Twin SOC: Counterfactual Digital-Twin Verification for AI-Driven Incident Response

Ramya S[1], Sinthuja V[2], Hemalatha G[3]

[1]Assistant Professor, [2]Assistant Professor, [3]Assistant Professor

Department of Cyber Security

Paavai Engineering College (Autonomous), Namakkal, Tamil Nadu, India.

Abstract:

Artificial intelligence increasingly drives detection and response in modern security operations centers, yet the industry lacks a rigorous mechanism to validate AI-recommended actions before they touch production systems. The proposed system introduces TwinSOC, a counterfactual verification framework in which machine-learning detectors and an LLM-based remediation planner operate in tandem with a high-fidelity digital twin of the enterprise environment to test proposed responses before execution. TwinSOC binds every AI decision to a reproducible counterfactual experiment inside the twin, uses temporal-logic safety contracts to ensure that isolation, identity, and network policy changes respect organizational constraints, and emits cryptographic attestations that record the exact models, data summaries, and policy versions that produced each pass-or-fail verdict. Actions only proceed to the orchestration layer when the counterfactual proves that service-level objectives and access boundaries will hold with high confidence, while failed proposals return actionable explanations to human analysts. We present the system architecture, modeling and validation methods, and an evaluation plan that measures detection lift, response correctness, analyst workload, and safety violations prevented on both replayed attack traces and live pilot deployments. TwinSOC advances cybersecurity and AI by converting opaque recommendations into verifiable, policy-safe interventions that can be audited end-to-end.

Keywords: cybersecurity; artificial intelligence; incident response; digital twin; counterfactual simulation; explainability; trusted execution; provenance; SOAR.

I. Introduction

Security teams have rapidly adopted machine-learning detectors and large language models to triage alerts, summarize incidents, and recommend containment steps, but they remain reluctant to grant these systems broad autonomy because the consequences of an erroneous action can be severe. Blocking a legitimate service, revoking the wrong identity role, or isolating a critical network segment without advance verification can precipitate outages that eclipse the harm of the original threat. TwinSOC addresses this trust gap by requiring every AI-generated response to be validated as a counterfactual in a digital twin that mirrors identity relationships, network topology, and policy semantics. Rather than executing a remediation directly, the system first simulates the action in the twin, checks temporal-logic safety contracts that express organizational policies, and only then decides whether to promote the action to the production orchestrator or return it to an analyst with an explanation. This architecture preserves the speed benefits of AI while providing a principled safety layer that translates recommendations into verifiable, auditable interventions.

II. Background and Related Work

Contemporary security operations combine rule-based correlation, anomaly detection, supervised classifiers, and increasingly LLM-assisted triage to manage high alert volumes. Digital-twin techniques have matured in cyber-physical domains but are under-explored for enterprise security, where identity graphs, routing policies, and access controls change continuously. Automated response platforms can execute playbooks at scale, yet their safeguards typically rely on static approvals or coarse pre-checks rather than formal contracts tied to the exact proposed action. Research on explainable AI has improved analyst understanding of model outputs, but explanations alone do not guarantee that a recommended change will satisfy service-level objectives or respect least-privilege principles at runtime. TwinSOC integrates these strands by coupling detectors and LLM planners with a live twin that can evaluate counterfactuals under the same policy semantics as production, thereby creating a closed loop in which recommendations are tested, constrained, and attested before they reach the environment they are meant to protect.

III. Problem Statement and Objectives

The central problem is to design a security automation system that can transform AI recommendations into safe actions with guarantees that matter to operators. The system must construct counterfactual experiments that are faithful to production state, express organizational policies as machine-checkable temporal contracts, and quantify uncertainty so decisions remain conservative under partial knowledge. It must also generate evidence that links data, models, policies, and outcomes so that every decision can be audited long after the incident has closed. The objectives of TwinSOC are to raise detection quality through ensemble learning, to convert remediation plans into constrained action sets that satisfy policy contracts in the twin, to promote only those actions whose simulated effects keep availability and access within bounds, and to reduce analyst workload by returning explanations that pinpoint the failing preconditions when a recommendation is rejected.

IV. System Overview and Design

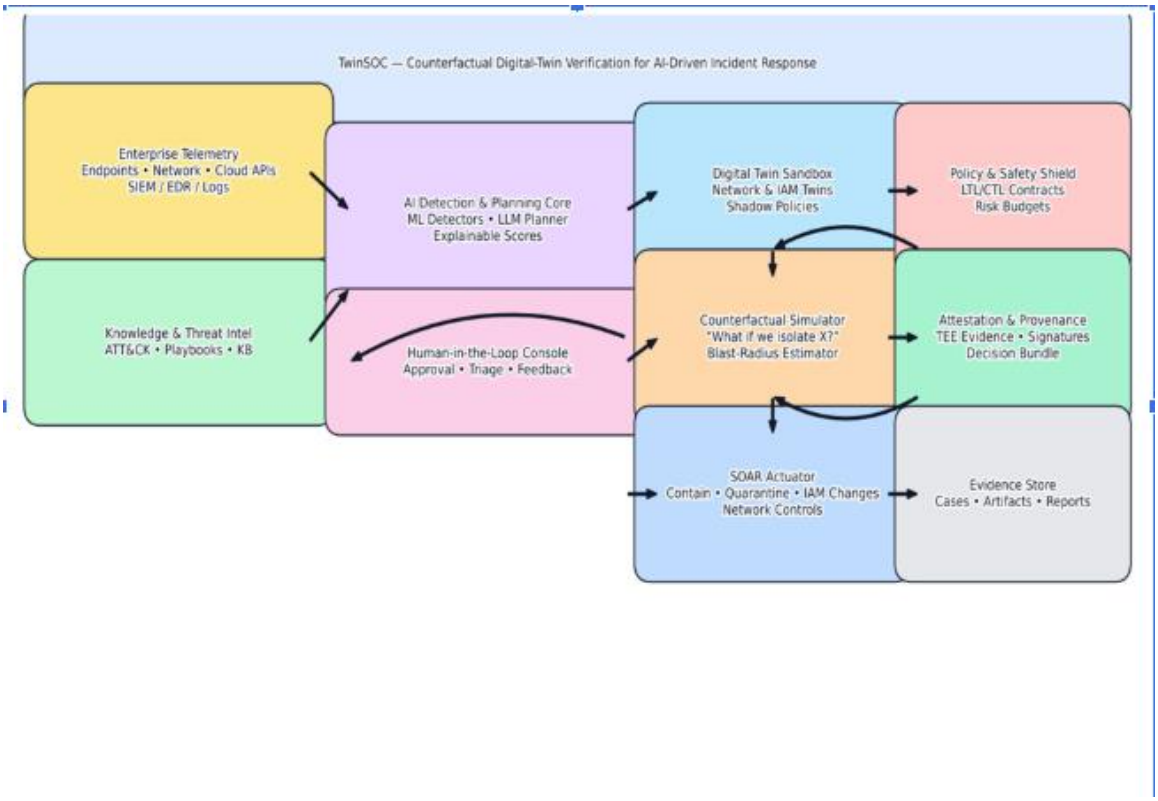

Figure 1: TwinSOC architecture showing the flow from telemetry and intelligence into AI detection and planning, counterfactual verification inside a digital twin, policy-safe decisioning with attestations, and controlled execution through the orchestration layer.

TwinSOC is organized around five cooperating subsystems integrated with existing telemetry and orchestration stacks. An ingestion layer consolidates endpoint, network, and cloud control-plane signals and refreshes a digital twin that encodes identity relationships, routing, and access policies at a level of abstraction appropriate for counterfactual simulation. A detection and planning core combines supervised models with an LLM that synthesizes candidate response plans from alerts and playbooks while emitting explanations that reference the evidence chain. A counterfactual engine executes proposed actions inside the twin, computes blast-radius metrics, and evaluates temporal-logic safety contracts that capture invariants such as mandatory reachability, least-privilege constraints, and segregation of duties. A policy and safety shield consumes these results and issues a pass or fail verdict with calibrated confidence, and a provenance component running in trusted execution environments signs a decision bundle that binds model hashes, policy versions, data summaries, and the final verdict to a case identifier. Only when the verdict passes with sufficient confidence is an action forwarded to the orchestration layer for execution in production; otherwise, the system returns a structured explanation to the analyst console that describes which contract failed and how the plan can be adjusted.

V. Modeling, Contracts, and Verification Methods

Reliable counterfactuals require models and contracts that reflect operational realities. The digital twin captures service dependencies, access control graphs, and network segments as typed relations so that actions such as isolating a host, revoking a role, or tightening an egress rule can be simulated with fidelity. Temporal-logic contracts describe invariants over these relations, including that specific health checks must remain reachable, that privileged identities cannot obtain broader access as a side effect of a change, and that cross-segment isolation preserves required east-west paths for critical

workflows. The counterfactual engine executes proposed actions against snapshots of twin state, evaluates contracts with a model checker, and computes risk metrics such as predicted service impact and residual exposure. Uncertainty arising from incomplete inventory or delayed telemetry is handled through conservative bounds and targeted queries that request fresh evidence before a verdict is issued, thereby keeping the system safe under realistic data conditions.

VI. Integration with Security Orchestration and Human Oversight

TwinSOC integrates with existing security information and event management, endpoint detection and response, and orchestration tools so that adoption does not require a wholesale replacement of the toolchain. Promoted actions are executed by the orchestrator using standard connectors, while the analyst console provides an approval workflow that can require human sign-off for high-risk changes or during early stages of deployment. Feedback from analysts is looped back to the planning models and playbooks so that the quality of recommendations improves over time, and the system maintains a case-centric evidence store that links alerts, simulations, contracts, decisions, and outcomes for later investigation and compliance reporting.

VII. Experimental Design and Evaluation

The evaluation of TwinSOC should demonstrate improvements in detection quality, response correctness, analyst efficiency, and safety. A trace-replay study can benchmark the system on representative attack scenarios by mirroring traffic and control-plane events into the twin, issuing model-generated plans, and observing pass or fail outcomes under the safety shield before comparing them with ground-truth impact measurements in a staging environment. Live pilot deployments can measure the reduction in erroneous actions, the number of policy violations prevented by the shield, changes in mean time to respond, and analyst workload as reflected in case handling time and cognitive load surveys. Statistical analysis should report confidence intervals for each outcome and include ablations that remove the counterfactual engine or relax contracts to quantify the specific contributions of verification to overall performance.

VIII. Results and Discussion

A well-implemented TwinSOC deployment is expected to show that AI recommendations can be transformed into safe, auditable actions without sacrificing response speed. Counterfactual verification should prevent classes of outages caused by overly aggressive isolation or misapplied role changes, while explanations and evidence bundles should improve analyst trust and facilitate post-incident reviews. The discussion should interpret observed improvements in terms that matter to operations, such as avoided minutes of service unavailability, reduction in improper privilege grants, and decrease in escalations, and it should examine the trade-offs between stricter contracts and the throughput of automated actions.

IX. Threats to Validity and Limitations

Digital twins inevitably approximate reality and may miss configuration drift or ephemeral dependencies, which can lead to false assurances; mitigation requires continuous synchronization and conservative contracts that favor safety when doubt exists. The quality of AI recommendations depends on training data and prompt design, and model drift can degrade performance over time; regular evaluation and fallback to human-only workflows preserve resilience. Formal contracts may be incomplete or too strict at first, so organizations must iterate on their specification and monitor for unintended blocking of legitimate responses. Finally, the attestation mechanism must protect sensitive information while remaining verifiable by authorized stakeholders, which motivates the use of trusted execution environments and careful redaction policies.

X. Ethical, Legal, and Pedagogical Considerations

Automated defenses influence user access and service availability, so the system must operate under clear governance with accountable human oversight and transparent evidence for every decision. TwinSOC provides such transparency by attaching explanations and signed provenance to each action, enabling audits and appeals where necessary. From an educational standpoint, the framework unifies core Computer Science and Engineering themes such as modeling, formal specification, machine learning, systems design, and security compliance into a coherent, hands-on project that can be reproduced in academic laboratories without exposing real production systems.

XI. Conclusion

TwinSOC reframes AI-driven security automation as a verifiable process in which recommendations are validated as counterfactuals inside a faithful digital twin, constrained by explicit safety contracts, and promoted to production only when evidence supports a positive verdict. By combining detection, planning, formal verification, and provenance in a single pipeline, the framework delivers practical assurances that help security teams adopt AI responsibly while maintaining availability and compliance. The architecture and methods described here can be implemented incrementally on top of existing tools, creating a path from advisory AI to trustworthy, policy-safe autonomy in incident response.

References

1) M. Naeem, “Contract-based Verification of Digital Twins,” arXiv preprint arXiv:2506.10993, Jun. 2025.
2) D. Allison, “Digital Twin-Enhanced Incident Response for Cyber-Physical Systems,” in Proc. ACM Conf. (ACM Digital Library), 2024/2025.
3) K. E. Kampourakis, “Digital Twin-Enabled Incident Detection and Response,” Journal / Springer Article, 2025.
4) A. Perišić, “Digital Twins Verification and Validation Approach through Systems-Engineering Processes,” Electronics, vol. 13, 2024.
5) M. Homaei, et al., “The Dark Side of Digital Twins: Adversarial Attacks on AI-Driven Water Forecasting,” arXiv preprint arXiv:2504.20295, Apr. 2025.
6) Author(s) unknown at index) “Integrating digital twin security simulations in the SOC,” ACM Digital Library / Conference Paper, (work on process-based security frameworks for SOC integration).
7) ABS, “Guidance Notes on Verification and Validation of Models, Simulations, and Digital Twins,” ABS Publication, Nov. 2024.
8) S. T. Mulder, et al., “Dynamic Digital Twin: Diagnosis, Treatment, Prediction Applications in Healthcare,” Frontiers / PMC, 2022.
9) M. Homaei and coauthors, “A review of digital twins and their application in cybersecurity and AI,” Artificial Intelligence Review / Survey, 2024.

10) M. Bordukova, “Generative artificial intelligence empowers digital twins,” Journal (Taylor & Francis), 2024.
11) “Digital Twins for Incident Detection and Response,” ResearchGate (systematic/position pieces summarizing DT role in incident detection & response).
12) (TechRxiv) “AI-Driven Digital Twin Framework for Security Threat Simulation, Compliance & Optimization,” preprint/tech report (AI-driven DT framework for threat simulation and incident triage).