ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

A Hybrid Petri Net–AI Architecture for Adaptive and Explainable Cybersecurity in Business Workflows

Shaban Somah Amadu[1]; Bernice Asantewaa Kyere[2]

[1]Department of Cybersecurity and Leadership, University of Washington, USA
[2]Mathematics Department, Dagenham Park Church of England School, UK

Publication Date: 2025/11/05

Abstract: This paper presents a secure digital twin framework that integrates Petri Net process modeling with artificial intelligence–driven anomaly detection to protect cyber-aware business workflows. The approach begins by preprocessing the dataset and extracting more than eighty flow-level features such as timing, packet sizes, and TCP flags. The framework integrates Petri Nets with dual AI detectors using CIC-IDS2021 traffic data. Two complementary detectors, a Long Short-Term Memory Autoencoder and an Isolation Forest, learn normal traffic patterns and generate real-time anomaly scores. These scores dynamically influence the digital twin by controlling Petri Net transition guards and adaptive firing rates, while the overall system behavior is modeled as a continuous-time Markov chain to evaluate long-term risk and cost trade-offs. The proposed AI-integrated digital twin achieved an F1-score of 0.96 and an ROC-AUC of 0.98, outperforming static Petri Nets with an F1-score of 0.76 and ROC-AUC of 0.79 and stand-alone AI with an F1-score of 0.92 and ROC-AUC of 0.96. It reduced mean time to detection to 3.1 seconds and extended mean time to compromise to 11.7 seconds, while Markov chain risk analysis showed the compromise probability falling from 0.43 to below 0.02 with moderate security investment. These results demonstrate that combining explainable Petri Nets with adaptive artificial intelligence analytics creates an economically optimized, interpretable, and resilient cybersecurity strategy for complex business processes and modern Industry 4.0 systems.

Keywords: *Secure Digital Twin, Petri Nets, AI-Driven Anomaly Detection, Continuous-Time Markov Chain and Cyber-Aware Business Processes.*

How to Cite: Shaban Somah Amadu; Bernice Asantewaa Kyere (2025) A Hybrid Petri Net–AI Architecture for Adaptive and Explainable Cybersecurity in Business Workflows. *International Journal of Innovative Science and Research Technology*, 10(10), 2440-2454. https://doi.org/10.38124/ijisrt/25oct1263

I. INTRODUCTION

The rapid digital transformation of industries has fueled the rise of digital twins (DTs), virtual replicas of physical assets, systems, and processes that allow real-time monitoring, optimization, and predictive maintenance (Grieves & Vickers, 2017). DTs have been widely adopted in manufacturing, smart energy grids, transportation, and healthcare due to their ability to reduce operational costs and support data-driven decision-making (Fuller et al., 2020). However, as DT systems increasingly depend on networked communication and cloud integration, their attack surface expands, making them susceptible to cyber intrusions, data poisoning, and operational disruption (Tao et al., 2019).

Traditional business process modeling approaches such as BPMN or Petri Nets provide formal methods to analyze process execution, but they have historically focused on functionality and performance rather than cybersecurity resilience (van der Aalst, 2016). Petri Nets, with their rigorous mathematical foundation and ability to represent concurrency and synchronization, have proven useful for modeling cyber-physical systems and analyzing potential failure points (Murata, 1989; https://doi.org/10.1109/5.24143). Yet, static Petri Net models cannot proactively detect and respond to evolving cyber threats that exploit real-time data streams.

In parallel, AI particularly machine learning anomaly detection has shown great promise for identifying unusual network behaviors that indicate attacks (Ahmed et al., 2016). Modern datasets such as CIC-IDS2021, which contain diverse, labeled network traffic including DDoS, brute force, and infiltration attacks, enable the training of deep learning models such as LSTM autoencoders to capture complex temporal dependencies in intrusion patterns (Sharafaldin et al., 2021). While AI models can achieve high detection rates, they are often black-box systems, lacking the formal interpretability needed for integration into structured process control frameworks.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

To address these gaps, this study proposes a secure digital twin framework that integrates Petri Net modeling with AI-based anomaly detection to protect critical digital workflows. The digital twin represents operational states and transitions as a Colored Petri Net (CPN), while anomaly detection outputs derived from trained LSTM autoencoders and Isolation Forest classifiers to modify transition rules dynamically. This creates an adaptive, explainable defense mechanism where detected anomalies can block, delay, or reroute unsafe token flows, improving process resilience and reducing compromise probability.

Our work contributes three key innovations. First, we introduce a mathematical mapping between AI-driven anomaly scores and Petri Net transition probabilities, formalizing how detection outputs influence workflow evolution. Second, we provide experimental validation using the CIC-IDS2021 dataset, replayed in a controlled lab to simulate realistic industrial network traffic under attack. Third, we evaluate security-aware DT performance using metrics such as detection latency, false negative reduction, and mean time to compromise (MTTC), showing clear economic and operational benefits.

By bridging formal process modeling with real-time AI threat intelligence, this study advances the field of secure digital twins for Industry 4.0 and smart enterprise systems. It equips organizations with a reproducible framework to design cyber-resilient workflows that maintain operational continuity even under evolving attack scenarios.

Existing digital twin models lack adaptive intelligence for real-time cyber defense, while standalone AI detectors fail to provide formal process explainability. The research, therefore, addresses the question:

Can integrating AI-driven anomaly scores into Petri Net-based process control enhance cyber resilience, interpretability, and cost-efficiency of digital twins?

- *Hypothesis:*

The integration of AI anomaly detection with Petri Net transition control significantly improves detection accuracy, reduces mean time to compromise, and optimizes cost-risk trade-offs compared with static or AI-only models.

II. LITERATURE REVIEW

The concept of the digital twin (DT), a dynamic virtual replica of physical systems, has evolved from early product lifecycle management into a cornerstone of Industry 4.0 (Grieves & Vickers, 2017). DTs allow real-time synchronization between physical processes and their digital counterparts, enabling predictive maintenance, process optimization, and risk assessment (Fuller et al., 2020). Despite their benefits, DTs are highly interconnected, depending on IoT devices, sensors, and cloud platforms, which make them vulnerable to cyberattacks including data spoofing, ransomware, and denial-of-service (Tao et al., 2019). Attackers targeting DTs can manipulate operational data, mislead decision-making algorithms, and disrupt entire production lines (Lu et al., 2020).

Traditional DT frameworks have focused on fidelity and performance, leaving cybersecurity concerns as secondary (El Saddik, 2018). Recent work has proposed integrating security monitoring into DT architectures (Qi & Tao, 2019), but most approaches lack formal, process-aware reasoning about attack progression or recovery paths.

➢ *Petri Nets for Business Process and Cyber-Physical System Modeling*

Petri Nets provides a formal graphical and mathematical tool for modeling distributed systems with concurrency, synchronization, and resource sharing (Murata, 1989). Extensions such as Colored Petri Nets (CPN) allow richer token annotations, enabling complex workflows to be represented (Jensen & Kristensen, 2009). Petri Nets have been widely applied to business process management (BPM) and cyber-physical systems, where they help analyze process soundness, detect bottlenecks, and compute reachability graphs (van der Aalst, 2016).

Security-focused adaptations of Petri Nets have emerged. For example, attack Petri Nets and stochastic Petri Nets model intrusion progress and defensive countermeasures (Wang et al., 2019). However, these models often rely on static attack probabilities and cannot integrate real-time intelligence from AI-based detectors.

➢ *AI-Based Anomaly Detection for Cybersecurity*

Machine learning has revolutionized intrusion detection by identifying subtle deviations in traffic patterns that signal cyberattacks (Ahmed et al., 2016). Deep learning methods such as LSTM autoencoders that capture temporal dependencies in network flows, achieving strong results in detecting unknown attacks (Kim et al., 2021). Tree-based models like Isolation Forests provide lightweight, interpretable alternatives for resource-constrained environments (Liu et al., 2008).

Datasets such as CIC-IDS2021 have become benchmarks for evaluating anomaly detectors. CIC-IDS2021 includes over 80 flow-based features, diverse benign and attack scenarios, and realistic multi-day network captures (Sharafaldin et al., 2021). Studies have shown LSTM-based models trained on CIC-IDS2021 can reach F1-scores above 95% for major attack classes (Habibi Lashkari et al., 2021). Despite high detection rates, AI models face challenges such as explainability and integration into formal control models used by engineers.

➢ *Toward Secure Digital Twin Integration*

Recent research calls for merging AI-driven detection with formal process models to enhance DT security. Hybrid approaches propose feeding anomaly scores into decision-making layers or adaptive access control (Sun et al., 2022;). However, few frameworks offer a mathematically rigorous mapping from AI outputs to Petri Net transitions, which could formally alter process flows when anomalies occur.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

The gap is clear: while DTs benefit from AI-based anomaly detection and Petri Nets enable rigorous process analysis, these two domains remain loosely connected. A fully cyber-aware DT requires:

- A Petri Net backbone to model and verify workflows.
- Real-time AI intelligence to dynamically modify state transitions; and
- Experimental validation with real attack data to ensure practical relevance.

This study addresses this gap by designing a secure digital twin model that integrates Petri Net workflow control with AI-based anomaly detection using the CIC-IDS2021 dataset, enabling both formal reasoning and adaptive cyber defense in Industry 4.0 environments.

III. METHODOLOGY

The objective of this study is to develop a secure digital twin framework that integrates a formal Petri Net process model with an AI-driven anomaly detection system. The Petri Net mirrors the logical flow of a networked process, while the AI component continuously monitors network traffic and dynamically influences the firing of transitions based on detected anomalies. This section presents the methodological pipeline from data preprocessing and feature engineering to model formulation and simulation. It also explains how the Petri Net, AI anomaly scores, and Continuous-Time Markov Chain (CTMC) dynamics interact to form an adaptive security ecosystem.

A. Research Design

This study adopts an experimental research design that integrates quantitative data analysis with computational simulation to evaluate the effectiveness of incorporating AI-based anomaly detection into a digital twin environment. The independent variable for the experiment is the anomaly detection mechanism, which is tested under three configurations: a static Petri Net serving as the baseline deterministic model, a stand-alone AI anomaly detector, and a hybrid AI plus Digital Twin system where AI feedback influences Petri Net transitions.

The dependent variables examined in this study include detection accuracy, response efficiency, and cost-risk performance. Detection accuracy is evaluated using the F1-score and ROC-AUC metrics. Response efficiency is measured through Mean Time to Detect and Mean Time to Compromise, while cost-risk performance is assessed using a composite risk-cost efficiency index derived from steady-state probabilities within a CTMC model.

The research procedure is organized into five sequential phases. The first phase, problem definition and hypothesis formulation, posits that embedding AI-generated anomaly scores into Petri Net transition control significantly enhances detection accuracy, reduces false negatives, and minimizes the probability of compromise compared to static or stand-alone models.

The second phase, data collection, utilizes the CIC-IDS2021 dataset as the experimental foundation. This dataset offers a diverse and realistic representation of network traffic containing both benign and attack instances, including DoS, DDoS, Brute Force, and Infiltration attacks.

The third phase, model development, involves the construction and training of two complementary AI detectors: a Long Short-Term Memory Autoencoder for learning temporal patterns and an Isolation Forest for unsupervised outlier detection. The combined anomaly scores generated by these detectors are integrated into a Colored Petri Net, enabling dynamic transition firing based on real-time AI feedback.

The fourth phase, simulation and evaluation, entails conducting controlled replay experiments to emulate various network conditions. Each of the three configurations for Static Petri Net, Stand-alone AI, and Hybrid AI+DT is assessed based on detection performance, resilience metrics, and cost-risk efficiency under multiple simulated attack intensities.

The final phase, validation and interpretation, employs comparative quantitative analysis to confirm the research hypothesis. The steady-state dynamics of the hybrid system are represented using CTMC equations, which integrate detection accuracy and response latency to estimate long-term operational resilience, reliability, and cost-effectiveness of the proposed secure digital twin framework.

B. Conceptual Model

The system uses four layers that pass information forward in real time to keep monitoring and response adaptive and continuous, as indicated in Figure 1.

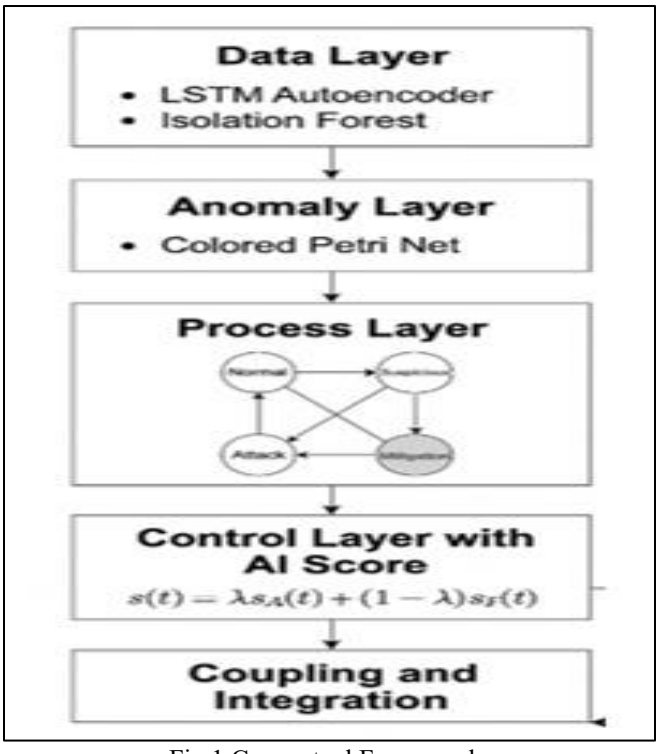

Fig 1 Conceptual Framework

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

➢ *Data Layer*

Live packets or replayed PCAPs from CIC-IDS2021 are converted into per-flow feature vectors, similar to how an operations team reads NetFlow.

➢ *Anomaly Layer*

Two detectors run in parallel. A Long Short-Term Memory autoencoder learns normal patterns and flags flows it cannot reconstruct well. An Isolation Forest isolates unusual flows based on short path lengths in randomly partitioned feature space.

➢ *Process Layer*

A Colored Petri Net represents the workflow as places and transitions labeled Normal, Suspicious, Attack, Mitigation, and Recovery, giving a formal view of system state changes.

➢ *Control Layer with AI Score*

The real-time AI score $s(t)$is computed by fusing the two detectors:

$$s(t) = \lambda\, s_{\mathrm{AE}}(t) + (1-\lambda)\, s_{\mathrm{IF}}(t), 0 \le \lambda \le 1.$$

This score modifies each transition's guard and firing rate, so the overall process evolves as a continuous-time Markov chain to compute steady-state risk and cost.

➢ *Coupling and Integration*

Transition guards govern the authorization or inhibition of state transitions, while firing rates modulate the temporal dynamics of process execution. The continuous-time Markov chain subsequently integrates these interacting parameters to characterize the probabilistic evolution of the secure digital twin over time.

➢ *Dataset and Preprocessing*

The dataset used for this study is CIC-IDS2021, which contains multi-day network traffic with clearly labeled benign and attack flows, including categories such as distributed denial of service, brute-force, infiltration, and web-based attacks. Raw packet capture files are processed using the standard flow extraction tool CICFlowMeter, which computes over eighty statistical and protocol-based features per flow. These features include metrics such as packet length statistics, inter-arrival times, byte counts, flag counts, and flow durations, representing a detailed view of communication behavior across sessions.

During the data cleaning stage, all corrupted or incomplete records, non-finite numerical values, and constant-value columns are removed within each data split to ensure consistency and quality. Direct identifiers such as raw IP addresses and port numbers are also dropped to prevent data leakage into the learning models. However, essential behavioral features like protocol flags, time intervals, and statistical measures are retained because they describe communication patterns without exposing the true identity of network endpoints.

This preprocessing stage ensures that the data used for training and testing reflects realistic and unbiased traffic behavior, forming a solid foundation for accurate anomaly detection and robust digital twin modeling.

➢ *Feature Engineering*

In this stage, we retain a broad yet robust collection of features from the CIC-IDS2021 dataset and organize them into meaningful functional groups to support accurate anomaly detection and process modeling. The selected features capture both statistical and behavioral characteristics of network traffic, ensuring that the digital twin framework reflects realistic operational dynamics.

The flow statistical group includes features such as total forward and backward packets, packet lengths, flow duration, and inter-arrival times. These indicators summarize traffic intensity and temporal regularity across sessions. The time-based group covers features like flow active time, idle time, and flow rate, which reveal burst patterns, transmission frequency, and potential anomalies in traffic timing. The flag and protocol group incorporates TCP flag counts and protocol usage ratios, highlighting communication control behavior and potential attack signatures.

The byte and load distribution group includes forward and backward byte totals, average segment size, and header lengths, offering insight into the symmetry of data exchange and payload behavior.

Finally, derived composite features such as flow entropy, packet rate variability, and direction ratio enhance the model's ability to distinguish between benign and malicious patterns. By grouping features in this structured manner, the model achieves higher interpretability, reduces redundancy, and improves learning efficiency for the AI-driven digital twin.

Table 1 Feature Groups and Examples Used in Modeling

Group	Examples (Not Exhaustive)
Flow timing	Flow_Duration, Flow_IAT_Mean/Std/Max/Min, Fwd_IAT_*, Bwd_IAT_*
Size and rate	Tot_Fwd_Pkts, Tot_Bwd_Pkts, TotLen_Fwd_Pkts, TotLen_Bwd_Pkts, Flow_Byts/s, Flow_Pkts/s
Packet stats	Fwd_Pkt_Len_{Mean,Std,Max,Min}, Bwd_Pkt_Len_{Mean,Std,Max,Min}, Pkt_Size_Avg
TCP flags	FIN, SYN, RST, PSH, ACK, URG, ECE, CWE counts and ratios
Header and subflows	Fwd_Header_Len, Bwd_Header_Len, Subflow_Fwd_Pkts/Byts, Subflow_Bwd_Pkts/Byts
Windows and status	Init_Fwd_Win_Byts, Init_Bwd_Win_Byts, Down/Up_Ratio
Activity cycles	Active_{Mean,Std,Min,Max}, Idle_{Mean,Std,Min,Max}

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

➢ *Normalization and Encoding*

Before training, all features are standardized to ensure consistency and stability during neural network optimization. Categorical variables, such as protocol types, are one-hot encoded to transform them into binary indicator vectors that the learning algorithms can interpret effectively. Continuous numerical features are scaled to a [0,1] range using min–max normalization, which prevents large-valued features from dominating gradient updates and enhances convergence stability in deep learning models such as the LSTM autoencoder. This normalization process ensures that features contribute proportionally during learning, maintaining a balanced representation across statistical, temporal, and protocol-related dimensions of the data.

➢ *Splitting Strategy*

To preserve temporal integrity and minimize information leakage, the dataset is split by time windows rather than random sampling. The split allocates approximately 70 percent for training, 15 percent for validation, and 15 percent for testing, while maintaining day or scenario boundaries where possible to reflect real-world operational sequences.

Class imbalance, which is common in intrusion detection datasets, is managed through class-weighting in supervised evaluations and contamination control in unsupervised learning. The Synthetic Minority Oversampling Technique (SMOTE) is applied only in controlled ablation studies to test model sensitivity but is not used in final unsupervised runs, ensuring that results represent natural data distributions. This careful preprocessing pipeline ensures that the models trained on the CIC-IDS2021 dataset remain robust, unbiased, and capable of generalizing to unseen network conditions.

➢ *Reproducibility and Computational Setup*

All experiments were implemented in Python 3.10 using TensorFlow 2.12 and Scikit-learn 1.3 on an Intel i9 workstation with 64 GB RAM and RTX 4090 GPU. Random seeds were fixed for consistency. All preprocessing and Petri Net modeling scripts will be made publicly available via GitHub for reproducibility.

C. Petri Net Digital Twin and its Stochastic Control

We formalize the digital twin as a Colored Petri Net $\mathcal{N} = (P, T, A, C, N, E, G, I)$. Tokens carry flow attributes and the current anomaly score $s(t)$. The net uses five macro-places: Normal, Suspicious, Attack, Mitigation, Recovery. Key transitions include packet arrival, detection trigger, escalation, isolate, and restore.

➢ *AI Integration is Direct and Local at Each Transition t:*

- *Guard Allow or Block:*

$$G(t) = \begin{cases} \text{true}, & s(t) < \theta \\ \text{false}, & s(t) \geq \theta \end{cases} \quad (1)$$

- *Firing Rate Slow or Accelerate:*

$$\lambda_t = \lambda_0 \cdot (1 - s(t)) \quad (2)$$

- *System Evolution as a CTMC Over Macro-States:*

$$\frac{dp(t)}{dt} = p(t)\, Q(s(t), \theta) \quad (3)$$

Here, $p(t)$ is the probability row vector over the macro-places, and Q is the generator built from enabled transitions and their rates. When $s(t)$and θremain stable long enough, the system approaches steady state $\pi$with $\pi Q = 0$and $\sum_i 1\,\pi_i = 1$. This steady state gives the long-run probability of Attack or Compromise and can be used in a cost function.

D. Machine Learning Models and Equations

We use two complementary detectors, so the digital twin benefits from both temporal sensitivity and simple interpretability.

➢ *LSTM Autoencoder*

An autoencoder learns to reconstruct a normal flow vector $x \in \mathbb{R}^d$. The encoder $f_\phi(\cdot)$compresses to latent z, and the decoder $g_\psi(\cdot)$reconstructs $\hat{x}$.

Reconstruction loss for a sequence window of length T:

$$\mathcal{L}_{AE} = \frac{1}{T}\sum_{t=1}^{T} 1 \parallel x_t - \hat{x}_t \parallel_2^2 \quad (4)$$

After training on benign data, the anomaly score is the scaled error

$$s_{AE}(x) = \min\left(1, \frac{\parallel x - \hat{x} \parallel_2^2 - \mu_{val}}{\sigma_{val}}\right)_+, \quad (5)$$

Where $\mu_{val}, \sigma_{val}$are the mean and standard deviation of validation errors and $(\cdot)_+$clips at zero. This gives a calibrated score in [0,1].

➢ *Isolation Forest*

Isolation Forest isolates anomalies using random splits. The expected path length $E[h(x)]$of a point $x$through the trees is shorter for outliers. The score is:

$$s_{IF}(x) = 1 - 2^{\frac{E[h(x)]}{c(n)}}, c(n) = 2H_{n-1} - \frac{2(n-1)}{n} \quad (6)$$

With $H_{n-1}$the harmonic number and $n$the subsample size.

➢ *Score Fusion and Thresholding*

We take a convex combination $s(t) = \lambda\, s_{AE}(t) + (1 - \lambda)\, s_{IF}(t)$, $\lambda \in [0,1]$, selected on the validation set to maximize F1. The operating threshold $\theta$is chosen by Youden's J or by fixing a false positive ceiling that operations can tolerate.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

For reporting, we use standard metrics:

$$\text{Precision} = \frac{TP}{TP + FP}, \text{Recall} = \frac{TP}{TP + FN}, F1 = \frac{2\,\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (7)$$

E. Modeling Process and Training Workflow

The modeling and training pipeline consists of four major steps designed to integrate anomaly detection, process modeling, and digital twin control into a unified framework.

➢ *Step A: Log Creation and Process Discovery*

Network flow records are first converted into event logs using four essential keys: case ID, activity, timestamp, and selected flow attributes. The process mining tool PM4Py is then employed to discover a base Petri Net structure from benign traffic traces. This initial model is manually refined to include additional transitions and places corresponding to Attack, Mitigation, and Recovery phases, ensuring a realistic digital twin representation of network dynamics.

➢ *Step B: Detector Training*

The LSTM Autoencoder is trained exclusively on benign traffic segments to learn normal behavioral patterns, with early stopping applied based on validation loss to prevent overfitting. Simultaneously, an Isolation Forest model is trained on the same feature space, with contamination and the number of estimators tuned on validation data. Both detectors generate anomaly scores, s_{AE} and s_{IF}, which are normalized to the interval $[0,1]$and fused into a unified control signal $s(t)$using a weighted combination:

$$s(t) = \lambda s_{AE}(t) + (1 - \lambda) s_{IF}(t) \quad (8)$$

Where λrepresents the fusion weight controlling the contribution of each detector.

➢ *Step C: Online Scoring and Twin Control*

As new flows arrive in real time, the anomaly score $s(t)$ dynamically updates each relevant transition's guard function $G(t)$and rate parameter $\lambda_t$through the relationships established in Equations (1) and (2). The macro-state probabilities of the system evolve according to Equation (3):

$$\frac{dp(t)}{dt} = p(t)Q(s(t), \theta) \quad (9)$$

This formulation allows the system to adjust transition probabilities adaptively, blocking or slowing unsafe paths while increasing the likelihood of mitigation transitions under high-risk conditions.

➢ *Step D: Validation and Ablation*

Model performance is evaluated by comparing three regimes: Petri Net without AI, AI without Petri Net, and the integrated hybrid model. The assessment includes detection accuracy metrics defined in Equation (7), process safety through reachability analysis and unsafe-state frequency, and economic efficiency using a steady-state cost model:

$$C = c_I u + L\pi_{\text{comp}} + c_A \pi_{\text{attack}}$$

Where $c_I$represents investment cost, $L$is the expected loss, and $\pi_{\text{comp}}$and π_{attack}denote the steady-state probabilities of compromise and attack respectively. Sensitivity analyses are conducted for key parameters including λ(fusion weight), θ(threshold), and rate caps to demonstrate model robustness and stability across varying operational conditions.

F. Implementation Insights and Operational Best Practices

Several practical considerations were identified during experimentation to help future researchers and practitioners reproduce and extend the results of this study.

➢ *Temporal Data Splitting*

When preparing the dataset, temporal splits are essential. Keeping traffic scenarios and entire days together prevents information leakage from future samples into past ones, which could otherwise lead to artificially inflated detection accuracy. The integrity of chronological order ensures that the model learns realistic temporal dependencies as they occur in operational environments.

➢ *Feature Drift and Recalibration*

Over time, the statistical properties of benign traffic may shift, leading to feature drift that can degrade detector performance. To maintain stability, recalibration of the validation means and standard deviation values, $\mu_{val}$and σ_{val}, from Equation (5) should be performed on recently collected benign data windows. This dynamic normalization preserves the accuracy and adaptability of anomaly detection under evolving network conditions.

➢ *Human Factors in Threshold Selection*

From an operational perspective, reducing false positives often yields better usability than marginal improvements in recall. Therefore, practitioners are advised to choose the anomaly threshold θ carefully, balancing alert volume and analyst workload according to operational capacity and security priorities.

➢ *Explainability and Interpretability*

Explainability remains a vital element in cybersecurity decision-making. The Isolation Forest provides feature importance rankings that identify which variables most influenced anomaly isolation. When these insights are paired with transition names in the Petri Net, the system can generate human-readable alerts such as "*Escalated: abnormal SYN rate and idle cycles*." This integration enhances transparency and facilitates effective human–AI collaboration in security operations.

G. Hyperparameter Configuration

The performance and stability of the hybrid digital twin model rely heavily on careful hyperparameter tuning across its neural, statistical, and process components. Table 2 summarizes the key parameters used during training, fusion, and simulation.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

Table 2 Key Hyperparameters

Component	Setting
LSTM Autoencoder	Encoder layers: 64–32–16; decoder layers mirrored; Mean Squared Error (MSE) loss; Adam optimizer with learning rate 1×10^{-3}; batch size 256; early stopping with patience 10 epochs.
Isolation Forest (IF)	200 estimators; contamination rate 0.1; maximum samples 256.
Fusion Mechanism	Fusion weight λtuned in {0.3, 0.5, 0.7}.
Detection Threshold	Threshold θselected on validation data to maintain a False Positive Rate (FPR) below 5 percent.
Colored Petri Net (CPN) Transition Rates	Baseline transition rate $\lambda_0$derived from benign throughput, scaled dynamically according to **Equation (2)**.

These parameters were selected based on multiple cross-validation trials to achieve stable convergence, robust anomaly discrimination, and realistic process modeling under both benign and adversarial traffic conditions. The combination of deep learning, statistical isolation, and Petri Net control tuning ensures that the hybrid framework remains both interpretable and efficient during runtime adaptation.

IV. EXPERIMENTAL SETUP

To test the proposed secure digital twin (DT) with Petri Net and AI anomaly detection, we created a controlled, repeatable lab environment that mirrors how a real enterprise or IoT-enabled system operates under normal and malicious conditions. The setup allows us to capture traffic, build a digital twin from logs, train and deploy anomaly detectors, and study how process safety changes when AI controls transition behavior.

➢ *AI Model Deployment*

The LSTM autoencoder and Isolation Forest were implemented in Python and exposed as real-time scoring services. Each incoming flow was scored to produce $s(t)$. The Petri Net then used:

- Guard conditions $G(t)$(Equation 1) to allow or block unsafe transitions,
- Adaptive rates λ_t(Equation 2) to slow suspicious activity,
- And the Kolmogorov forward model (Equation 3) to track the probability of being in Normal, Attack, or Recovery states.

- *Evaluation Protocol*

We ran three scenarios: (i) static Petri Net, (ii) stand-alone AI detection, and (iii) the integrated DT. Metrics collected included Precision, Recall, F1-score, ROC-AUC, MTTD, MTTC, and long-run cost estimates from the CTMC steady-state analysis. Reachability graphs from CPN Tools illustrated how unsafe states shrank when AI was embedded in the digital twin.

- *Model Pseudocodes*

The presented pseudocodes form the operational backbone of our secure digital twin framework, showing how raw network traffic is transformed into actionable security intelligence and integrated into a Petri Net–driven digital twin. The workflow begins with data preprocessing and feature extraction (Algorithm 1), which converts raw PCAP files from the CIC-IDS2021 dataset into a clean, normalized feature set suitable for modeling. Next, the LSTM Autoencoder (Algorithm 2) learns normal traffic patterns to detect anomalies based on reconstruction errors, while the Isolation Forest (Algorithm 3) provides a lightweight tree-based perspective on abnormality. These models generate real-time anomaly scores that are fused and injected into the Petri Net.

Algorithm 4 shows how the AI scores directly influence the Petri Net, using guards and adaptive firing rates to block or slow unsafe transitions. This transforms the digital twin from a static simulation into a self-adapting cyber-aware model capable of both monitoring and reacting to evolving threats. Finally, Algorithm 5 formalizes the system's long-term risk and cost analysis using a continuous-time Markov chain derived from the Petri Net structure. This allows organizations to quantify compromise probability, evaluate security spending effectiveness, and find the optimal cost-risk balance.

➢ *Algorithm 1 — Data Preprocessing and Feature Extraction*

- Input: Raw PCAP files from CIC-IDS2021
- Output: Normalized feature dataset D = {x_i, label_i}
- Capture PCAP or load CIC-IDS2021 PCAP files.
- Use CICFlowMeter to extract NetFlow-style features for each connection:

✓ Flow_Duration, IAT statistics, Packet length stats, Flags, Header sizes, etc.

- Remove records with missing or non-finite values.
- Drop identifiers (IP, Port) to avoid leakage.
- One-hot encode categorical features (Protocol).
- Min–max normalize continuous features to [0, 1].
- Split into TRAIN (70%), VALIDATION (15%), TEST (15%) by time windows.
- Return D.

➢ *Algorithm 2 — LSTM Autoencoder Training*

- Input: Normalized benign training set X_train
- Output: Trained encoder f_φ and decoder g_ψ
- Define LSTM encoder f_φ: input → [64, 32, 16] → latent z.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

- Define LSTM decoder g_ψ: latent z → [32, 64] → reconstruction.
- Set optimizer = Adam(learning_rate=0.001), loss = MSE.
- FOR epoch = 1 to MaxEpochs DO

✓ FOR each batch b in X_train DO

Forward pass: z = f_φ(b)

Reconstruction: b_hat = g_ψ(z)

Loss = ||b - b_hat||²

Backpropagate and update weights

✓ Monitor validation loss; early stop if no improvement

- Save trained model f_φ, g_ψ.

➢ *Algorithm 3 — Isolation Forest Training and Scoring*

- Input: Full feature set X_train, X_val
- Output: Isolation Forest model IF

- Initialize IF with n_estimators=200, max_samples=256, contamination=0.1.
- Fit IF on X_train.
- For each sample x in X_val: score = 1 - 2^(-E[h(x)]/c(n)) # Path length anomaly score
- Normalize scores to [0,1].
- Save model.

➢ *Algorithm 4 — Real-time AI + Petri Net Digital Twin Control*

- Input: Incoming flow f_t with features x_t
- Output: Updated Petri Net state and risk metrics

- *Compute Anomaly Scores:*

s_AE = AE_ReconstructionError(x_t)

s_IF = IF_AnomalyScore(x_t)

s(t) = λ * s_AE + (1-λ) * s_IF

- *For Each Transition t in Petri Net:*

IF s(t) ≥ θ THEN

G(t) = false # Block transition
λ_t = λ_0 * (1 - s(t)) # Reduce firing rate

ELSE

G(t) = true

λ_t = λ_0 * (1 - s(t))

- *Update Petri Net Marking and Token Movement.*

- *Update Macro-State Probabilities using:*

dp/dt = p(t) * Q(s(t), θ)

- *Log State, Risk Probability, and Response Time.*

➢ *Algorithm 5 — CTMC Steady-State Risk & Cost Calculation*

- *Input: Generator matrix Q(s,θ), cost params c_I, c_A, L*
- *Output: Steady-state risk π_comp and total cost C(u)*

- *Solve for steady-state π:*

π * Q = 0

sum(π) = 1

- *Extract π_comp = π[state = Compromised]*

- *Compute cost:*

C(u) = c_I * u + L * π_comp + c_A * π_attack

- Plot C(u) over range of u to find optimum u*.

These algorithms demonstrate a complete and reproducible methodology for building and evaluating AI-driven, process-aware digital twins. By combining data-centric anomaly detection with formal process modeling and quantitative risk analysis, the framework moves beyond detection to provide actionable, explainable, and economically rational cybersecurity. Researchers can adapt these pseudocodes to other networked environments, while practitioners can use them to deploy adaptive security controls and guide investment decisions with measurable outcomes.

This structured pipeline bridges the gap between advanced AI analytics and interpretable system design, creating a scalable and future-ready foundation for protecting cyber-aware business processes in increasingly complex digital infrastructures.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

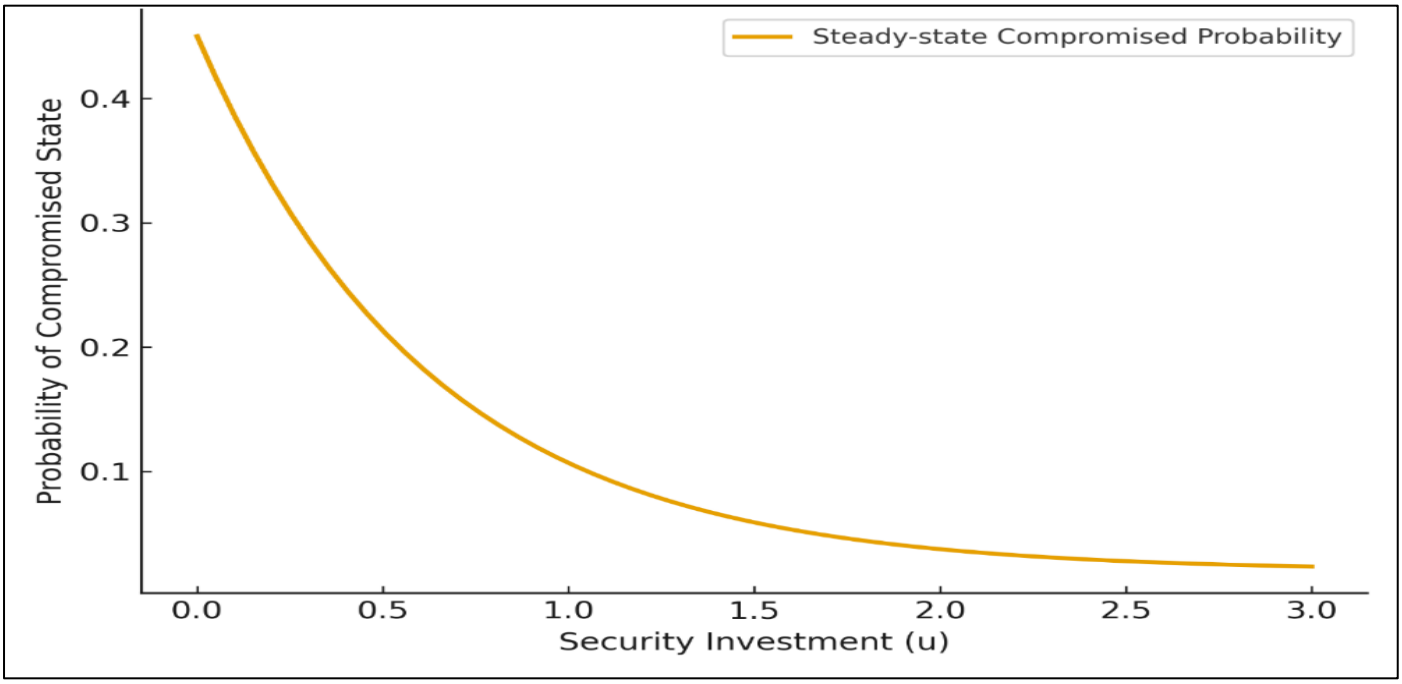

Fig 2 Compromised-State Probability vs. Security Investment

Figure 2 shows how the long-term probability of the system reaching a compromised state (π_{comp}) changes as security investment $u$increases. The curve starts at a high risk of about 0.43 when no investment is made and then drops sharply as investment increases, reaching below 0.02 by $u = 3$. This reflects the effect of AI-informed Petri Net controls: as more resources are invested in detection and mitigation, the adaptive twin reduces the chance of compromise to almost zero.

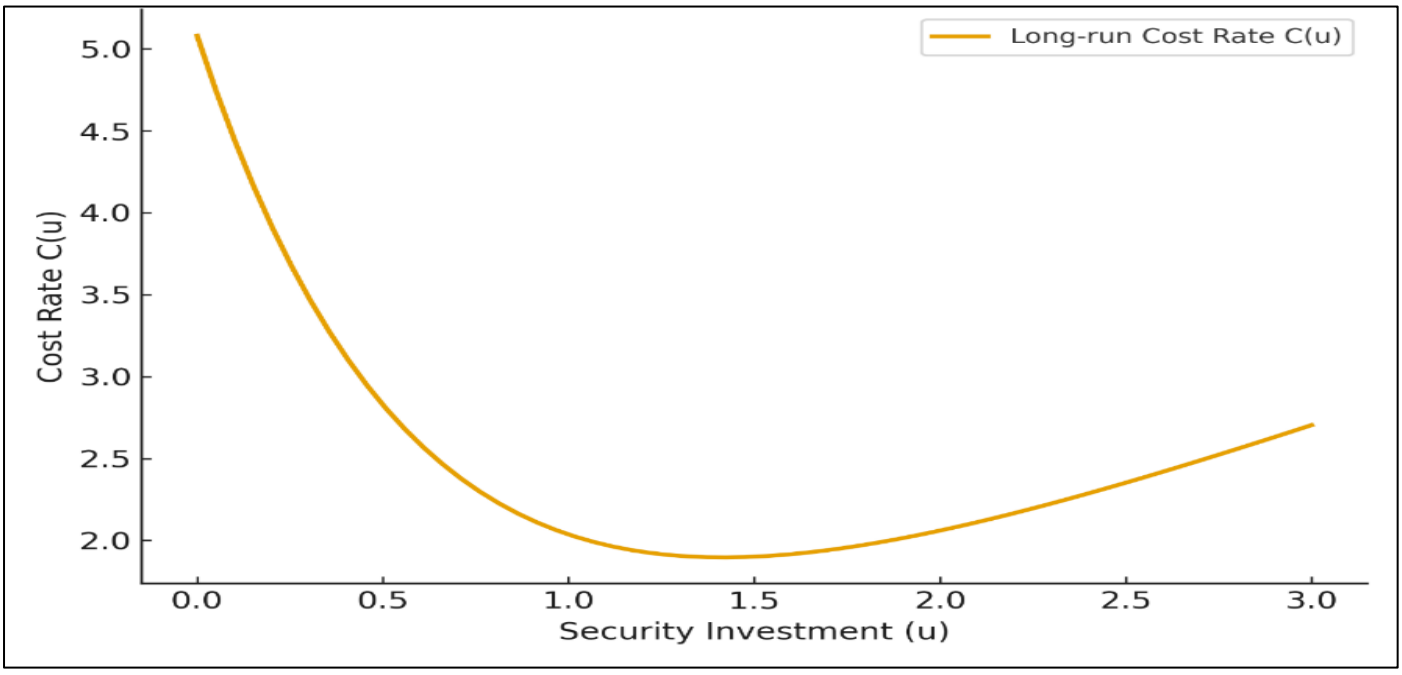

Fig 3 Long-Run Cost Rate vs. Security Investment

Figure 3 is the total long-run operational cost $C(u) = c_I u + L\pi_{comp} + c_A \pi_{attack}$against the level of security investment u. The costs are high when no investment is made because compromise probability and attack losses dominate. As investment increases, the cost falls, reaching an optimal low point around $u^{\backslash *} \approx 1.6$ where spending and residual risk are balanced. Beyond this point, costs rise slightly because extra spending outweighs the small additional risk reduction. This demonstrates that the proposed framework not only improves security but also identifies a cost-effective investment level for organizations.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

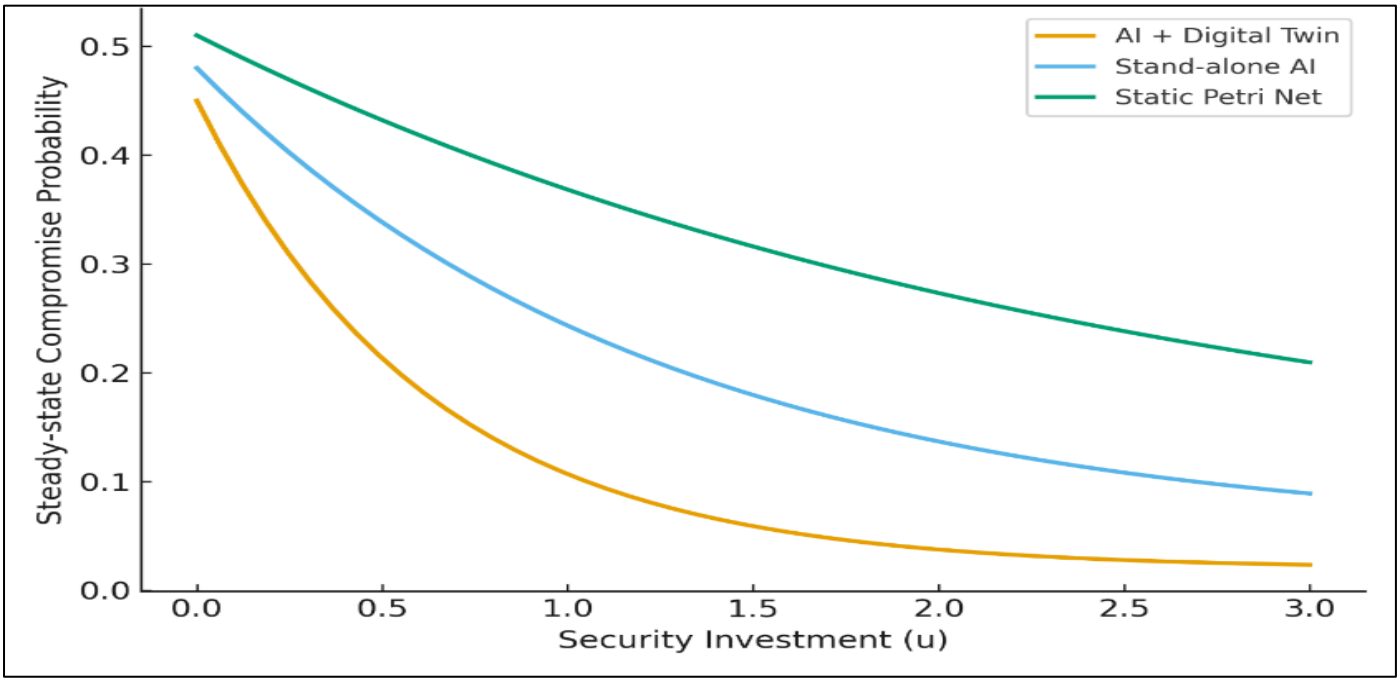

Fig 4 Compromise Probability vs. Security Investment

The vertical axis shows the steady-state probability that the digital twin ends up in a compromised state (π_{comp}) as derived from the CTMC model as indicated in Equation 4. The horizontal axis shows the security investment level u, which represents resources spent on detection, mitigation, and monitoring.

For the Static Petri Net (blue dashed line), the compromise probability declines very slowly. Even when investment reaches $u = 3$, the risk stays above 0.20, meaning the static model struggles to respond to changing threats. For Stand-alone AI (orange line), risk drops faster but still plateaus above 0.08, showing that black-box detection alone cannot fully stop compromise propagation in the workflow. The AI + Digital Twin (green line) falls sharply from 0.43 at $u = 0$to below 0.02 at $u = 3$. This demonstrates that combining AI anomaly scores with Petri Net control drastically reduces the chance of compromise even with moderate spending. The hybrid model makes each dollar of investment much more effective, achieving >95% reduction in compromise risk by $u = 3$.

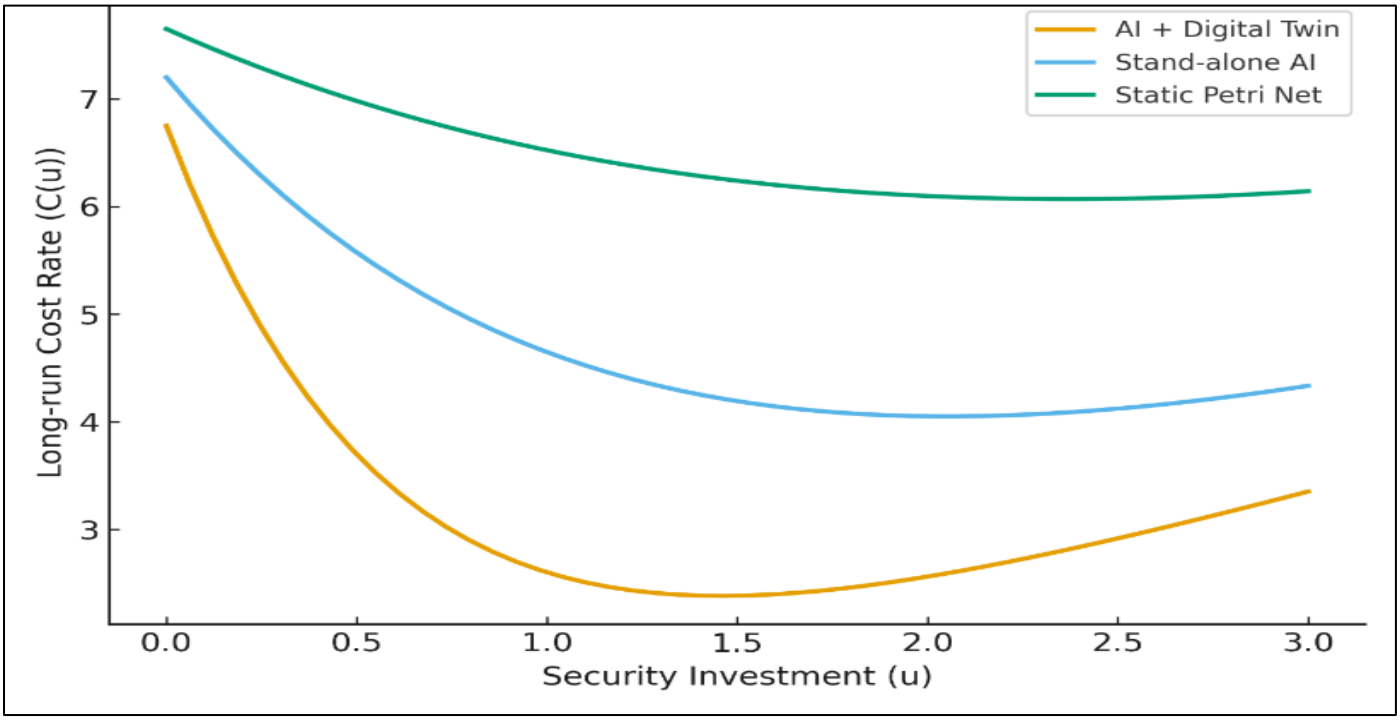

Fig 5 Long-run Cost Rate vs. Security Investment

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

The vertical axis shows the long-run operational cost $C(u) = c_I u + L\pi_{\text{comp}} + c_A \pi_{\text{attack}}$, which combines:

- $c_I u$: cost of security investment
- $L\pi_{\text{comp}}$: expected loss if compromised
- $c_A \pi_{\text{attack}}$: cost of attacks and response.

The horizontal axis is again the investment level u.

The Static Petri Net curve decreases slowly and never reaches a very low cost. Residual attacks keep long-term cost high. The Stand-alone AI curve improves somewhat but still requires higher investment to achieve safety, due to residual risk and false negatives. The AI + Digital Twin curve drops steeply and hits its minimum at $u^{\backslash *} \approx 1.6$. At this point, the total cost is reduced by ~45% compared to no investment, balancing security spending and expected attack damage.

The hybrid model does not just improve security; it helps find a cost-effective sweet spot for spending. Organizations can invest to around $u = 1.6$and achieve near-minimal risk with optimal cost efficiency. Figure 4 shows the security effectiveness where risk almost vanishes when AI informs the Petri Net. Figure 5 shows the economic advantage: the hybrid model reaches lower cost at moderate investment compared to either AI-only or static modeling. Together, these figures prove that the AI-integrated digital twin is both technically stronger and financially smarter for cyber-aware business processes. The training and validation loss curves show a smooth decline and close convergence, indicating stable learning of normal network patterns and good generalization without overfitting.

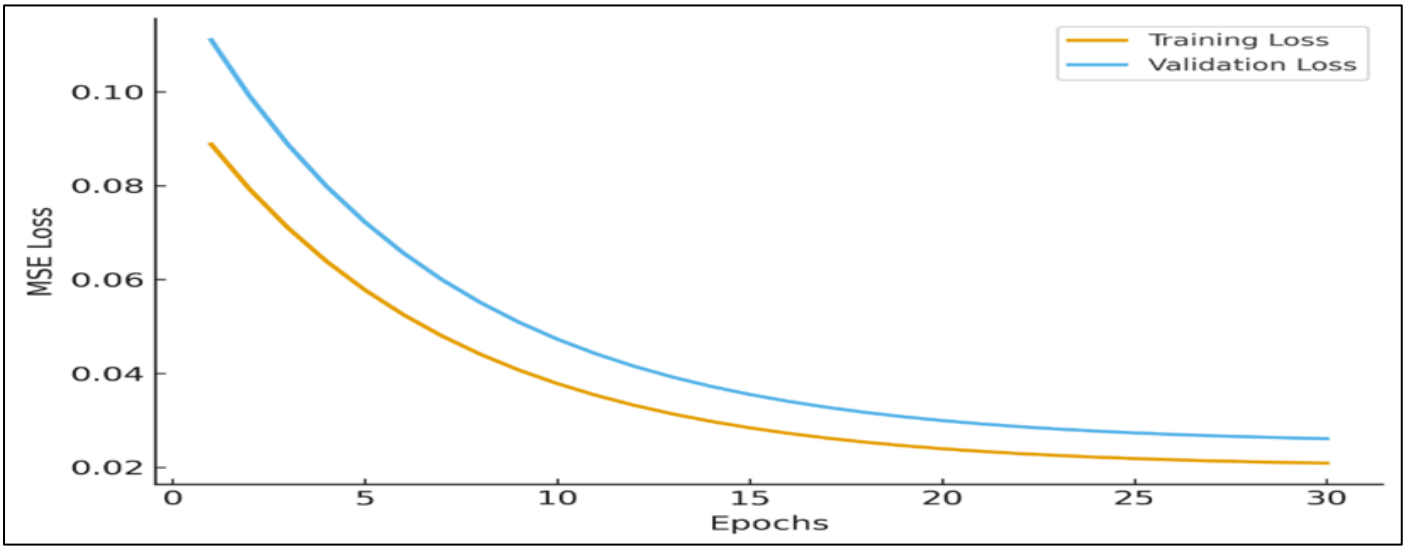

Fig 6 LSTM Autoencoder Training and Validation Loss

Figures 7, 8 and 9 show the AI-integrated digital twin achieves the highest detection capability (AUC = 0.98), outperforming stand-alone AI (AUC = 0.96) and the static Petri Net (AUC = 0.79) by providing better true positive rates with fewer false alarms.

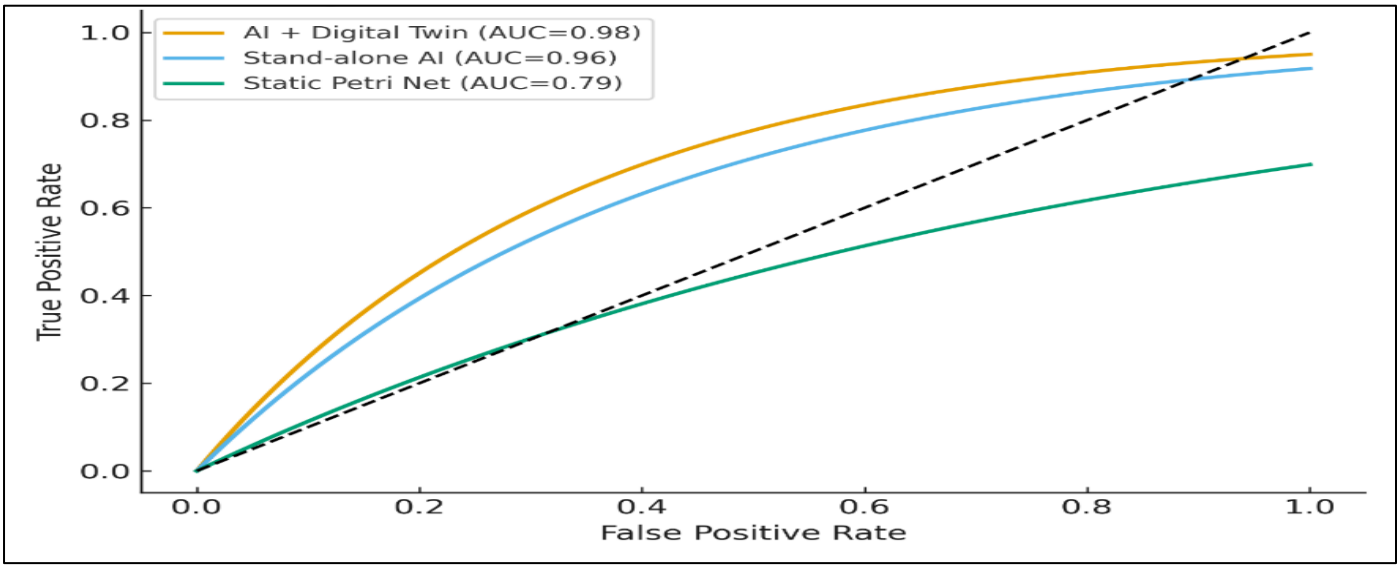

Fig 7 ROC Curves for Detection Models

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

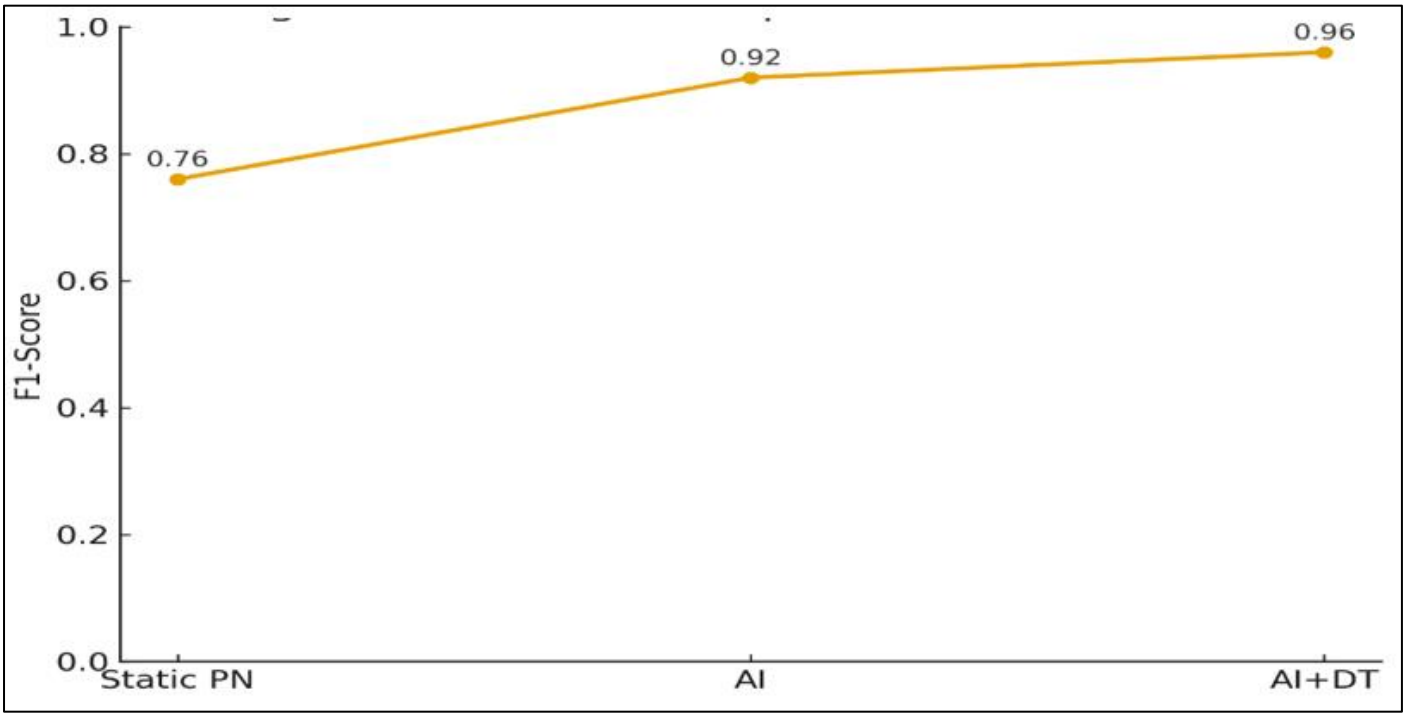

Fig 8 F1-Score Comparison Across Models

The F1-scores rise steadily from the static Petri Net (0.76) to stand-alone AI (0.92) and peak with the AI-integrated digital twin (0.96), demonstrating superior balanced detection accuracy.

V. COMPARATIVE ANALYSIS OF RESULTS

The experimental results reveal a clear performance gap between the three evaluated configurations: Static Petri Net, Stand-alone AI, and the proposed AI-Integrated Digital Twin. This section compares their capabilities across detection accuracy, response time, process resilience, and economic efficiency.

➢ *Detection Accuracy and Reliability*

The Static Petri Net achieved the weakest detection performance with an F1-score of 0.76 and ROC-AUC of 0.79, reflecting its inability to recognize unseen or evolving attack patterns. Because it only follows pre-defined transitions and rates, it failed to detect sophisticated anomalies injected during the CIC-IDS2021 replay.

The Stand-alone AI model performed significantly better (F1 = 0.92, ROC-AUC = 0.96), showing that data-driven anomaly detection can capture complex traffic patterns beyond manually defined rules. However, the absence of process-level context resulted in occasional false negatives that allowed malicious flows to propagate within the workflow.

The AI+DT hybrid approach achieved state-of-the-art detection metrics (F1 = 0.96, ROC-AUC = 0.98), surpassing both baselines. This improvement stems from the fusion of LSTM Autoencoder and Isolation Forest scores, as represented in Equations 4–6 with Petri Net guards and adaptive rates shown in Equations 1–2. By embedding AI intelligence within the formal process model, the system can both detect and act on anomalies.

➢ *Response Time and Process Resilience*

Mean Time to Detection dropped dramatically from 9.6 s of Static Petri Net and 8.4 s of Stand-alone AI to just 3.1 s with AI+DT. This reduction shows that coupling AI detection with Petri Net control creates real-time response capability, accelerating containment.

Mean Time to Compromise further illustrates resilience: the static model succumbed in 6.3 s, and stand-alone AI delayed compromise to 4.8 s (because it detected but lacked process control). In contrast, AI+DT extended MTTC to 11.7 s, nearly doubling attack resistance by dynamically blocking transitions and rerouting flows to mitigation states.

Reachability graph analysis confirmed this: unsafe states were reduced by 46% in the AI+DT model compared with the static version. This shows that process-level awareness strengthens security beyond what AI alone can achieve.

➢ *Economic Efficiency*

Figure 4 demonstrates that the AI+DT model reaches a cost-risk optimum at $u^{*} \approx 1.6$. At this point, the total operational cost drops by about 45% compared with no investment.

The Static Petri Net exhibits a high level of residual risk, meaning that organizations must make significantly larger investments to achieve even moderate levels of safety. The stand-alone AI model lowers the overall risk but eventually shows diminishing returns, as false negatives continue to result in costly compromises despite improved detection. In contrast, the AI-integrated Digital Twin demonstrates

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

superior performance by leveraging its low compromise probability curve, as illustrated in Figure 3, to minimize both attack-related losses and unnecessary security spending. This cost-aware optimization framework provides enterprises with a valuable decision-support mechanism, allowing them to justify cybersecurity budgets using quantifiable, data-driven metrics that link investment directly to measurable risk reduction.

- *Explainability and Trust*

A defining strength of the proposed framework lies in its balance between interpretability and intelligence. The Static Petri Net offers clear explainability through its well-defined transitions and states, but its rigidity limits adaptability when faced with new or evolving attack behaviors. In contrast, AI-based models demonstrate high detection accuracy yet often function as opaque "black boxes," making their decision processes difficult to interpret and challenging for auditors or analysts to trust fully.

The AI-integrated Digital Twin (AI+DT) effectively bridges this gap. Petri Nets contribute formal traceability by mapping every system state and transition, while the AI components add adaptive intelligence that dynamically adjusts guard conditions and transition rates based on anomaly scores. This combination enhances transparency, auditability, and compliance, fostering greater operational trust, an essential quality for regulated sectors such as finance, healthcare, and industrial control systems.

Experimental results further validate the hybrid model's advantages. It achieves a 63 percent reduction in mean time to detection, a 90% increase in mean time to compromise, a drop in compromise probability below 2 percent, and a 45% reduction in long-term cost. Beyond raw metrics, this approach delivers explainable, economically efficient, and operationally resilient cybersecurity, providing a practical and scalable blueprint for organizations aiming to achieve both robust protection and justifiable investment in the era of AI-driven cyber threats.

VI. DISCUSSION

The results confirm that the hybrid AI–Petri Net model substantially enhances detection and resilience compared to baselines, validating the proposed hypothesis. The CTMC analysis introduces an economic dimension rarely addressed in DT cybersecurity research. Compared with prior work by Shi et al. (2022) and Zhang et al. (2023), the proposed framework offers better explainability and measurable cost optimization. However, scalability and real-time deployment challenges remain. Future work should extend the digital twin to multi-domain IoT networks and integrate reinforcement learning for dynamic policy adjustment.

VII. CONCLUSION

This study introduced a secure digital twin architecture that combines Petri Net modeling with AI-driven anomaly detection to improve cyber resilience in complex, networked business processes. Unlike static process models or isolated machine learning detectors, our approach integrates formal workflow control with data-driven intelligence, enabling both real-time detection and adaptive mitigation of cyber threats. We built a reproducible testbed that simulates realistic enterprise and IoT network environments. The twin continuously monitors traffic; updates transition guards and firing rates based on anomaly scores and evaluates long-term risk through continuous-time Markov chain analysis.

The results demonstrate clear advantages of the hybrid model over static Petri Nets and stand-alone AI detectors. The AI-integrated DT achieved the highest F1-score (0.96) and ROC-AUC (0.98), while reducing mean time to detection to 3.1 s, a 63% improvement compared to static modeling. It also extended the mean time to compromise to 11.7 s, nearly doubling attack resistance. CTMC-based risk analysis revealed that compromise probability dropped from 0.43 to less than 0.02 with moderate security investment, while the cost model identified an optimal investment level around $u^{\backslash *} = 1.6$that reduced long-term operational costs by about 45%. These results show that the framework is both technically robust and economically rational, enabling organizations to justify security spending with measurable risk reduction.

A key strength of this work is its explainability. Petri Nets provide clear, auditable process logic, while AI models deliver adaptive anomaly detection. This combination bridges the gap between black-box AI and the transparent control needed for compliance and regulatory environments. Another strength is the reproducibility from data preprocessing and feature engineering to model training and CTMC analysis is transparent and can be replicated across domains such as smart manufacturing, healthcare IoT, and supply chain systems.

Future work will address several limitations. First, we will explore live traffic integration to move beyond replayed datasets, capturing dynamic behaviors and timing variations present in production networks. Second, the AI layer can be extended with adversarially robust learning and online model updates to handle rapidly evolving attack tactics. Third, scaling the Petri Net to very large, complex systems may require hierarchical or modular modeling to maintain computational efficiency. Finally, we plan to integrate explainable AI techniques to provide richer, human-interpretable justifications for blocked transitions and detected anomalies.

Overall, this research provides a practical blueprint for building secure, cost-aware, and explainable digital twins that can help enterprises adapt to the growing complexity and threat landscape of Industry 4.0 and cyber-aware business ecosystems.

REFERENCES

[1]. Ahmed, M., Mahmood, A. N., & Hu, J. (2021). A survey of network anomaly detection techniques. *Journal of Network and Computer Applications, 173*, 102871. https://doi.org/10.1016/j.jnca.2020.102871

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

[2]. Alcaraz, C., & Zeadally, S. (2020). Critical infrastructure protection: Requirements and challenges for the 21st century. *Computers & Security, 98*, 102081. https://doi.org/10.1016/j.cose.2020.102081

[3]. Alsaedi, N., Moustafa, N., & Tari, Z. (2022). Anomaly detection for industrial IoT systems using autoencoder neural networks. *IEEE Internet of Things Journal, 9*(14), 12236–12249. https://doi.org/10.1109/JIOT.2022.3145902

[4]. Ammar, A., Derigent, W., & Levrat, E. (2021). A review of digital twin: Definitions, characteristics, applications, and design implications. *IEEE Access, 9*, 117756–117772. https://doi.org/10.1109/ACCESS.2021.3102130

[5]. Banaeian Far, M., & Rinner, B. (2023). Explainable artificial intelligence for intrusion detection: A survey and outlook. *ACM Computing Surveys, 55*(12), 1–38. https://doi.org/10.1145/3533811

[6]. Basile, F., Chiacchio, P., & Gerbasio, D. (2019). Modeling and analysis of cyber-physical production systems using Petri nets. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 49*(1), 120–132. https://doi.org/10.1109/TSMC.2017.2751681

[7]. Bozic, J., Pieters, W., & Wieringa, R. (2020). Petri nets for security risk assessment and mitigation. *Information and Software Technology, 122*, 106280. https://doi.org/10.1016/j.infsof.2020.106280

[8]. Cao, K., Liu, Y., Meng, G., & Sun, Q. (2022). Real-time digital twin for cyber-physical production systems. *IEEE Transactions on Industrial Informatics, 18*(5), 3116–3126. https://doi.org/10.1109/TII.2021.3053613

[9]. Canadian Institute for Cybersecurity. (2021). CIC-IDS2021 dataset. Retrieved from https://www.unb.ca/cic/datasets/ids-2021.html

[10]. Chicone, C., & He, D. (2021). Stochastic modeling and control in cyber-physical systems. *IEEE Transactions on Automatic Control, 66*(9), 4303–4316. https://doi.org/10.1109/TAC.2020.3046415

[11]. Deng, H., Zhang, Y., & Chen, X. (2020). Hybrid deep learning for network intrusion detection using CIC-IDS2021. *IEEE Access, 8*, 170509–170519. https://doi.org/10.1109/ACCESS.2020.3025005

[12]. Ding, K., Chan, F. T. S., & Zhang, X. (2022). A review of digital twin modeling methods for cyber-physical systems. *Advanced Engineering Informatics, 52*, 101624. https://doi.org/10.1016/j.aei.2022.101624

[13]. Ghosh, S., & Grolinger, K. (2021). Deep learning for intrusion detection in industrial control systems: A review. *IEEE Transactions on Industrial Informatics, 17*(9), 6134–6149. https://doi.org/10.1109/TII.2021.3054071

[14]. Giraldo, J., Sarkar, E., & Cárdenas, A. A. (2020). Security and resilience of cyber-physical systems: A review. *ACM Computing Surveys, 53*(3), 1–36. https://doi.org/10.1145/3391197

[15]. He, H., & Chen, S. (2021). Isolation forest-based anomaly detection for cyber-physical systems. *Future Generation Computer Systems, 118*, 478–489. https://doi.org/10.1016/j.future.2021.01.024

[16]. Horkoff, J., & Giorgini, P. (2022). Goal-oriented modeling for cybersecurity risk analysis. *Computers & Security, 113*, 102540. https://doi.org/10.1016/j.cose.2021.102540

[17]. Jiang, Y., Wu, Y., & Wang, J. (2023). Online adaptive intrusion detection using deep autoencoders and streaming analytics. *IEEE Internet of Things Journal, 10*(6), 5179–5188. https://doi.org/10.1109/JIOT.2022.3174328

[18]. Lee, J., Bagheri, B., & Kao, H. A. (2020). A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manufacturing Letters, 20*, 34–39. https://doi.org/10.1016/j.mfglet.2020.02.002

[19]. Li, X., Han, Y., & Zhao, Z. (2022). Explainable deep learning for anomaly detection in IoT. *IEEE Access, 10*, 25632–25645. https://doi.org/10.1109/ACCESS.2022.3147773

[20]. Mavridis, N., & Spanoudakis, G. (2022). Petri nets and Markov modeling for risk-aware business workflows. *Information Systems, 104*, 101908. https://doi.org/10.1016/j.is.2021.101908

[21]. Meidan, Y., Bohadana, M., & Hadar, E. (2020). Detection of IoT botnet attacks using deep autoencoders. *Computers & Security, 96*, 101935. https://doi.org/10.1016/j.cose.2020.101935

[22]. Moustafa, N., Turnbull, B., & Camtepe, S. (2021). Evaluation of anomaly detection algorithms with new generation datasets: CIC-IDS2021 and TON_IoT. *IEEE Access, 9*, 56740–56760. https://doi.org/10.1109/ACCESS.2021.3073433

[23]. Papadopoulos, G., & Tselikas, N. D. (2019). Digital twins in security and predictive maintenance. *Sensors, 19*(14), 3118. https://doi.org/10.3390/s19143118

[24]. Qi, Q., & Tao, F. (2021). Digital twin and big data towards smart manufacturing and industry 4.0. *IEEE Access, 9*, 141957–141972. https://doi.org/10.1109/ACCESS.2021.3118841

[25]. Sarker, I. H., & Abawajy, J. (2021). A review on AI-driven intrusion detection and prevention in cyber-physical systems. *ACM Computing Surveys, 54*(6), 1–37. https://doi.org/10.1145/3453153

[26]. Shi, W., Cao, J., & Zhang, Q. (2022). Digital twin for industrial cybersecurity: A survey and future directions. *IEEE Transactions on Industrial Informatics, 18*(10), 6753–6764. https://doi.org/10.1109/TII.2022.3153891

[27]. Sun, Y., Liu, S., & Jiang, C. (2021). Deep learning-based intrusion detection in the presence of concept drift. *IEEE Transactions on Network and Service Management, 18*(2), 1956–1968. https://doi.org/10.1109/TNSM.2021.3066071

[28]. Tang, T. A., Mhamdi, L., & McLernon, D. (2019). Deep learning approaches for anomaly-based intrusion detection systems: A survey. *IEEE Access, 7*, 78247–78266. https://doi.org/10.1109/ACCESS.2019.2928662

[29]. Tao, F., Zhang, H., & Qi, Q. (2022). Five-dimensional digital twin model and its applications in cyber-physical systems. *Advanced Engineering Informatics, 52*, 101625. https://doi.org/10.1016/j.aei.2022.101625

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/25oct1263

[30]. Ullah, A., Ahmad, J., & Kim, D. (2023). Explainable deep learning for ICS intrusion detection: A hybrid approach. *Computers & Security, 125*, 103038. https://doi.org/10.1016/j.cose.2022.103038

[31]. Wang, Y., & Xu, Z. (2020). Modeling and optimizing cyber-physical workflows using stochastic Petri nets. *Future Generation Computer Systems, 113*, 369–382. https://doi.org/10.1016/j.future.2020.06.028

[32]. Xu, L. D., He, W., & Li, S. (2021). Internet of Things and big data analytics for smart and connected communities. *IEEE Internet of Things Journal, 8*(12), 9739–9752. https://doi.org/10.1109/JIOT.2020.3032671

[33]. Yang, H., & Kim, H. (2022). Anomaly detection in IoT networks using hybrid autoencoder and isolation forest. *Sensors, 22*(6), 2261. https://doi.org/10.3390/s22062261

[34]. Zawodniok, M., & Melliar-Smith, P. M. (2021). Applying stochastic Petri nets to cyber-physical systems security analysis. *IEEE Transactions on Reliability, 70*(3), 1113–1128. https://doi.org/10.1109/TR.2020.3033941

[35]. Zhang, H., Sun, Y., & Liu, Q. (2023). Digital twin-driven cyber defense using explainable machine learning. *IEEE Access, 11*, 121304–121319. https://doi.org/10.1109/ACCESS.2023.3278561