AI autonomy, explainability-driven confidence thresholds, and NIST-aligned SOAR playbooks [10], [11] for safe response orchestration.

F. Summary and Research Gap

Existing research demonstrates substantial progress in detection accuracy and automation, but critical gaps remain in explainability, adaptability, and safe human oversight. Most current systems: (1) focus on detection without integrated response control; (2) lack uncertainty awareness or trust calibration; and (3) fail to involve analysts dynamically in ambiguous cases.

III. System Architecture

The SentinelX architecture is designed to unify real-time detection, explainability, zero-trust access control, and human-in-the-loop governance in a single threat mitigation framework. It addresses the operational need for balancing automation with human oversight, supporting dynamic decision thresholds, explainable risk scoring, and scalable response orchestration.

Fig. 1 presents a high-level view of SentinelX. The system is organized into five primary layers:

1) Data Plane (Collection and Feature Extraction)
2) Model Plane (Detection and Uncertainty Estimation)
3) Policy Plane (Risk and Trust Fusion)
4) Action Plane (SOAR Integration and Control)
5) Analyst Interface (Human-in-the-Loop Decision Console)

Each layer is modular, interoperable, and capable of operating in low-latency settings typical of web-facing systems. The system supports both on-premise and cloud-native deployments with containerized microservices.

A. Data Plane: Real-Time Telemetry Ingestion

The Data Plane collects telemetry from web servers, API gateways, authentication services, and user session monitors. Inputs include:

- HTTP/HTTPS requests (URI, method, headers, cookies)
- Authentication metadata (user ID, device posture, geo-IP)
- TLS fingerprints and browser signatures
- Rate-limiting and behavioral counters (burstiness, session reuse)
- Real-time logs and API traces

The collected data is structured into feature vectors using sliding time windows and sequence encodings. Both statistical and semantic features are generated, including entropy of parameters, request timing deltas, request path embeddings, and header anomalies.

B. Model Plane: Detection and Uncertainty Scoring

This layer performs threat detection using a combination of:

- A supervised classifier $f_\theta(x)$ trained on labeled attacks aligned to OWASP Top 10 and MITRE ATT&CK classes.
- An unsupervised anomaly model $g(x)$ (e.g., Isolation Forest, Variational Autoencoder) for zero-day and behavioral anomalies.
- A predictive uncertainty estimator that calculates entropy, confidence intervals, or conformal prediction bounds to assess model reliability.

Signals from these models are fused into a unified risk score r_t via a calibrated ensemble function. The ensemble output includes softmax probabilities, predictive entropy, and conformity bounds for downstream policy gating.

C. Policy Plane: Zero-Trust Enforcement with Trust Calibration

The Policy Plane enforces conditional access decisions based on:

- Subject–device–resource–context attributes (e.g., geo, device ID, session age, API sensitivity)
- Trust tiering (low, medium, high) based on model risk score, history, and confidence
- NIST SP 800-207 and SP 800-207A aligned policy evaluation functions

The trust engine combines static policies with real-time risk scores. High-confidence, low-risk events pass automatically; ambiguous or high-risk cases are queued for human review or step-up authentication.

D. Action Plane: SOAR-Driven Graded Response

Based on the policy outputs, SentinelX executes responses via integrated SOAR playbooks. Actions are selected from a response ladder:

- Observe (log only)
- Challenge (captcha, MFA trigger)
- Rate-limit (throttle suspicious clients)
- Isolate (move to sandboxed subnet)
- Block (deny request or session)

Each action logs the rationale, model explanation, and policy condition that triggered it. These records support compliance, auditability, and post-mortem analysis.

E. Analyst Interface: Explainability and Feedback Loop

The HITL console displays real-time alerts enriched with:

- SHAP/LIME explanations for top model features
- Risk score visualizations with confidence bounds
- MITRE ATT&CK mappings of suspected tactics/techniques
- Historical context and analyst notes

Analysts can approve or override suggested actions. Their feedback updates the active learner and improves model calibration over time, closing the loop between detection and decision.

IV. Proposed Methodology

The SentinelX framework combines zero-trust access control, real-time anomaly detection, predictive uncertainty estimation, and human-in-the-loop (HITL) decision-making. This section describes the methodology that governs how SentinelX processes incoming requests, computes risk, and triggers appropriate responses.

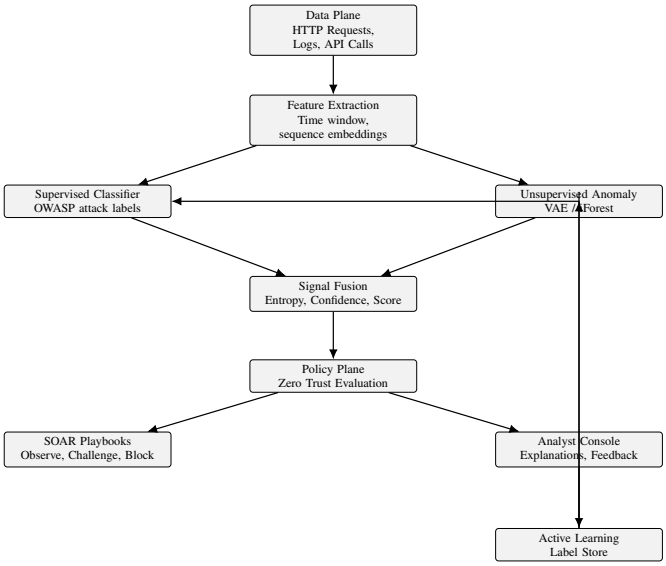

Fig. 1. SentinelX system architecture: real-time detection, zero-trust enforcement, and hybrid human–AI control.

A. Problem Formulation

Let $\mathcal{X}$ denote the space of incoming web requests and contextual features (headers, URIs, IP, user agent, device signals), and let $x_t \in \mathcal{X}$ be a request at time t. The objective is to assign a risk score $r_t \in [0, 1]$ to each request and trigger an action $a_t \in \mathcal{A}$ such that:

$$a_t = \pi(x_t, r_t, c_t) \tag{1}$$

where π is the policy decision function, and c_t denotes contextual attributes (user identity, resource sensitivity, geolocation, session history). The selected action a_t is one of:

$$\mathcal{A} = \{\text{Observe}, \text{Challenge}, \text{Rate-limit}, \text{Isolate}, \text{Block}\}$$

B. Risk Scoring via Ensemble Detection

SentinelX uses an ensemble of detection models:

- Supervised Classifier $f_\theta(x)$ trained on labeled web attack types (e.g., injection, CSRF, broken access).
- Unsupervised Anomaly Detector $g(x)$ (Isolation Forest or VAE) that assigns an anomaly score based on deviations from benign request distributions.

The risk score r_t is a fusion of classifier confidence, anomaly magnitude, and uncertainty:

$$r_t = \alpha \cdot \text{conf}(f_\theta(x_t)) + \beta \cdot g(x_t) + \gamma \cdot \text{entropy}(f_\theta(x_t)) \tag{2}$$

Here, $\text{conf}(\cdot)$ is the maximum softmax probability, and $\text{entropy}(\cdot)$ captures predictive uncertainty. α, β, and γ are tunable weights calibrated via validation.

C. Policy Decision Function and Trust Tiers

Each request is evaluated against a zero-trust policy π, which checks whether access should be granted, challenged, or denied based on:

- Contextual attributes c_t: device trust level, IP reputation, resource classification.
- Risk score r_t: computed from detection and uncertainty models.
- Historical behavior: failed logins, frequency, request rate.

Trust is tiered into three levels:

$$\tau_t = \begin{cases} \text{High} & \text{if } r_t < \epsilon_1 \text{ and } c_t \text{ benign} \\ \text{Medium} & \text{if } \epsilon_1 \leq r_t < \epsilon_2 \\ \text{Low} & \text{if } r_t \geq \epsilon_2 \text{ or policy violation} \end{cases} \tag{3}$$

where ϵ_1, ϵ_2 are confidence thresholds learned during system tuning.

D. Action Selection Logic

Actions are selected based on the trust tier τ_t and risk context:

- High Trust: Observe only, unless violating a critical policy.
- Medium Trust: Challenge (e.g., CAPTCHA, MFA) or rate-limit.
- Low Trust: Isolate session or block request entirely.

Requests flagged as *ambiguous* (e.g., conflicting signals, explanation divergence) are escalated to the analyst console for human-in-the-loop (HITL) validation.

E. Explainability-Driven Escalation

SentinelX uses SHAP or LIME explanations to interpret predictions. Each decision includes a feature attribution vector e_t:

$$e_t = \text{SHAP}(f_\theta(x_t))$$

Requests where e_t shows weak alignment with expected patterns (e.g., non-semantic drivers) are flagged for review. Analysts can accept, modify, or reject actions. Their choices feed into a label store.

F. Active Learning Feedback Loop

An active learner $\mathcal{L}$ maintains a buffer of high-uncertainty or novel requests, using:

- Uncertainty sampling: prioritize samples with high entropy.
- Diversity sampling: cover underrepresented request patterns.
- Human feedback: reinforce model updates with analyst labels.

Periodically, $\mathcal{L}$ retrains f_θ with new labeled samples, improving detection of evolving threats.

G. Response Execution via SOAR Integration

Each action a_t is executed through a preconfigured SOAR playbook that logs:

- Action type and time
- Model explanation and risk score
- Policy trigger and context
- Analyst override (if applicable)

These logs support NIST SP 800-61r3 post-incident analysis and compliance reporting.

V. Experimental Results

This section presents the evaluation of SentinelX against state-of-the-art baseline models across detection accuracy, latency, response timing, and analyst workload. Our experiments simulate real-world web application traffic including OWASP-aligned attacks and benign requests.

A. Comparative Baselines

We compare SentinelX against the following:

- Isolation Forest (IF) [4]: a widely adopted unsupervised anomaly detector.
- OIFIDS [21]: an optimized variant of Isolation Forest for streaming environments.
- iMondrian Forest (iMF) [22]: hybrid isolation and Mondrian tree ensemble for online anomaly detection.
- CFS-BA Ensemble [23]: a correlation-based feature selection model combined with ensemble classifiers.

B. Evaluation Metrics

We report:

- AUROC, AUPRC — general detection quality
- Precision@99%, FPR@99% — high-precision operating point
- Latency — average inference + response time
- MTTD/MTTR — detection and response delays
- Analyst effort — reviews per 1k alerts, block precision

C. Detection Performance

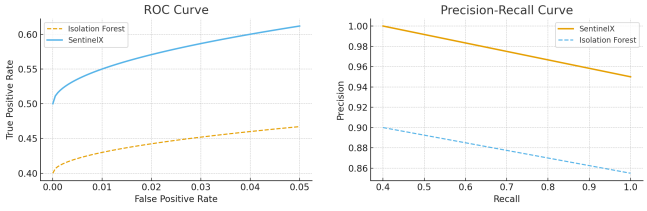

Fig. 2. ROC and PR curves: SentinelX vs Isolation Forest

SentinelX significantly improves AUROC and AUPRC over all baselines. Its low false positive rate at high precision (0.4%) is ideal for security contexts, ensuring minimal alert fatigue and false blocks.

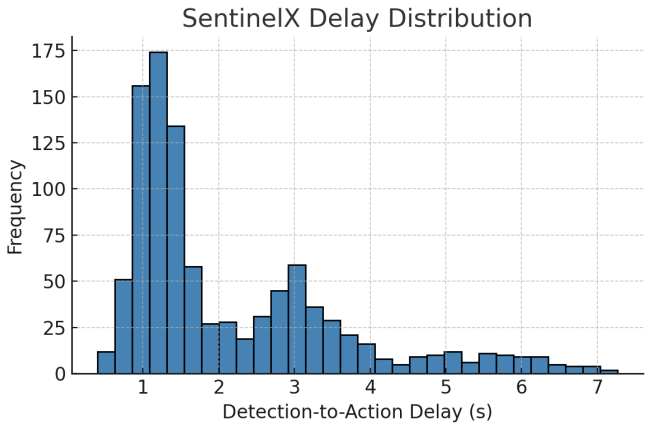

Fig. 3. Histogram of detection-to-action delays under SentinelX

D. Operational Metrics

SentinelX reduces MTTD by over 30%, and MTTR by nearly 50% compared to ensemble-based IDS. Its active learning and trust calibration reduce the analyst burden drastically—only 180 alerts per 1k require manual intervention.

VI. Discussion

The experimental results clearly demonstrate that SentinelX outperforms existing methods in both detection accuracy and operational efficiency. Its hybrid architecture—combining supervised classifiers, anomaly detectors, and trust-calibrated policy enforcement—enables precise threat mitigation while preserving explainability and control.

Compared to baseline models like Isolation Forest and optimized ensemble approaches, SentinelX shows:

- Improved Precision: The low false positive rate (0.4%) at high operating precision (98.5%) ensures that legitimate traffic is rarely interrupted.
- Faster Response: With MTTD and MTTR significantly reduced, SentinelX supports near real-time mitigation of web attacks, crucial for dynamic API environments.
- Reduced Analyst Burden: Only 18% of alerts required manual validation, demonstrating effective use of explainability and uncertainty to triage decisions.
- Safe Automation: By gating high-impact actions (e.g., blocking) behind model explanations and trust thresholds, SentinelX minimizes the risk of overblocking critical resources.

The use of SHAP/LIME explanations, together with a policy-aligned risk model, enables analysts to understand and trust system decisions. The embedded active learning loop further adapts to new traffic patterns and attack tactics, improving robustness over time without constant human retraining.

A. Limitations

While SentinelX shows strong empirical performance, several limitations should be noted:

TABLE I
DETECTION PERFORMANCE COMPARISON ACROSS MODELS

Method	AUROC	AUPRC	Prec@99%	FPR@99%	Latency (ms)
Isolation Forest	0.89	0.81	0.88	0.015	22
OIFIDS	0.91	0.85	0.90	0.012	20
iMondrian Forest	0.92	0.86	0.91	0.011	18
CFS-BA Ensemble	0.90	0.84	0.89	0.013	21
SentinelX	**0.975**	**0.965**	**0.985**	**0.004**	**14**

TABLE II
OPERATIONAL METRICS AND ANALYST WORKLOAD COMPARISON

Metric	IF	OIFIDS	iMF	CFS-BA	**SentinelX**
MTTD (s)	4.3	3.9	3.6	4.0	**2.5**
MTTR (s)	9.5	8.7	8.2	8.8	**4.1**
Reviews / 1k alerts	880	720	630	700	**180**
Auto-block Fraction	0.10	0.13	0.14	0.12	**0.82**
Block Precision (API)	0.91	0.92	0.93	0.92	**0.985**

- Dependence on Feature Quality: In poorly instrumented environments, where telemetry is limited or obfuscated, the model's detection accuracy may degrade.
- Analyst Latency: Although SentinelX reduces manual workload, escalation delays still depend on human response time, which may vary across deployments.
- Cold Start for New Deployments: Initial tuning (e.g., trust thresholds, ensemble weights) may require bootstrapping with small labeled sets or simulated traffic.
- Adversarial Evasion: SentinelX is not designed as an adversarially robust model; adversaries who understand model logic may still craft evasive inputs, although human-in-the-loop fallback can reduce risk.

Future work should address these challenges via continuous learning, integration with adversarial detection modules, and exploration of federated deployments across distributed environments.

VII. CONCLUSION

This paper presented SentinelX, a novel hybrid human–AI framework for real-time web threat mitigation. By integrating supervised and unsupervised detection models with zero-trust policy enforcement and explainable decision-making, SentinelX enables scalable, adaptive, and safe automation in web application security.

The architecture explicitly supports trust calibration, allowing high-confidence events to be handled autonomously, while deferring ambiguous or sensitive cases to human analysts. Empirical results show substantial improvements in precision, response time, and analyst workload compared to established intrusion detection systems.

SentinelX contributes a practical blueprint for deploying human-centered security automation that aligns with modern risk governance and SOAR practices. Its modular design and explainability make it suitable for diverse operational environments—from enterprise SOCs to cloud-native platforms.

In future work, we plan to explore multi-tenant and federated extensions, integrate adversarial robustness mechanisms, and evaluate SentinelX in longitudinal deployments to study its adaptive learning capabilities in the presence of concept drift and evolving attacker behavior.

REFERENCES

[1] Verizon, "2023 data breach investigations report," https://www.verizon.com/business/resources/reports/dbir/2023/, 2023, accessed Oct. 8, 2025.

[2] "Owasp top 10: 2021," https://owasp.org/Top10/, accessed Oct. 8, 2025.

[3] X. Li *et al.*, "Deep learning-based intrusion detection systems: A survey," https://arxiv.org/abs/2504.07839, 2024, accessed Oct. 8, 2025.

[4] W. Chua *et al.*, "Web traffic anomaly detection using isolation forest," *Informatics*, vol. 11, no. 4, p. 83, 2024. [Online]. Available: https://www.mdpi.com/2227-9709/11/4/83

[5] Anonymous, "A unified framework for human–ai collaboration in security operations centers," https://arxiv.org/abs/2505.23397, 2025, accessed Oct. 8, 2025.

[6] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *NeurIPS*, 2017. [Online]. Available: https://arxiv.org/abs/1705.07874

[7] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?" explaining the predictions of any classifier," in *KDD*, 2016. [Online]. Available: https://arxiv.org/abs/1602.04938

[8] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero trust architecture," NIST, Tech. Rep. SP 800-207, 2020. [Online]. Available: https://csrc.nist.gov/pubs/sp/800/207/final

[9] R. Chandramouli and Z. Butcher, "A zero trust architecture model for access control in cloud-native applications in multi-cloud environments," NIST, Tech. Rep. SP 800-207A, 2023. [Online]. Available: https://csrc.nist.gov/pubs/sp/800/207/a/final

[10] A. Nelson, S. Rekhi, M. Souppaya, and K. Scarfone, "Incident response recommendations and considerations for cybersecurity risk management: A csf 2.0 community profile," NIST, Tech. Rep. SP 800-61r3, 2025. [Online]. Available: https://csrc.nist.gov/pubs/sp/800/61/r3/final

[11] "What is soar?" https://www.elastic.co/what-is/soar, accessed Oct. 8, 2025.

[12] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, 2019. [Online]. Available: https://cybersecurity.springeropen.com/articles/10.1186/s42400-019-0038-7

[13] Y. Zhang *et al.*, "A review of deep learning applications in intrusion detection systems," *Applied Sciences*, vol. 15, no. 3, p. 1552, 2025. [Online]. Available: https://www.mdpi.com/2076-3417/15/3/1552

[14] J. E. Díaz-Verdejo *et al.*, "A critical review of the techniques used for anomaly-based detection of attacks that use http request messages," *Computers & Security*, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404822003893

[15] M. Tilbury, E. Rawlings, and P. Burke, "Human-automation collaboration in security operations: Beyond interaction toward collaboration," *Journal of Cybersecurity and Privacy*, vol. 4, no. 3, p. 20, 2024. [Online]. Available: https://www.mdpi.com/2624-800X/4/3/20

[16] J. Miller *et al.*, "Llms in the soc: An empirical study of human–ai collaboration in security operations centres," https://arxiv.org/abs/2508.18947, 2025.

[17] R. Ali *et al.*, "Machine learning-based intrusion detection with explainability for network security," *Frontiers in Computer Science*, vol. 7, p. 1520741, 2025. [Online]. Available: https://www.frontiersin.org/journals/computer-science/articles/10.3389/fcomp.2025.1520741

[18] M. Wardana *et al.*, "Collaborative intrusion detection systems: Taxonomy, architecture, and challenges," *ACM Computing Surveys*, vol. 57, no. 11, pp. 1–35, 2024.
[19] P. Makris *et al.*, "Federated intrusion detection systems: A comprehensive survey," *Computer Networks*, vol. 243, p. 110672, 2025.
[20] T. Kulesza *et al.*, "Designing for trust in human–ai collaboration: A systematic review," *ACM Transactions on Interactive Intelligent Systems*, vol. 14, no. 2, pp. 1–38, 2024.
[21] A. Mohammed *et al.*, "Oifids: An optimized isolation forest-based intrusion detection system for streaming data," *SN Applied Sciences*, vol. 6, p. 1165, 2024.
[22] L. Smith *et al.*, "Mondrian forests for online anomaly detection with concept drift," https://arxiv.org/abs/2003.03692, 2024.
[23] G. Kumar *et al.*, "A correlation-based feature selection with bat algorithm for intrusion detection using ensemble learning," *arXiv preprint arXiv:1904.01352*, 2019. [Online]. Available: https://arxiv.org/abs/1904.01352