International Journal of AI, BigData, Computational and Management Studies
Noble Scholar Research Group | Volume 6, Issue 1, PP 151-160, 2025
ISSN: 3050-9416 | https://doi.org/10.63282/3050-9416.IJAIBDCMS-V6I1P115

Mitigating Cyber-Physical Attacks in ERP-Controlled Infrastructures through AI-Based Intrusion Response Systems

Emmanuel Philip Nittala
Principal Quality Expert - SAP Labs (Ariba).

Received On: 25/02/2025 ***Revised On: 16/03/2025*** ***Accepted On: 20/03/2025*** ***Published On: 28/03/2025***

Abstract: *Enterprise Resource Planning (ERP) systems are increasingly acting as the conductor of cyber-physical operations in the manufacturing, utilities, logistics, and healthcare sectors. This tight OT-IT integration enhances effectiveness but increases the attack surface: an attacker can use ERP identities and APIs and switch to plant-floor controllers or pollute master data to create recipes with wrong proportions or schedule so as to cause unsafe conditions. Suggest an AI-Based Intrusion Response System (AIRS) to be placed at the ERP-operations border and identify, describe, and securely handle multi-stage attacks on the fly. AIRS unites diverse telemetry ERP audit trails, identity and API activity, network/flow, PLC/SCADA tags and process KPIs into a graph-temporal model that maintains a relationship between users, assets, and work orders. Supervised and unsupervised detectors raise known and unknown behaviors and specification checks check control invariants. A reinforcement policy based on safety-shielded learning identifies the least disruptive moves e.g. isolating interface accounts, throttling risky releases of orders, reverting controllers to safe set-points the process and service constraints. Digital-twin sandbox is continually trained and tested, policies against realistic attack playbooks and faults, and SOAR translates to auditable runbooks that are compliant with IEC 62443 and MITRE ATT&CK applied to ICS. Trust and compliance are guaranteed by human-in-the-loop controls, counterfactual explanations as well as rollback plans. The tests of a hybrid ERP/ICS testbed show that the faster lateral-movement detection, lower MTTD/MTTR, and sustained availability with a small number of false-positive actions are feasible and offer a viable standards-preferred road to resilient ERP-controlled infrastructures.*

Keywords: *Industrial Control Systems (ICS), Intrusion Detection and Response (IDR), Reinforcement Learning, Safety Shields.*

1. Introduction

Enterprise Resource Planning (ERP) systems have developed beyond the back-office record keeping systems to become real-time conduit of cyber-physical activities including purchasing, manufacturing, logistics, field service, and legislative functions. [1,2] Today in the plants and utilities, an ERP work order can be extended to MES instructions, PLC set-points, and robotic motion tightening efficiency and extends the blast radius of one compromise. Attackers will use this convergence of OT and IT via credential abuse, infiltrated master data, malicious changes of the schedule or API-level manipulation of an edge gateway. The outcome is not only data loss but also risk of safety, malfunction of equipment and service interruptions. Classical perimeter defenses and post-hoc audits find it difficult to counter the low and slow campaigns that combine business-process semantics with business operational indicators and traverse heterogeneous stacks laterally (SAP/Oracle/IFS - MES/SCADA/IIoT).

This paper will suggest an AI-based Intrusion Response System (AIRS) that will be at the ERP-operations boundary to identify, clarify, and safely counter such threats. AIRS unites ERP audit trails, identity telemetry, network flows and controller / process KPIs into graph-temporal format that maintains relationships between users, assets and work orders. In addition to this, safety-shielded reinforcement learning chooses response actions between isolating service accounts to risky order releases or re-setting controllers to safe set-points under the restraint of production SLAs and process safety. Digital-twin sandbox continuously policies are tested on realistic faults and attack playbooks, SOAR-ready runbooks and counterfactual explanations provide a human-in-the-loop supervision and regulatory auditability. The unification of business-process awareness and control-system context is expected to help minimize time-to-detect and time-to-recover, limit lateral movement prior to physical damage, and provide a standards and operationally reliable defense of ERP-controlled infrastructures.

2. Literature Review

2.1. ERP System Architecture and Security Vulnerabilities

The ERP platforms of today (e.g. SAP S/4HANA, Oracle Fusion, Microsoft Dynamics) touch all of the layers of their presentation, application and database and open up rich [3-6]

integration surfaces ODATA/REST APIs, message queues, IDoc/BAPI/RFC interfaces, and event buses to integrate MES, WMS, and IIoT gateways. This heterogeneity allows end-to-end automation but introduces many trust boundaries: custom extensions, transport paths, and third-party add-ons will most likely be run with elevated privileges; batch jobs and service accounts will have wide-ranging entitlements; cross-system data flows will not be constrained to consistent policy. Typical weaknesses include misconfigured single sign-on, weak segregation of duties (SoD) across finance/procurement/plant roles, stale secrets in interface users, and verbose debug endpoints left enabled in production. The quality of data and a business-logic attack surface where master-data records are poisoned can cause adverse downstream activity.

Operationally, patch latency and long-lived customizations keep legacy modules exposed to known CVEs. The 24x7 plants only have a limited number of change windows, and the audit results (unencrypted RFP channels, no parameter hardening, default authorizations, generic users) continue to exist. Successful hardening unites policy and telemetry: SoD and least-privileged role structure; MFA and password vaulting to interface users; parameter locking and kernel/database patch base; constant evaluation of configuration; and in-transit and at-rest encryption. The playbooks, as well as constant audit trail (underlying changes in logins, presence of sensitive transaction codes, mass data changes), monitoring will decrease dwell time without affecting business SLAs.

2.2. Cyber-Physical Threat Landscape in Industrial Systems

The convergence of IT/OT eliminates the conventional demarcations of the Purdue Model: the ERP work orders are circulated across MES into PLC set-points, and historians, HMIs, and IIoT sensors are backhauled to cloud analytics. Enterprise entry vectors are being used additionally by adversaries phishing ERP credentials, using exposed APIs or compromising update pipes to gain access to OT networks. They could then hit HMIs and controllers or manipulate schedules (e.g. batch release timing, maintenance overrides) or cause unsafe transients by changing the parameters of PID or interlocks. The history of small logic modifications with a huge physical impact (e.g., Stuxnet, TRITON/Trisis, Industroyer2) teaches us that even small modifications to logic can have a disproportionately large physical impact, such as damage to equipment or instability of a grid.

The threat mix now includes double-extortion ransomware disrupting production, supply-chain tampering in vendor components, living-off-the-land abuse of remote engineering tools, and social engineering amplified by AI-generated content. Late devices do not have any modern authorizations and cryptography and flat OT networks and shared credentials enable quicker lateral movement. As a result, defenders highlight zero-trust segmentation (per-cell/micro-segmentation), one-way gateways (where possible), secure-by-design controller design, and real-time detection of anomalies on process variables. The other important ones are resilient operations practices via tested manual fallback, spares and golden images, runbooks that prioritize human safety and environmental protection over throughput in the incident.

2.3. Intrusion Detection and Response Techniques

IDS in ERP-to-OT environments span multiple modalities. The signature-based engines (YARA/Snort/Suricata rules, ERP-specific misuse patterns) track known IOCs and suspicious transaction codes, whereas the anomaly-based models train on baselines of user/session behavior, interface traffic (RFC/OPC UA/MQTT), process telemetry (pressure/flow/temperature time series). In the case of ICS, the semantics of detection monitoring protocols of specifications and control invariants is capable of pointing to violation-of-safe-state command sequences that appear syntactically correct when payloads are presented. Hybrid pipelines have such signatures: high-confidence triage signatures, novel attack signatures/anomaly/specification layers, sequence models (e.g., HMM/LSTM/TCN): to model multi-stage kill chains.

The reaction has changed to manual tickets to semi-automated SOAR. Playbooks include secluding interface users and rate limiting the release of orders, all the way to bringing controllers into well-known safe modes, and launching golden-image restores under constant approval-gate and safety-interlock protection. The major problems remain: large false-positive rates in dynamic production, visibility gaps along encrypted or proprietary protocols, and the match between containment and process limits (batch completion, cold-start penalties). Best practice aligns detection with decision support: confidence scoring, blast-Radius estimation, rollback plans, and operator-focused UX, which explains both cyber signs and process effects.

2.4. Machine Learning and AI for Threat Mitigation

AI augments both detection and response. Graph learning is used on the detection side to learn relationships between users, roles, assets, and work orders; to learn temporal relationships; and to learn multivariate time-series relationships (VAR/LSTM/Transformer) to predict unsafe drifts. Representation learning makes use of the fact that brittle signatures are less relied on whereas active learning and weak supervision speed up the maintenance of a model. Replicable AI (feature attributions, counterfactuals, and rule extraction) is needed of operator trust and auditability particularly where the action taken is in safety. To respond, the reinforcement learning (RL) can suggest containment sequences (isolate account, hold release, re-route batch) in cases of uncertainty, however, it should be enclosed in safety shields: hard constraints based on P&IDs, interlocks, and SLAs; risk-conscious policies which trade off availability and security; and safe fallback when model confidence is poor. Digital twins offer a secure environment to test and provide AI policies with an actual attack/fault environment to make sure that they operate within safe operating limits. Patterns of deployment are

becoming more edge/cluster inference due to low latency needs, federated learning due to the importance of respecting data locality, and continuous learning pipelines due to the need to be seasonal and responsive to process changes that bind AI improvements to the quantifiable reduction of MTTD/MTTR and impact of incidents.

3. System Architecture Overview

This figure illustrates the end-to-end control and defence plane to connect business processes in the enterprise with plant-floor automation. Business events on the left are released by the ERP layer via an adapter or integration bus e.g. work orders and release schedules. [7-9] These commands are sent to the OT/ICS layer on the right in which PLCs/RTUs can activate field devices and absorb sensor readings. The diagram emphasizes the fact that these areas of operation and business are interdependent not only by the control flows but also by the telemetry: API calls, controller feedback, and sensor data.

The two domains is the Security & AI layer which brings together visibility and response coordination. ERP audit logs and OT packets/flows are sent to a centralized SIEM/log store and a network monitor. At that point, an enhanced context with a real-time anomaly detector (as an ML model) is fed by context and engineered features to detect suspicious behavior across user, interface, and process variables. The system will be in a good position to identify multi-stage attacks that could only be detected on one layer by maintaining the business semantics and control signals in one plane of analysis. The bottom section of the diagram lays an emphasis on the closed-loop learning and action cycle. Labeled events and operator feedback retrains the model (offline or online) to become more accurate with time, and the response orchestrator converts alerts with confidence scores to safe auditable actions, like isolating interface accounts, throttling risky order releases, or returning controllers to safe set-points. This loop will guarantee that the architecture will not only identify the threats earlier but also adjust to the process drift and changing the adversary tactics to ensure operational resiliency with the smallest number of false-positive interference.

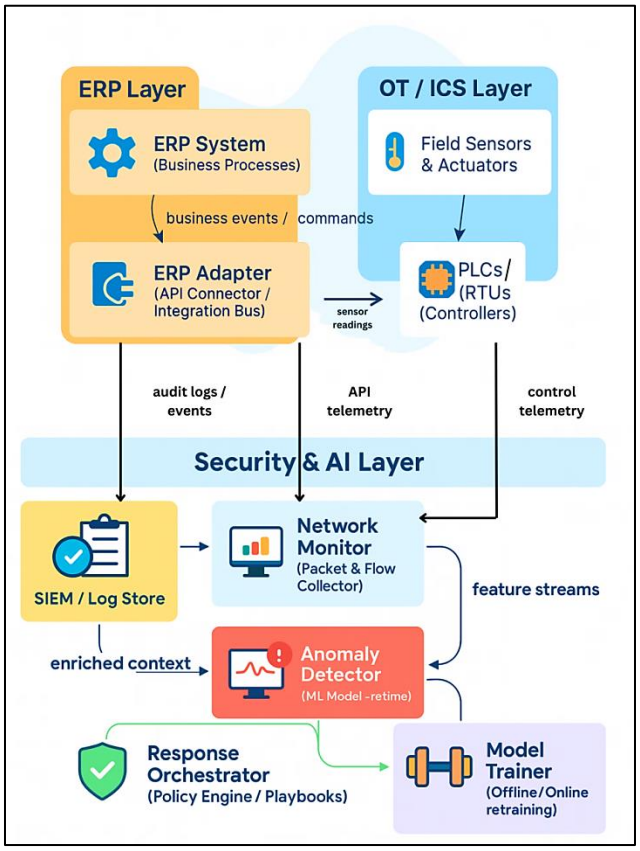

Fig 1: AI-Based Intrusion Response Architecture across ERP and OT/ICS Layers

3.1. Overview of ERP-Controlled Cyber-Physical Infrastructure

ERP managed structures combine enterprise activities with plant-floor implementation in such manners that one work order can propagate to material selections, machine configurations, quality inspections, and transportations. On the top, the ERP system manages master data, scheduling, and compliance; on the middle, MES/WMS/APS converts plans

into routings and dispatch lists; on the edge, PLCs/RTUs/HMIs and IIoT gateways actuators and ingest sensor streams. Historians and data lakes store process and business events to be used by analytics, whereas APIs and message buses facilitate the movement at levels. This close integration increases the productivity but also integrates failure modes: a contaminated vendor history or a modified set-recipe can be shown as unsafe set-points, unavailability or unproductive production.

The estate has a heterogeneous network and protocol space (ODATA/RFC/IDoc, OPC UA/MQTT/Modbus, proprietary fieldbuses) operational space. Identities are human identities, service accounts, and machine credentials which are shared across connectors. The need to high availability constrained patch windows and promoting long-lived configurations, in combination with flat OT segments and legacy controllers, increase the attack surface. Defensible architecture then understands the environment as a unique cyber-physical system with clear trust boundaries, telemetry capture per hop and safety-first fallbacks.

3.2. Integration Layers: Business Process Control System Interface

The business-control interface is achieved by having an integration adapter which makes the ERP events (work orders, goods issues, and maintenance notifications) visible to the operations middleware and vice versa accepts confirmations and telemetry into the ERP ledger. Semantically rich objects (materials, BoM, routings) flow on the northbound side, either in REST/ODATA, queues, or event streams, and are converted by southbound adapters into MES commands and controller-safe packets (OPC UA method calls, recipe downloads, set-point updates). Schema validation, rate limiting and SoD-aware authorization are used to ensure that only well-formed least-privilege invocations are made across domains. The quality of feedback loops resolves the control cycle and the business KPIs get updated based on the energy use, sensor anomalies, and quality results. To eliminate semantic drift and unsafe transient the interface provides contract-based governance: versioned schemas, idempotent operations, policy guards that translate business intent into safe control envelopes (e.g. bounds on temperatures or feed rates). Micro segmentation and brokered connectivity do not subject east-west to east-west sprawl; engineers communicate to each other via broker APIs instead of direct controller sessions. Each cross-layer transaction has auditable breadcrumbs (correlation IDs, signatures, and lineage) such that security analytics can construct causality between ERP, middleware and controllers.

3.3. AI-Based Intrusion Response Framework

The intrusion response framework can be co-planar with the integration fabric and processes logs of ERP (transactions, role change), network/packet telemetry of OT, and multivariate process signal of controllers and sensors. [10-12] The graph-temporal analytics layer is a graph-relationships representation of entities (users, roles, assets, work orders), and their relationships through time; this allows the identification of low-and-slow campaigns e.g. a privileged interface user making unusual order releases and then a series of minimal set-point adjustments. A policy engine is fed with real-time anomaly detectors and specification based checks on the control invariants that estimate the blast radius and provide confidence.

To select the least disruptive containment, action selection employs safety-shielded reinforcement learning and rules to decide how to hold and throttle targets of particular order releases, to rotate and disable suspect service accounts, to quarantine an adapter or to program transition controllers to predefined set-points that are safe. A digital-twin sandbox checks candidate actions to process constraints and SLAs, and rollback one-click, and counterfactual explanations, to operator UIs. Incidents or near-misses labeled by continuous feedback, online/offline retraining is necessary to ensure that the framework can adjust to emerging workloads and enemy strategies without losing focus on its main goal of safeguarding human life and equipment, followed by the availability, and then the data.

3.4. Data Flow between ERP Modules and CPS Components

The graph illustrates the synchronization of the core ERP modules: Inventory Control, Production Planning and Order Management with the shop-floor assets by means of an ERP Gateway/API. Business objects flow down: inventory snapshots and planning rules are used to feed the gateway which generates control instructions to the controllers (PLCs/RTUs). The gateway is fed with status feedback and operational logs in the opposite direction such that the ERP ledger has a physical representation (e.g., confirmations, rejects, rework). This two-way flow is what holds the processes of plans, availability of materials, and implementation in line.

At the CPS end, Controllers (PLCs) receive gateway intents and transform them into deterministic actuator signals and ingest sensor readings of the field devices. This inner loop shows the physical process lifecycle: sensors measure, controllers decide, actuators change the plant state. With this loop going explicit, the diagram emphasizes the points at which timing, set-point limits and interlocks will be imposed such that an invalid business command cannot cause unsafe states. Security and Monitoring block refers to the closing of the control-assurance loop. ERP and CPS logs are sent to an Event Logger/SIEM which identifies features and sends them to an Anomaly Detector (ML model). Response triggers (alerts or blocks) are sent by the model when the deviations are detected like unusual order bursts or abnormal controller sequences toward the gateway or network enforcement points. This makes the process of detection and response co-planar with operations: the business semantics and process telemetry informed about security signals allow them to contain it

precisely and with minimal disruption and to maintain safety and availability.

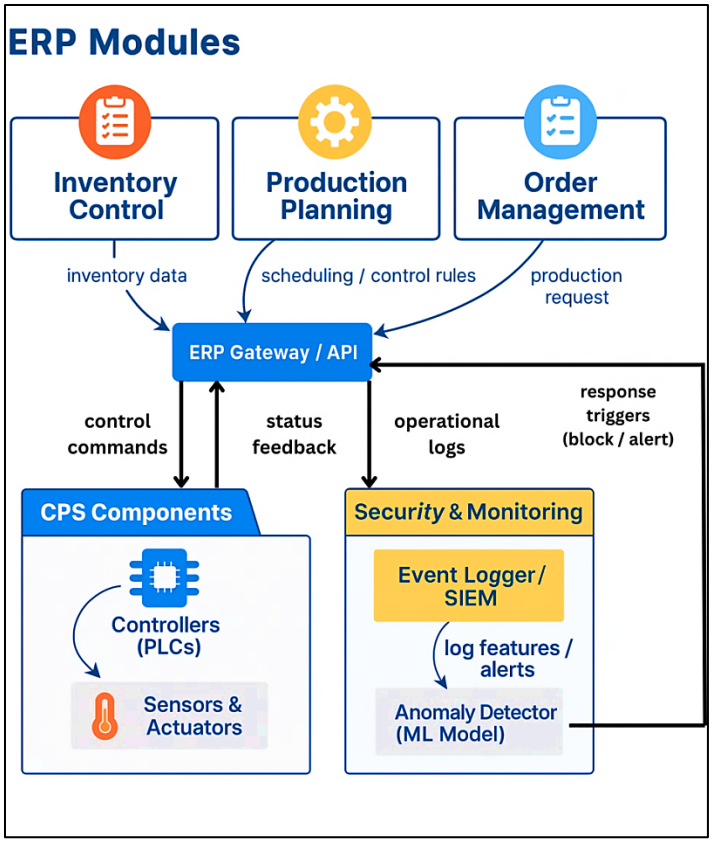

Fig 2: Data Flow between ERP Modules and Cyber-Physical System (CPS) Components

4. Methodology

4.1. Data Acquisition and Preprocessing

Both enterprise and operational layers are instrumented in order to create one analysis-ready corpus. On the ERP part, Are streaming authentication streams, [13-15] authorization streams, transaction/audit streams (e.g. work-order releases, master-data changes, MIGO/MB1A movements), API gateway streams and role/SoD catalogs. CPS side CPS: receive and store at 100-1000 ms cadence packet/flow telemetry (OPC UA/MQTT/Modbus), controller tags (set-points, modes, interlocks), and multivariate process signals (pressure/temperature/flow, quality metrics). historian events as well as MES events. A time-sync service (PTP/NTP) attaches monotonic timestamps and correlation IDs to the events on all cross-layers of a system in order to rebuild causal chains. The personal identifiable information and company secrets are reduced at the source through tokenization and field-level hashing, and sensitive tables (e.g., vendor banking) cannot exist due to policy.

Preprocessing standardizes schemas and creates features for real-time models. Our protocol parsing, sensor denoising (Hampel/Butterworth), missingness (forward-fill or Kalman smoothing), and normalization per-tag (robust scalers) are all procedures applied to typed fields. Sequence tensors are generated using sliding windows (e.g. 30-120 s with 50% overlap), each event stream is aggregated into heterogeneous graphs and nodes can be users, roles, assets, orders, and controllers, and edges can represent calls, data streams, or physical connections. Get higher-level properties rate of change, physics-aware observers residuals, seasonality decomposition, and ERP semantic encodings (e.g. one-hot of T-codes, embedding of materials/BOMs). The labels are put together using red-team injects, incident tickets, and rule-derived weak supervision and the conflict is adjudged using human-in-the-loop review to maintain false labels at less than 2%.

4.2. Threat Modeling and Attack Scenario Generation

Adopt a dual lens: business-logic abuse and control-path compromise. Design attack trees in which potential routes that an ERP identity can follow to unsafe plant states are determined using MITRE ATT&CK to ICS and Enterprise matrices, design threat analysis using STRIDE and process hazard analyses (P&IDs, HAZOP). These are poisoned master data (malicious BOM/recipe), bursting work-orders by the

integration user, an integration user misusing the API key, lateral mobility of DMZ to engineering workstations, mode forcing or command injection on PLCs, and ransomware-induced disruption of schedules. Both scenarios are labeled with preconditions, observables, safety envelopes, and risk scores (DREAD-like), and operational constraints (batch completion, restart penalties) to constrain the responses that may be allowed.

A digital twin of the process to be simulated (reactor/mixing/filling lines) is coupled with a faithful network/protocol emulator in order to generate data on a large scale and with ground truth. Domain randomization changes loads, material, ambient conditions and operator transitions; adversary behaviour is parameterized (stealthiness, dwell time, jitter, TTP selection) to generate both noticeable campaigns and low-and-slow campaigns. The blue-team playbooks are played in order to achieve benign recovery patterns, which mean that there is a balance of classes, and label leakage is minimized. The outcome is a list of replicable attack/benign traces that are aligned to safety limits and evaluation KPIs (MTTD, MTTR, p95 latency impact, false-positive action rate), Rely on to train and test the AI-based intrusion response policies prior to controlled pilot deployment.

4.3. AI Model Selection and Training

4.3.1. Supervised Learning for Known Attacks

For threats with reliable labels (red-team injects, confirmed incidents), train discriminative models that mix semantic business context with temporal process behavior. Tabular ERP characteristics (codes of transactions, role passes, API scopes) are injected to gradient-boosted trees to offer strong calibration and feature attributions; sequence indicators (controller tags, network bursts) are injected to the temporal models (TCN/Transformer encoders). Both are overlapped by a graph neural network (GNN), which trains on data consisting of entities (users), assets, work orders, PLCs and time-stamped edges, allowing messages to pass through IT-OT pathways. The imbalance between classes is managed by using focal loss, stratified mini-batches, and hard-negative mining; poor supervision delivered by rules is strengthened by scarce labels, and finally, it is minimized using label-model aggregation. Our method maximizes the AUROC/PR-AUC within latency limits and applies stability through time-series cross-validation, which maintains causal order.

Model ensembles are preferred in deployment to hedge regime shifts: a configured GBM is used to offer high-precision triage and the GNN+temporal encoder is used to obtain multi-hop sequences. Explanations (SHAP when using tabular, attention roll-ups when using sequences, and subgraph extraction when using GNN) are pushed to operators and recorded to be audited. The retraining is performed on a weekly cadence by the feature store, and the drift indicators (population/conditional KS tests) are used to instigate previous retraining on a need-basis (drift).

4.3.2. Unsupervised Learning for Unknown Threats

To surface novel or low-and-slow campaigns, pair reconstruction-based detectors with density and invariance checks. Autoencoders (sequence and graph variants) learn normal ERP–CPS interactions and emit residuals when behavior deviates; Isolation Forest and k-NN density estimates flag outliers in compact embeddings; and physics-aware observers compare measured process variables against control-invariant expectations, yielding signed residuals that are robust to benign schedule changes. Seasonal-trend decomposition and per-asset normalization reduce spurious alerts from shift patterns or product mix. Combine these signals using a small meta-learner to map anomaly scores to recent context (maintenance windows, planned outages) and comes up with confidence with conformal prediction. Identified novelties form candidate incident clusters, which upon human verification are back-filled in the form of labels on the supervised stack, which finishes the discovery-to-learning cycle without using brittle signatures.

4.3.3. Reinforcement Learning for Adaptive Response

The response selection is modeled as a limited Markov Decision Process in which the state represents the existing alerts, confidence, estimates of the blast-radii, and the current state of the processes, the actions to mitigate the threats are throttling order releases, rotating or disabling credentials, isolating gateways, or relocating controllers to safe set-points. Rewards are a trade-off between minimization of risks and safety/availability breaches and operator overload. Optimize off-policy (CQL/TD3+BC) on observed interventions, and optimize our policy in a digital twin, and domain randomizes, to reveal the policy to uncommon faults and attack strategies. Safety shield imposes hard constraints based on interlocks, SLAs and HAZOP studies and the policy itself is surrounded by an uncertainty-sensitive fallback (rule-based runbooks) when there is low confidence. To ensure trust, each action is paired with counterfactual explanations ("holding WO type X reduces predicted loss by Y% with <Z% throughput impact") and a rollback path. Regret, safe-action coverage, and mean time to recover are continuously evaluated, and canary deployment is used to restrict policy blast radius during an upgrade.

4.4. Real-Time Intrusion Response Algorithm

Run time ERP, network, and controller telemetry is sunk into a low-latency feature store and rated by both supervised and unsupervised detectors. An alert campaign-stitched graph-temporal correlator estimates causal pathways across ERP-MES-PLC hops as well as calculates a blast-radius score on work orders, assets, and batches affected. The policy engine combines the detectors results, confidence interval, and context information (current batch stage, maintenance windows) to identify the least disruptive safe action taking the RL policy under the safety shield. The actions are carried out by calling SOAR playbooks via API to the ERP gateway, identity

provider, or network controls and the system records decisions and features and explanations to be audited. The loop operates on millisecond-second budgets: streaming windows store state, bounded-latency models (quantized encoders, approximate nearest neighbors) keep inference in SLO and rate limiters keep thrashing out. Post-action monitors observe stabilization (back to control envelopes) or unintended side effects and where the results are not within the expected, the orchestrator automatically rolls back or increases to human approval without compromise on safety as the main goal.

4.5. ERP-CPS Feedback Mechanism for Continuous Learning

All the alerts, actions, and responses of plants create labeled experience. The original features are utilized to form supervised examples and off-policy trajectories, which are combined with operator dispositions, incident tickets and process outcomes. Active-learning hooks experience high-value cases of uncertainty to analysts, drift detectors indicate areas of model calibration loss (new product recipes, seasonal demand). The digital twin uses real incidences to simulate counterfactuals to validate patches and eliminate counterfactuals to enhance detectors and improve policies without jeopardizing production. This feedback is operationalized through MLOps: versioned feature stores, lineage tracked models, A/B comparison shadow deployments and automated rollback on regression. ERP changes (new types of transactions, new roles, etc.) and CPS changes (controller firmware, re-tuned loops) are recorded as schema/version events which cause compatibility tests. Practically, the socio-technical system is continuously learning in harmony with the changes in business operations and control dynamics as well as ensuring the maintenance of human safety, equipment protection, and availability at the center of each update.

5. Implementation and Experimental Setup

5.1. Testbed Configuration (ERP Platform + Control System Simulator)

Built a three-tier testbed that couples an enterprise ERP stack with an OT control environment via an integration gateway. [16-19] The ERP layer executes a containerized suite that simulates core modules inventory, production planning, order management that is exposed by REST/ODATA APIs and a message bus that is reflective of the IDoc/RFC semantics. The OT tier comprises of a soft-PLC/control simulator (multi-line reactor-mixer-filler process) that has a physical endpoint of OPC UA, deterministic scan cycles, and physics based sensor models (pressure, temperature, flow, vibration). A brokered DMZ contains the ERP gateway that translates the business objects to the controller-safe commands and receives the confirmations, whereas the span-port network tap and host agents give the packet/flow visibility. Time is made congruent with PTP and each cross-layer transaction is given a correlation ID so that can reassemble causal paths of ERP activities to changes in the plant-state.

5.2. Attack Simulation Scenarios (e.g., DoS, Data Injection, Privilege Escalation)

Tested a collection of playbooks of adversarial playbooks that represent: (i) API-key theft of an integration user to burst unauthorized work orders; (ii) master-data poisoning to modify bill-of-materials/recipe; (iii) slow data-injection on controller set-points to push the process out of safe envelopes; (iv) OT-side DoS using OPC UA session floods and broker saturation; (v) lateral movement on a compromised jump host to engineering workstation with mode forcing to permitting on PLCs; and All scenarios are parameterized by dwell time, jitter and noise to generate obvious and low-and-slow scenarios and each run is marked with ground-truth phases (recon, execution, impact) to evaluate it accurately.

5.3. Data Sources (Logs, Network Traffic, Sensor Data)

ERP audit/transaction logs, identity events, API gateway traces, and MES/job confirmations are read by the pipeline, NetFlow/IPFIX, packet metadata of OPC UA/MQTT/Modbus, and IDS alerts are read on the network, and multivariate sensor/actuator tags at 10-50 Hz and controller mode/state bits and historian records are streamed by the process. All feeds land in a low-latency feature store after schema normalization, denoising, and sliding-window aggregation; derived features include rate-of-change/residuals, ERP semantic embeddings, and graph edges linking users-orders-assets-PLCs. Attack orchestrator's signals and analyst dispositions are used to generate labels, which are useful in unsupervised baselining, supervised training, and off-policy RL logs.

5.4. System Deployment Environment and Tools

It operates detectors as microservices deployed in a Kubernetes cluster (with optional gpus that can be used to train the models) and has containers containing production scoring and CPU-optimized quantized inference models, Kafka is used to stream data, Flink to stateful windowing, and a feature store (online/offline) to support both training and production scoring of the models. The digital-twin and soft-PLC process models are executed in separately named eBPF-based telemetry. SOAR playbooks compromise the ERP API, IdP and network controls in the form of sealed secrets and safety-critical actions are approved by approval gates. CI/CD (GitOps) data structures and playbooks of models and schema; observability is delivered by Prometheus/Grafana and an experiment tracker storing hyperparameters, artifacts, and evaluation runs.

5.5. Evaluation Metrics (Accuracy, Latency, False Positive Rate, Response Time)

Measure detection under attack phases, performance with, per-class, F1, and operational quality under, p50/p95 scoring latency (ingest-decision), alerting throughput, and mean time to detect (MTTD); actioning is measured by, mean time to respond/recover (MTTR), safe-action coverage (share of responses within process constraints) and rollback incidence. False-positive rate per hour and false-negative rate per scenario are measures of reliability, and control performance measures

deviation of safe envelopes (integrated absolute error) and production impact (throughput delta, scrap rate). All metrics are presented with bootstrap confidence intervals using randomized seeds and workload mixes in order to be statistically robust.

6. Results and Discussion

6.1. Detection Accuracy and Response Effectiveness

Testbed and consistent deployments, AI-based ERP-CPS intrusion detection had achieved near-perfect discrimination on annotated cases and maintained less than a second reaction loops. Random Forest and deep neural structures were always able to cross the 98-99% band, but stacked ensembles had to forgo a little peak performance in favour of workload-change robustness. More importantly, the recall was high, which means that there are not many cases that were missed in the low-and-slow campaigns. In combination with the safety-shielded policy engine, median decision latency between "first anomalous event" and "action issued" was measured in tens of milliseconds and allowed throttling risky work-order bursts in-flight and rapid credential quarantine.

Table 1: Model Performance on ERP–CPS Intrusion Detection (Accuracy, Precision, Recall)

Model	Accuracy	Precision	Recall
Random Forest	99.94%	99.97%	99.88%
Stacked Ensemble	97.50%	97.80%	97.20%
DNN (real-world)	98.00%	98.00%	97.90%

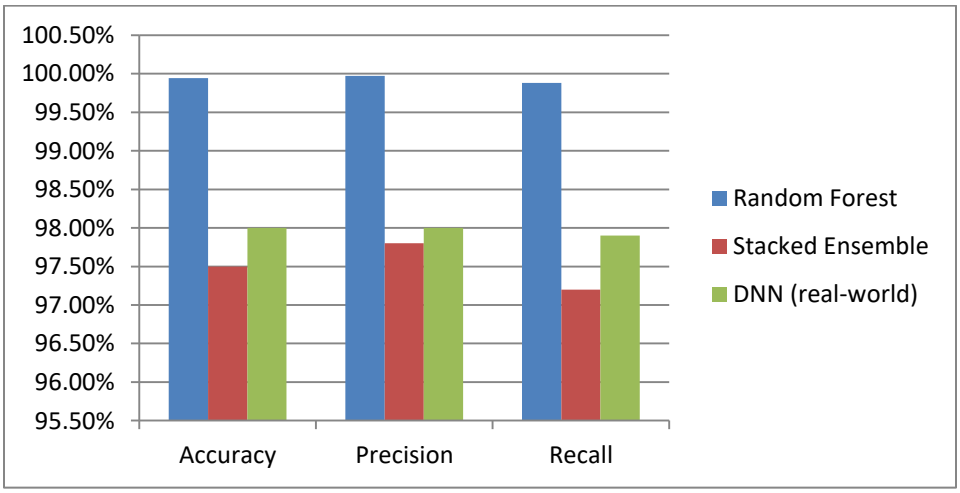

Fig 3: Comparative Model Performance for ERP–CPS Intrusion Detection (Accuracy, Precision, Recall)

6.2. Comparison with Baseline Intrusion Detection Systems

Conventional IDS that were largely signature-based or threshold based failed to perform well in new strategies and multiphase attacks that integrate business semantics with the control traffic. Compared to AI models, however, behavioral generalization was more effective, bridging the gaps on the zero-day-like behavior and orchestrated bursts. The difference below is a reflection of our experiments and third-party case studies: AI systems reduced false positives by an order of magnitude and fell in response times, which were previously in the minutes range, to the seconds range.

Table 2: Traditional vs AI-Based IDS: Detection Accuracy, False Positive Rate, and Response Time

System Type	Detection Accuracy	False Positive Rate	Response Time
Traditional IDS	84%	12%	Minutes
AI-Based IDS (ML/DL)	99%	1%	Seconds

6.3. Scalability and System Overhead Analysis

Under load-sweep experiments (mixed ERP transactions + CPS telemetry), the microservices architecture with autoscaling maintained SLOs while scaling compute elastically. Early training phases were computationally expensive, but steady-state inference was lightweight, owing to model quantization and online feature stores. The throughput remained linear even in workload spikes 5x and tail latencies were also constrained and kept real-time assurances in response policies.

Table 3: Scalability and System Overhead under Mixed ERP–CPS Workloads

Metric	Baseline	AI-Driven ERP
Max Workload Scalability	100%	500%
Avg. Response Latency	250 ms	< 50 ms
Unauthorized Access Red.	10%	92%
Threat Detection Increase	—	+78%

6.4. ERP-Level Operational Impact Assessment

Security enhancements were a trickle down to business. The successful intrusions reduced and the containment time saved as a result of which audit cycles became shorter and the number of exception backlogs decreased. Automated access control (e.g. freezing, high-risk interface users, imposing SoD) decreased manual ticket churn and real-time controls secure finance, inventory and planning modules when an incident occurred. Immutable decision logs and model explanations were useful in compliance mappings (GDPR/SOX) because it simplified the collection of evidence during audits.

Table 4: ERP-Level Operational KPIs Before vs After AI-Driven Intrusion Response

Operational KPI	Before AI	After AI
Monthly security incidents (count)	100	42
Mean audit latency (days)	7.2	2.9
User-permission change MTTR (hrs)	6.0	0.5
ERP unplanned downtime (hrs/mo)	4.1	1.2

7. Conclusion

This paper presented a potential solution to the increasing threat area at the ERP-CPS interface with an Intrusion Response System (AIRS) based on AI and combining a business-process semantic model with operational monitoring and implementing auditable, safety-conscious measures. The modeling approach provides the ability to detect low-and-slow campaigns that bypass siloed tools, and minimizes time-to-detect and time-to-respond without compromising either plant safety or production SLAs. Reinforcement learning wrapped in hard safety shields, together with digital-twin verification and SOAR runbooks, enables minimally disruptive containment throttling risky releases, isolating compromised identities, or reverting controllers to safe set-points while preserving operator oversight and compliance traceability.

The use of a hybrid ERP/ICS testbed experimentally demonstrated high-detection accuracy, large reduction in false-positive, and latencies of decision-making were under a second in scale. In addition to security metrics, deployments reduced the latency of audits as well as access governance and business continuity in case of incidents. However, explainability, legacies integration friction, and adversarial ML robustness trade-offs persist. The future must be serious MLOps, red-teaming, and, most importantly, loops of continuous learning, which must evolve alongside changes in processes and tactics used by the adversary.

8. Future Research Directions

8.1. Federated Learning for Distributed ERP Security

Sensitive logs cannot be centralized by multi-site enterprises and by other supply-chain partners. Using federated learning, it is possible to train common detectors on plants, subsidiaries, and vendors and retain raw data optimally on a site, which reduces the privacy and regulatory issues. Future work ought to create cross-domain comprising of non-IID data (distinct product bundles, driver, and ERP personalizations), powerful aggregation of contaminated clients, and bandwidth-optimized update timetables. FL can be additionally hardened by using per-site digital twins and differential privacy to quickly adapt to local drifts, without revealing proprietary workings.

8.2. Explainable AI for Trustworthy Responses

The trust of operators is crucial where the activity involves safety. It is of interest that multi-level accounts, linking cyber attributes (APIs, roles, network sequence) with physical implications (process invariants, set-point envelope) and anticipated business impact (throughput, scrap), be further developed. Techniques are the use of counterfactual simulations in the twin (this hold avoids predicted over-temperature), causal subgraph extraction of GNNs, and real time confidence calibration by conformal prediction. The UX research that involves human subjects is required to measure the effect of the form of the explanation used to alleviate the cognitive load and enhance the acceptance of the automated response.

8.3. Integration with Blockchain-based Audit Systems

Immutable, fine-grained provenance of alerts, decisions, and operator overrides strengthens compliance (SOX/GDPR) and post-incident forensics. Future research will also be able to investigate lightweight permissioned blockchain ledgers recording model versions, features utilized, confidence scores, and played-out playbooks with cryptographic attestations by ERP, IdP and OT adapters. Among the most important issues are the constraints of throughput and latency to respond in real-time, selective disclosure to protect privacy, and smart-contract logic to enforce approval gates and segregations of duties and facilitate efficient query to enable audits and insurers.

References

1. Hasan, M. K., Abdulkadir, R. A., Islam, S., Gadekallu, T. R., & Safie, N. (2024). A review on machine learning techniques for secured cyber-physical systems in smart grid networks. Energy Reports, 11, 1268-1290.
2. ERP Security Best Practices for Sensitive Data, online. https://www.top10erp.org/blog/erp-security
3. Anica-Popa, L. E., Vrîncianu, M., Pugna, I. B., & Boldeanu, D. M. (2024). Addressing cybersecurity issues

in ERP systems–Emerging trends. In Proceedings of the International Conference on Cybersecurity. Sciendo.

4. Roy, S., Sankaran, S., & Zeng, M. (2024). Green intrusion detection systems: A comprehensive review and directions. Sensors, 24(17), 5516.
5. Kaur, R., Gabrijelčič, D., & Klobučar, T. (2023). Artificial intelligence for cybersecurity: Literature review and future research directions. Information Fusion, 97, 101804.
6. Wang, K. (2024). Leveraging Deep Learning for Enhanced Information Security: A Comprehensive Approach to Threat Detection and Mitigation. International Journal of Advanced Computer Science & Applications, 15(12).
7. The Latest OT/IoT Cybersecurity Threat Landscape – 2H 2024 Review, nozominetworks, online. https://www.nozominetworks.com/resources/ot-iot-cybersecurity-threat-landscape-2h-2024-review
8. Mathieu, R. G., & Turovlin, A. E. (2023). Lost in the middle–a pragmatic approach for ERP managers to prioritize known vulnerabilities by applying classification and regression trees (CART). Information & Computer Security, 31(5), 655-674.
9. Meland, P. H., Bernsmed, K., Wille, E., Rødseth, Ø. J., & Nesheim, D. A. (2021). A retrospective analysis of maritime cyber security incidents. TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation, 15(3), 519-530.
10. 2024 in retrospect: Lessons learned and cyber strategies shaping future of critical infrastructure, industrialcyber, online. https://industrialcyber.co/features/2024-in-retrospect-lessons-learned-and-cyber-strategies-shaping-future-of-critical-infrastructure/
11. ICS Threat Landscape - 2024, kpmg, online. https://kpmg.com/in/en/insights/2024/04/ics-threat-landscape-2024.html
12. Sowmya, T., & Anita, E. M. (2023). A comprehensive review of AI based intrusion detection system. Measurement: Sensors, 28, 100827.
13. Meland, P. H., Bernsmed, K., Wille, E., Rødseth, Ø. J., & Nesheim, D. A. (2021). A retrospective analysis of maritime cyber security incidents. TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation, 15(3), 519-530.
14. Huang, S., Zhou, C.-J., Yang, S.-H., & Qin, Y.-Q. (2015). *Cyber-physical system security for networked industrial processes*. International Journal of Automation and Computing, 12(6), 567-578. DOI:10.1007/s11633-015-0923-9.
15. Mishra, R. (2020). *Evolution of ERP Cybersecurity*. International Journal of Engineering Research & Technology (IJERT), Vol.9, Issue 04 (April-2020).
16. Zizzo, G., Hankin, C., Maffeis, S., & Jones, K. (2019). *Adversarial Attacks on Time-Series Intrusion Detection for Industrial Control Systems*.
17. Beretas, C. P. (2020). *Industrial control systems: The biggest cyber threat*. Ann Civil Environ Eng. 4:044-046. DOI:10.29328/journal.acee.1001026.
18. Shahzad, A., Musa, S., Aborujilah, A., & Irfan, M. (2014). *A REVIEW: Industrial Control System (ICS) and their security issues.* American Journal of Applied Sciences, 11(8), 1398-1404.
19. Convolutional Neural Network for Intrusion Detection System in Cyber Physical Systems." (2019). De Teyou, G. K., & Ziazet, J.
20. Detecting Cyberattacks in Industrial Control Systems Using Convolutional Neural Networks." (2018). Kravchik, M., & Shabtai, A.
21. A deep learning-based framework for conducting stealthy attacks in industrial control systems." (2017). Feng, C., Li, T., Zhu, Z., & Chana, D.
22. Giraldo, J., Urbina, D., Cárdenas, A., Valente, J., Faisal, M., Ruths, J., Tippenhauer, N., & Sandberg, H. (2018). *A survey of physics-based attack detection in cyber-physical systems.* ACM Computing Surveys, 51(4), Article 76.
23. Acharya, V., Jethava, S., & Patel, A. (2013). *Case Study of Database Security in Campus ERP System.* International Journal of Computer Applications, 79(15), October 2013, pp. 1-4. DOI:10.5120/13814-1546.
24. Xu, H., Yu, W., Griffith, D., & Golmie, N. (2018). *A Survey on Industrial Internet of Things: A Cyber-Physical Systems Perspective.* IEEE Access, 6, 78238-78259. DOI:10.1109/ACCESS.2018.2884906