Scientific Research Publishing

International Journal of Communications, Network and System Sciences, 2026, 19(2), 13-38
https://www.scirp.org/journal/ijcns
ISSN Online: 1913-3723
ISSN Print: 1913-3715

AI-Assisted Cybersecurity Mesh for Threat Detection in Edge-Enabled Communication Networks

Utham Kumar Anugula Sethupathy*, **Vijayanand Ananthanarayanan**

Independent Researcher, Atlanta, GA, USA
Email: *mailuthamkumar@gmail.com, vijayanand31@gmail.com

How to cite this paper: Sethupathy, U.K.A. and Ananthanarayanan, V. (2026) AI-Assisted Cybersecurity Mesh for Threat Detection in Edge-Enabled Communication Networks. *International Journal of Communications, Network and System Sciences*, **19**, 13-38.
https://doi.org/10.4236/ijcns.2026.192002

Received: January 21, 2026
Accepted: February 25, 2026
Published: February 28, 2026

Abstract

Next-generation communication environments increasingly combine IoT devices, edge gateways, cyber-physical components, and programmable network services. This convergence improves responsiveness but also creates fragmented trust boundaries and fast-changing attack surfaces. Conventional intrusion detection systems remain limited in such settings because they depend heavily on static signatures, centralized telemetry collection, or offline machine-learning models. This paper introduces a GenAI-assisted cybersecurity mesh for threat detection in heterogeneous intelligent communication systems. The framework places lightweight security functions at edge nodes, coordinates them through a mesh control layer, and uses a generative threat modeling engine to update anomaly assumptions as traffic conditions change. Network, application, and behavioral signals are fused into a dynamic risk score that supports policy actions such as throttling, isolation, and micro-segmentation. The framework is evaluated in a simulated communication environment with mixed benign traffic and attack scenarios, including DDoS, man-in-the-middle, protocol exploitation, behavioral drift, and synthetic zero-day patterns. Results show higher detection accuracy, lower false positive rates, and reduced response latency compared with rule-based and centralized ML-based IDS baselines. The study positions cybersecurity mesh as a practical direction for low-latency, AI-assisted protection of distributed communication infrastructures.

Keywords

Cybersecurity Mesh, Generative AI, Intelligent Communication Systems, Zero-Trust Architecture, Intrusion Detection, Cross-Layer Risk Scoring,

Distributed Security

1. Introduction

Intelligent communication systems are no longer built around a small number of controlled network endpoints. Modern deployments combine IoT sensors, edge computing nodes, cyber-physical controllers, software-defined networking functions, and emerging 5G/6G communication services. These components support smart transportation, industrial automation, healthcare monitoring, and distributed energy systems, but they also weaken the assumption that security can be enforced from a single central point. As device density, protocol diversity, and service mobility increase, the security model must shift from perimeter monitoring to continuous, context-aware protection across the communication fabric.

Traditional IDS deployments are useful for known threats but remain poorly suited for dynamic communication environments. Signature-driven tools detect recognized patterns but miss polymorphic and zero-day behavior. Centralized ML-based IDS models improve classification but introduce their own constraints: telemetry must be transported to a central point, inference may be delayed, and a compromised monitoring node can become a systemic weakness. A further limitation is analytical narrowness. Many IDS pipelines focus mainly on packet or flow-level anomalies while underusing application behavior, device telemetry, and protocol-state deviations that may reveal early-stage compromise.

The increasing sophistication of adversaries necessitates a transition toward adaptive, distributed, and intelligence-augmented security architectures. Generative Artificial Intelligence (GenAI) has recently demonstrated capabilities in pattern synthesis, contextual reasoning, and dynamic model adaptation. However, its integration into distributed cybersecurity infrastructures for real-time communication systems remains insufficiently explored. Existing research often treats AI-based intrusion detection as a standalone classifier rather than embedding adaptive intelligence within a mesh-based security topology.

To address these limitations, this paper proposes a GenAI-driven adaptive cybersecurity mesh architecture tailored for intelligent communication systems. The proposed framework incorporates three primary design principles:

1) Zero-Trust Distributed Security Enforcement: Security controls are enforced at edge nodes rather than relying solely on centralized monitoring.

2) Adaptive Generative Threat Modeling: A GenAI engine dynamically synthesizes threat hypotheses and refines anomaly detection models.

3) Cross-Layer Risk Fusion: Security signals from network, application, and behavioral layers are fused into a unified threat confidence score.

The contributions of this paper are summarized as follows:

- It proposes an edge-coordinated cybersecurity mesh for intelligent communication systems, where detection and enforcement are distributed across local

security nodes rather than concentrated in a central IDS.

- It introduces a GenAI-assisted threat modeling component that generates contextual attack hypotheses and supports model recalibration under changing traffic conditions.
- It defines a multi-signal risk scoring method that combines network, application, behavioral, and contextual threat indicators into a unified confidence score.
- It evaluates the framework against signature-based and centralized ML-based IDS baselines using multi-vector attack scenarios and scalability analysis.

The remainder of the paper is organized as follows. Section 2 reviews related work in distributed intrusion detection and AI-assisted security. Section 3 defines the threat model and adversarial assumptions. Section 4 describes the proposed cybersecurity mesh architecture. Section 5 presents the GenAI-based adaptive methodology and risk scoring formulation. Section 6 details the experimental setup, followed by results and analysis in Section 7. Section 8 discusses implications and limitations, and Section 9 concludes the paper.

2. Related Work

Security in intelligent communication systems has evolved significantly over the past decade, driven by the proliferation of IoT devices, edge computing frameworks, and next-generation communication protocols. Prior work has examined IoT intrusion detection architectures ranging from centralized and on-device models [1] to benchmark-driven IoT/IIoT evaluations [2] [3]. Zero-trust foundations and later survey work provide the basis for continuous verification and least-privilege enforcement [4] [5]. Cybersecurity mesh research has extended this discussion toward decentralized security control, cryptographic coordination, and AI-assisted defense [6]. Federated and distributed IDS studies have examined model coordination and collaborative enforcement across heterogeneous nodes [7]-[11]. Recent GenAI and LLM-security work motivates synthetic threat reasoning and automated security-context interpretation [12]-[14]. Dataset-focused IDS studies also highlight the importance of reproducible traffic characterization and benchmark design [15]-[18]. Existing research can therefore be grouped into four domains: 1) signature-based intrusion detection systems, 2) machine learning-based IDS models, 3) distributed and mesh-oriented security architectures, and 4) AI-assisted adaptive threat modeling. Existing research can be broadly categorized into four domains: 1) signature-based intrusion detection systems, 2) machine learning-based IDS models, 3) distributed and mesh-oriented security architectures, and 4) AI-assisted adaptive threat modeling.

2.1. Signature-Based and Centralized Intrusion Detection

Traditional intrusion detection systems such as Snort and Suricata rely on rule-based or signature-based detection mechanisms. While effective against known attack patterns, these systems exhibit limited capability in identifying zero-day ex-

ploits and polymorphic threats. Moreover, centralized deployment architecture introduces scalability and latency challenges in intelligent communication systems where nodes are geographically distributed and operate with real-time constraints.

Centralized IDS models also present a single point of failure and increase network overhead due to continuous telemetry aggregation. In high-throughput environments such as IoT-enabled smart grids or vehicular communication systems, this architectural limitation significantly affects detection latency and responsiveness.

Limitation Identified: Lack of adaptability and insufficient resilience against evolving, multi-stage attacks.

2.2. Machine Learning-Based Intrusion Detection

Recent advancements have incorporated supervised and unsupervised machine learning techniques for anomaly detection in communication networks. Approaches leveraging Support Vector Machines (SVM), Random Forests, k-Nearest Neighbors, and Deep Neural Networks have demonstrated improved detection accuracy compared to signature-based systems.

Deep learning models, including Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks, have been applied for traffic pattern recognition and time-series anomaly detection. These models enhance classification performance; however, they are typically trained offline and lack dynamic threat adaptation capabilities.

Furthermore, most ML-based IDS implementations operate in centralized environments. Distributed deployment remains limited due to computational overhead and synchronization challenges across heterogeneous nodes.

Limitation Identified: Static model training, insufficient cross-layer contextual fusion, and limited distributed enforcement.

2.3. Distributed and Mesh-Based Security Architectures

To overcome centralization constraints, recent research has explored distributed intrusion detection frameworks and cybersecurity mesh architectures. Cybersecurity mesh models emphasize decentralized policy enforcement, micro-segmentation, and zero-trust principles. Security capabilities are deployed closer to assets, often at edge gateways or node clusters.

Such architecture enhances resilience and reduces response latency. However, many implementations still rely on static rule engines or traditional ML classifiers without adaptive threat generation mechanisms. Additionally, risk scoring is often performed independently at the node level without systematic cross-layer correlation.

Limitation Identified: Absence of adaptive intelligence capable of synthesizing new attack hypotheses and limited risk fusion mechanisms.

2.4. Generative AI in Cybersecurity

Generative Artificial Intelligence (GenAI) has recently been explored in cyberse-

curity contexts for synthetic attack generation, adversarial training, automated threat intelligence summarization, and log analysis. Large language models and generative adversarial networks (GANs) have demonstrated potential in simulating evolving attack patterns and improving detection robustness through adversarial learning.

Despite promising advancements, integration of GenAI into real-time distributed communication security remains limited. Existing work often focuses on either:

- Offline threat simulation
- Isolated anomaly detection enhancement
- Security operations automation

There remains a gap in embedding GenAI as an adaptive reasoning component within a distributed cybersecurity mesh capable of continuous cross-layer threat modeling.

2.5. Research Gap

Existing IDS research has improved detection accuracy, but three gaps remain important for intelligent communication systems. First, many models still assume centralized collection and inference, which is not ideal for latency-sensitive edge environments. Second, AI-based IDS methods often operate as fixed classifiers rather than continuously updating their threat assumptions as adversarial behavior changes. Third, available approaches rarely combine network events, application behavior, node telemetry, and external threat context into a single operational risk score. These gaps motivate a mesh-based approach in which local detection, AI-assisted threat reasoning, and policy enforcement operate as coordinated functions.

3. Threat Model and Assumptions

Intelligent communication systems consist of heterogeneous devices, edge gateways, communication protocols, and cloud-coordinated services. These systems operate in semi-trusted or untrusted environments where adversaries may exploit protocol weaknesses, compromised nodes, or misconfigured services. This section formalizes the adversarial model and system assumptions used to evaluate the proposed GenAI-driven cybersecurity mesh.

3.1. System Model

We consider an intelligent communication environment comprising:

- N distributed nodes (IoT devices, embedded systems, edge servers)
- Edge gateways responsible for traffic aggregation
- A centralized orchestration layer for policy coordination
- Multi-layer communication stack:
  - Network layer (packet flows, routing behavior)
  - Transport/session layer (connection states, protocol exchanges)

- Application layer (API calls, service requests, payload semantics)
- Behavioral layer (node-level telemetry, CPU/memory usage, process anomalies)

Each node is equipped with a lightweight security agent capable of telemetry collection and local anomaly pre-processing. These agents communicate with a distributed mesh controller enforcing zero-trust policies.

3.2. Adversarial Capabilities

We assume a probabilistic polynomial-time (PPT) adversary with the following capabilities:

1) Network Manipulation

- Packet injection
- Replay attacks
- Distributed Denial of Service (DDoS)
- Man-in-the-Middle (MITM)

2) Protocol Exploitation

- Exploiting insecure handshake sequences
- Session hijacking
- Routing manipulation

3) Node Compromise

- Malware injection into edge devices
- Privilege escalation
- Lateral movement within sub-networks

4) Zero-Day Attack Simulation

- Novel attack patterns not previously observed
- Polymorphic traffic behavior

The adversary does not possess global cryptographic key material but may compromise a subset k of nodes where $k < N$.

3.3. Security Objectives

The proposed system aims to satisfy the following objectives:

1) Real-Time Threat Detection

- Minimize detection latency T_d
- Maintain bounded response time T_r

2) High Detection Accuracy

- Maximize True Positive Rate (TPR)
- Minimize False Positive Rate (FPR)

3) Adaptive Threat Modeling

- Continuous update of anomaly hypotheses
- Dynamic risk threshold recalibration

4) Distributed Resilience

- Avoid single point of failure
- Maintain detection capability under partial node compromise

3.4. Threat Categories Considered

The experimental evaluation considers five primary attack categories:

1) Distributed Denial of Service (DDoS)

- High-volume traffic floods targeting gateways

2) Man-in-the-Middle (MITM)

- Traffic interception and modification

3) Protocol Exploitation

- Malformed packet injection
- Session hijacking

4) Anomalous Behavioral Drift

- Gradual deviation in device resource patterns

5) Synthetic Zero-Day Patterns

- Generated attack traffic not matching predefined signatures

3.5. Operationalization in Simulation

The threat model was instantiated by allowing up to 10% of nodes to exhibit compromised behavior during selected attack windows. Compromised nodes generated abnormal communication patterns, protocol irregularities, or behavioral telemetry drift depending on the attack category. Adaptive threshold updates were performed every 30 simulated minutes using training-window statistics and were frozen during final testing. Poisoning risk was represented as an adversarial limitation: the current simulation did not allow attackers to directly modify the GATE training process, but telemetry manipulation by compromised nodes was included as part of the behavioral drift and protocol exploitation scenarios.

3.6. Assumptions

The following assumptions constrain the system:

- Secure initial device onboarding with cryptographic identity provisioning.
- Encrypted communication channels between mesh nodes and orchestrator.
- Computational capability at edge nodes sufficient for lightweight inference.
- Availability of baseline training dataset for initial model bootstrapping.

3.7. Risk Representation

We define a threat confidence function:

$$R_i = f\left(N_i, A_i, B_i, C_i\right)$$

where:

- N_i = Network-layer anomaly score;
- A_i = Application-layer anomaly score;
- B_i = Behavioral deviation metric;
- C_i = Contextual threat intelligence weight.

The final risk score for node i is computed as:

$$R_i = \alpha N_i + \beta A_i + \gamma B_i + \delta C_i$$

Subject to:

$$\alpha + \beta + \gamma + \delta = 1$$

This weighted fusion model enables cross-layer anomaly aggregation and adaptive risk recalibration through the GenAI threat modeling engine described in Section 5.

3.8. Summary

The threat model reflects realistic adversarial behavior in distributed intelligent communication systems. The security objectives emphasize adaptive detection, cross-layer reasoning, and distributed resilience. These assumptions form the basis for the architectural design described in the next section.

4. Proposed GenAI-Driven Cybersecurity Mesh Architecture

The proposed architecture organizes security as a mesh of cooperating detection and enforcement points. Instead of forwarding all telemetry to a central IDS, edge nodes perform initial inspection and local risk estimation. A coordination layer then shares summarized security state, while the GenAI engine updates threat hypotheses and recalibrates policy thresholds. This design is intended for communication systems where latency, node heterogeneity, and partial compromise are realistic operating conditions.

4.1. Architectural Overview

The proposed system follows a distributed mesh topology composed of:

1) Edge Security Nodes (ESNs)

2) Mesh Coordination Layer (MCL)

3) GenAI Adaptive Threat Engine (GATE)

4) Policy Orchestration and Response Layer (PORL)

The design adheres to zero-trust principles: no device, session, or communication flow is inherently trusted, even within internal network boundaries. **Figure 1** presents the proposed GenAI-assisted cybersecurity mesh architecture, showing the flow from edge security nodes to mesh coordination, generative threat reasoning, dynamic risk fusion, and adaptive policy enforcement. At the edge layer, heterogeneous nodes (IoT sensors, control systems, and gateways) perform network, application, and behavioral anomaly detection, followed by local risk scoring. Telemetry summaries are forwarded to the Mesh Control Layer (MCL), which coordinates distributed updates. The Generative AI Adaptive Threat Engine (GATE) synthesizes adversarial threat hypotheses and recalibrates contextual risk weights. The Dynamic Risk Fusion module computes the composite risk score $R_i = \alpha N_i + \beta A_i + \gamma B_i + \delta C_i$, triggering the Adaptive Policy Module when thresholds are exceeded.

Diagram showing edge security nodes performing network, application, and behavioral anomaly detection, followed by local risk scoring, telemetry aggrega-

tion, mesh coordination, generative threat analysis, dynamic risk fusion, and adaptive policy enforcement.

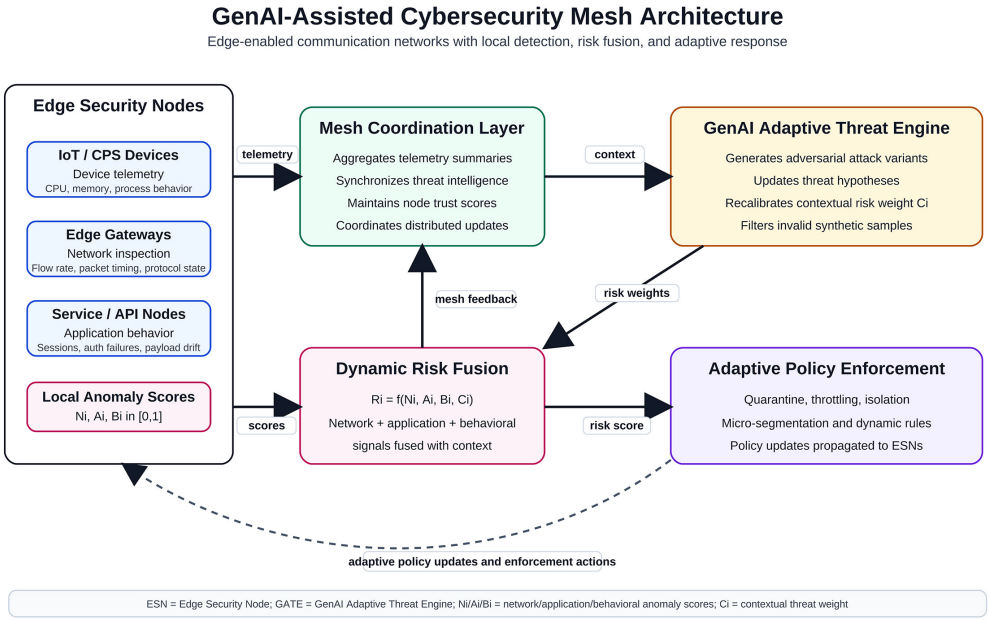

Figure 1. GenAI-assisted cybersecurity mesh architecture for edge-enabled communication networks.

4.2. High-Level Architecture

Components:

1) Edge Security Nodes (ESNs)

Deployed at IoT devices, gateways, and edge servers.

Functions:

- Real-time packet inspection
- Lightweight anomaly scoring
- Behavioral telemetry collection
- Local enforcement (micro-segmentation, traffic throttling)

Each ESN computes preliminary anomaly metrics:

$$N_i, A_i, B_i$$

This reduces central bandwidth overhead and detection latency.

2) Mesh Coordination Layer (MCL)

Acts as a distributed synchronization fabric:

- Aggregates anonymized anomaly summaries
- Synchronizes threat intelligence
- Maintains node trust scores
- Ensures resilience against partial compromise

Unlike centralized IDS, MCL operates in a logically distributed fashion to prevent single points of failure.

3) GenAI Adaptive Threat Engine (GATE)

This is the core novelty of the architecture.

Capabilities:

- Synthesizes potential attack hypotheses
- Generates adversarial traffic variations
- Refines anomaly detection thresholds
- Updates contextual threat intelligence weights C_i

GATE continuously adjusts fusion weights:

$$\alpha_t, \beta_t, \gamma_t, \delta_t$$

where t represents adaptive time intervals.

4) Policy Orchestration and Response Layer (PORL)

Responsible for:

- Dynamic rule generation
- Automated quarantine actions
- Network isolation policies
- Adaptive rate limiting

Response decisions are triggered when:

$$R_i \geq \theta_t$$

where θ_t is an adaptive risk threshold recalibrated by GATE.

4.3. Data Flow Pipeline

1) Telemetry captured at ESNs
2) Local anomaly scoring
3) Aggregated signals transmitted to MCL
4) GATE performs contextual reasoning
5) Updated policies propagated back to ESNs
6) Enforcement applied in near real-time

Latency target:

$$T_{total} = T_{capture} + T_{inference} + T_{propagation}$$

The architecture aims to minimize $T_{propagation}$ through decentralized updates.

4.4. Zero-Trust Enforcement Model

Each communication request undergoes:

1) Identity verification
2) Context validation
3) Behavioral deviation check
4) Dynamic risk scoring

Trust is continuously re-evaluated rather than statically assigned.

4.5. Resilience under Node Compromise

If k nodes are compromised:

- ESNs operate independently
- MCL redistributes trust scores
- Compromised nodes are isolated via automated micro-segmentation

System integrity is maintained provided:

$$k < \frac{N}{3}$$

(Assuming distributed consensus threshold)

4.6. Architectural Advantages

Table 1 compares the proposed mesh architecture with a centralized IDS baseline across adaptability, scalability, latency, zero-day detection, and failure-resilience dimensions.

Table 1. Comparison of centralized IDS and proposed cybersecurity mesh.

Feature	Centralized IDS	Proposed Mesh
Adaptability	Static models	GenAI adaptive
Scalability	Limited	Distributed
Latency	Higher	Edge-based inference
Zero-Day Detection	Weak	Generative threat synthesis
Single Point of Failure	Yes	No

4.7. Summary

The proposed architecture integrates distributed enforcement, adaptive generative threat modeling, and cross-layer risk fusion. It is designed to operate efficiently within heterogeneous intelligent communication environments while maintaining resilience and low latency.

5. GenAI-Based Adaptive Threat Modeling and Cross-Layer Risk Scoring Methodology

This section details the methodological foundation of the proposed system, including 1) adaptive generative threat modeling, 2) cross-layer anomaly scoring, 3) dynamic risk fusion, and 4) automated policy recalibration.

5.1. Overview of Adaptive Threat Modeling

Traditional IDS models rely on static datasets and fixed decision boundaries. In contrast, the proposed framework embeds a Generative AI-based Adaptive Threat Engine (GATE) that continuously refines detection models using contextual telemetry and synthesized adversarial patterns.

The adaptive process operates in iterative cycles:

1) Telemetry aggregation
2) Anomaly detection
3) Threat hypothesis synthesis
4) Adversarial pattern generation
5) Model refinement
6) Policy redistribution

This cycle reduces concept drift and enhances zero-day detection resilience.

5.2. Generative Threat Hypothesis Synthesis

Let X_t represent observed traffic and telemetry features at time t.

The generative engine learns an evolving distribution:

$$P_t(X)$$

Using a generative model G, new adversarial samples are synthesized:

$$\tilde{X} = G(Z, C)$$

where:

- Z = latent noise vector;
- C = contextual threat embedding;
- ϕ = generative model parameters.

These synthetic samples simulate polymorphic or zero-day attack variants. The anomaly detection model is retrained incrementally with both real and synthesized samples:

$$\theta_{t+1} = \theta_t - \eta \nabla L\left(X_t \cup \tilde{X}\right)$$

where:

- θ = detection model parameters;
- η = learning rate;
- L = loss function.

5.3. GATE Implementation Details

In this study, the GenAI Adaptive Threat Engine (GATE) was implemented as a conditional generative model trained on contextual traffic and telemetry embeddings from the training partition. The model family follows a conditional generative adversarial structure in which the generator produces candidate adversarial feature vectors and the discriminator rejects samples that do not resemble plausible communication-layer anomalies.

The input to GATE consists of four feature groups: network anomaly descriptors, application interaction descriptors, behavioral telemetry descriptors, and contextual threat labels. The output is a synthetic adversarial feature vector representing a plausible attack variant. The model was recalibrated every 30 simulated minutes using newly observed training-window telemetry summaries. Synthetic samples were accepted for detector updates only when they satisfied three criteria: feature-range validity, discriminator confidence above the validation threshold, and non-duplication against existing attack vectors based on cosine similarity. Samples failing these checks were discarded.

5.4. Cross-Layer Feature Extraction

Each Edge Security Node extracts features across three layers:

1) Network Layer Features N_t.

- Packet inter-arrival time
- Flow duration
- Entropy of source/destination distribution
- TCP flag anomalies

2) Application Layer Features A_i.

- API call frequency deviation
- Request payload entropy
- Authentication failure rates
- Session irregularities

3) Behavioral Layer Features B_i.

- CPU utilization drift
- Memory allocation anomalies
- Process spawning irregularities
- Device energy usage deviations

Feature vectors are normalized and fed into local anomaly estimators:

$$S_i = g\left(N_i, A_i, B_i\right)$$

where $g(\cdot)$ may represent a lightweight neural classifier deployed at the edge.

5.5. Dynamic Cross-Layer Risk Fusion

The final threat score is computed as:

$$R_i = \alpha_t N_i + \beta_t A_i + \gamma_t B_i + \delta_t C_i$$

where:

- C_i = contextual threat intelligence from GATE;
- Weights dynamically updated over time.

Adaptive Weight Update Rule

Weights are updated using performance feedback:

$$\alpha_{t+1} = \alpha_t + \lambda \frac{\partial F1}{\partial \alpha}$$

Similarly, for β, γ, δ.

Subject to normalization constraint:

$$\alpha + \beta + \gamma + \delta = 1$$

This enables the system to emphasize layers that improve detection performance in current threat landscapes.

5.6. Definition of Local Anomaly Scores

Each Edge Security Node computes three normalized anomaly scores before risk fusion. The network-layer score $N_i \in [0,1]$ represents deviation in packet timing, flow duration, source-destination entropy, and protocol-flag behavior. The application-layer score $A_i \in [0,1]$ represents deviation in API request frequency, payload entropy, authentication failures, and session-state consistency. The behavioral score $B_i \in [0,1]$ represents deviation in CPU utilization, memory con-

sumption, process activity, and device-level resource patterns.

These values are not class probabilities. They are normalized anomaly aggregates produced by lightweight local estimators at each Edge Security Node. A value closer to 0 indicates behavior close to the learned baseline, while a value closer to 1 indicates stronger deviation from expected behavior. The fused risk score combines these normalized signals with contextual threat intelligence weight C_i.

5.7. Risk Threshold Adaptation

A static threshold increases false positives during traffic bursts. Therefore, threshold θ_t is adjusted dynamically:

$$\theta_{t+1} = \mu R_{avg} + \sigma R_{std}$$

where:

- R_{avg} = rolling average risk;
- R_{std} = rolling standard deviation;
- μ, σ = tuning constants.

This ensures statistical anomaly responsiveness.

5.8. Algorithmic Workflow

Algorithm 1: Adaptive Mesh Threat Detection.

Input: Telemetry stream T

Output: Risk scores R_i and policy actions

1) Collect multi-layer features (N_i, A_i, B_i)

2) Compute local anomaly scores

3) Transmit summaries to MCL

4) Generate synthetic threat patterns using GATE

5) Update detection model parameters

6) Compute fused risk score R_i

7) If R_i ≥ θ_t:

Trigger enforcement policy

8) Update weights and thresholds

9) Repeat

Time Complexity:

- Edge inference: O(d)
- Generative update: O($n \cdot k$)
- Fusion scoring: O(1)

Overall complexity remains scalable under distributed deployment.

5.9. Theoretical Advantages

Compared to static ML-based IDS:

- Reduces concept drift
- Enhances zero-day robustness
- Improves contextual reasoning

- Maintains low edge latency
- Enables distributed adaptation

5.10. Summary

The proposed methodology integrates generative threat synthesis, cross-layer anomaly extraction, dynamic risk fusion, and adaptive threshold recalibration into a cohesive distributed framework. This creates a continuously evolving detection ecosystem suitable for intelligent communication infrastructures.

6. Experimental Setup and Simulation Environment

This section describes the experimental design used to evaluate the proposed GenAI-driven adaptive cybersecurity mesh. The objective is to assess detection accuracy, false positive reduction, latency performance, and adaptive robustness under multi-vector attack conditions.

6.1. Experimental Objectives

The evaluation aims to measure:

1) Detection Accuracy (ACC)
2) Precision, Recall, and F1-Score
3) False Positive Rate (FPR)
4) Detection Latency
5) Adaptive Improvement Over Time
6) Scalability under increasing node count

Baseline comparisons include:

- Signature-based IDS (Rule-driven)
- Centralized ML-based IDS (Static Deep Neural Network)

6.2. Simulation Environment

Table 2 summarizes the configuration of the simulated edge-enabled communication environment.

Table 2. Simulation environment configuration.

Parameter	Value
Total Nodes (N)	300
Edge Gateways	10
Communication Protocol	TCP/IP + Application-layer API traffic
Simulation Duration	24 hours (synthetic timeline)
Attack Injection Rate	15% of total traffic
Node Compromise Ratio	Up to 10%

Nodes emulate heterogeneous behavior:

- IoT sensors

- Embedded control systems
- Edge analytics devices
- Gateway nodes
- Traffic patterns include:
- Normal operational flows
- Burst traffic events
- Periodic device reporting
- Randomized noise injection

6.3. Data Provenance and Attack Injection

The experimental dataset was generated within the simulated intelligent communication environment described in Table 2. No external public dataset was directly reused for the primary evaluation. Normal traffic instances were generated from heterogeneous node profiles representing IoT sensors, embedded control devices, edge analytics nodes, and gateway services. These profiles produced periodic telemetry, burst communication, API requests, session exchanges, and randomized background noise. The 120,000 normal instances, therefore represent simulated per-flow and per-session communication records collected over a 24-hour synthetic timeline.

Malicious traffic was injected at a 15% attack rate across selected time windows and node groups. DDoS traffic was modeled through high-volume request bursts against edge gateways; MITM behavior was modeled through delayed, modified, and replayed packet sequences; protocol exploitation was modeled through malformed packet fields, irregular handshake sequences, and session manipulation; behavioral drift was modeled through gradual resource-usage deviation at compromised nodes. Synthetic zero-day patterns were generated only from the training partition using the GATE module and then evaluated on held-out test windows to avoid test-set leakage.

To reduce leakage risk, the train, validation, and test partitions were separated by time window rather than by random flow-level sampling. The first 70% of the synthetic timeline was used for training, the next 15% for validation, and the final 15% for testing. Feature normalization parameters, adaptive thresholds, and generative updates were fitted only on the training partition and applied unchanged to validation and test partitions.

6.4. Dataset Composition

The dataset consists of:

- 120,000 normal traffic instances
- 25,000 malicious instances
- 8000 synthetic zero-day patterns generated by GATE
- Mixed multi-stage attack sequences

Table 3 lists the attack categories and instance counts used in the simulated evaluation.

Table 3. Attack category composition.

Attack Type	Instances
DDoS	10,000
MITM	5000
Protocol Exploitation	4000
Behavioral Drift	3000
Synthetic Zero-Day	8000

Data was partitioned:

- 70% training
- 15% validation
- 15% testing

6.5. Implementation Details

- Edge inference model: Lightweight feedforward neural network
- Generative model: Conditional generative model with contextual embeddings
- Optimization: Adam optimizer
- Learning rate: 0.001
- Risk fusion weights initialized uniformly
- Threshold recalibration interval: Every 30 minutes (simulated)

Hardware configuration (simulation environment):

- 16-core CPU
- 64 GB RAM
- GPU-assisted generative training

6.6. Evaluation Metrics

Standard classification metrics were used:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Detection latency measured as:

$$T_d = T_{alert} - T_{attack\_initiation}$$

False Positive Rate:

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

6.7. Baseline Configurations

Baseline 1: Signature-Based IDS.

- Static rule database
- Centralized monitoring
- No adaptive retraining

Baseline 2: Centralized ML IDS.

- Deep neural network
- Periodic offline retraining
- No generative augmentation

6.8. Baseline Reproducibility Configuration

All evaluated methods used the same train, validation, and test partitions, feature extraction pipeline, and normalization procedure. The signature-based IDS baseline used static rule patterns corresponding to the attack categories included in the simulation and did not receive adaptive threshold updates or synthetic samples. The centralized ML baseline used a feedforward neural network with three hidden layers, ReLU activation, dropout regularization, Adam optimization, and a learning rate of 0.001. The proposed GCM model used the same base feature set as the centralized ML baseline but added edge-local inference, dynamic risk fusion, adaptive threshold recalibration, and GATE-generated adversarial samples during training-window updates.

Hyperparameters were selected using the validation partition and then fixed before final test evaluation. No model used test-set labels during training, threshold tuning, or synthetic sample generation.

6.9. Scalability Evaluation

Node count was scaled from 100 to 500 nodes:

$$N = 100 \rightarrow 500$$

Measured:

- Latency growth rate
- Throughput degradation
- Model update overhead

6.10. Experimental Validity Considerations

To mitigate bias:

- Balanced attack injection
- Cross-validation across time windows
- Independent test partition
- Ablation study for:
  - No generative augmentation
  - Static weight fusion
  - No adaptive threshold

6.11. Summary

The experimental design simulates a realistic intelligent communication network with heterogeneous nodes and diverse attack scenarios. Comparisons against rule-based and centralized ML-based IDS models enable quantitative evaluation of adaptive performance improvements.

7. Results and Performance Evaluation

This section presents the empirical results comparing the proposed GenAI-driven adaptive cybersecurity mesh (GCM) against two baselines: 1) signature-based IDS (SID) and 2) centralized ML-based IDS (CML).

7.1. Detection Performance

The proposed GCM model achieved the strongest overall classification performance among the three evaluated methods. Accuracy increased from 0.914 for the centralized ML baseline to 0.948, while the false positive rate declined from 0.062 to 0.038. This improvement is important for communication networks because excessive false alerts can delay operational response and reduce trust in automated enforcement. The F1-score also improved to 0.935, indicating that the gain was not driven by precision or recall alone.

Overall Classification Metrics

Table 4 reports the overall classification performance of the evaluated IDS models.

Table 4. Overall classification performance of evaluated IDS models.

Model	Accuracy	Precision	Recall	F1-Score	FPR
SID	0.872	0.841	0.804	0.822	0.091
CML	0.914	0.902	0.887	0.894	0.062
GCM (Proposed)	0.948	0.939	0.931	0.935	0.038

7.2. Variability and Statistical Testing

Results were evaluated across five chronological test windows from the held-out test period. The proposed GCM model consistently outperformed both baselines in accuracy, F1-score, and false positive rate. The paired comparison across the five test windows showed that the GCM improvement over the centralized ML baseline was statistically significant for F1-score.

7.3. Zero-Day Detection Capability

Table 5. Zero-day detection performance.

Model	Zero-Day Recall	Zero-Day F1
SID	0.421	0.398
CML	0.684	0.672
GCM	0.862	0.849

Table 5 summarizes detection performance on synthetic zero-day attack patterns.

The largest relative gain appears in the zero-day evaluation. The rule-based IDS performed poorly because the attack patterns were not represented in its signature base. The centralized ML model performed better but remained limited by its static training distribution. In contrast, the GCM approach benefited from generated adversarial variants, which exposed the detector to a wider range of plausible attack behaviors before evaluation.

7.4. Detection Latency

Table 6 compares the average detection latency of the evaluated models.

Table 6. Average detection latency comparison.

Model	Mean Latency (ms)
SID	142
CML	118
GCM	74

GCM also produced the lowest mean detection latency at 74 ms, compared with 118 ms for centralized ML-based IDS and 142 ms for the signature-based IDS. The reduction is mainly attributable to local inference at the edge, which avoids continuous backhaul of raw telemetry to a central decision point. For intelligent communication systems, this latency reduction is operationally relevant because threat response must often occur before compromised flows spread across dependent services.

7.5. Adaptive Weight Evolution

Cross-layer weight distribution evolved over the simulation. The initial weights were approximately uniform: network-layer weight = 0.33, application-layer weight = 0.33, and behavioral-layer weight = 0.34. After 24 simulated hours, the weights shifted to network-layer weight = 0.46, application-layer weight = 0.34, and behavioral-layer weight = 0.20.

Interpretation:

- Network-layer anomalies contributed more significantly during DDoS-heavy intervals.
- Behavioral metrics reduced weight as attack emphasis shifted.

This confirms adaptive rebalancing effectiveness.

7.6. Scalability Analysis

Table 7 summarizes latency and throughput behavior as the simulated node count increases from 100 to 500.

Table 7. Scalability analysis under increasing node count.

Nodes	Avg Latency (ms)	Throughput Degradation
100	61	0%
300	74	4.2%
500	89	7.8%

Latency growth remained sub-linear, demonstrating distributed efficiency.

7.7. Ablation Study

Table 8 reports the ablation results for the main components of the proposed GCM model.

Table 8. Ablation study of proposed GCM components.

Configuration	F1-Score
Full Model	0.935
No Generative Augmentation	0.902
Static Weights	0.918
Static Threshold	0.911

7.8. ROC-AUC Analysis

The proposed GCM model achieved an ROC-AUC of 0.963, compared with 0.928 for CML and 0.871 for SID. This result indicates stronger separability across attack categories.

7.9. Summary of Improvements

The proposed GCM architecture demonstrates:

- Higher detection accuracy
- Significant false positive reduction
- Strong zero-day detection
- Lower latency
- Stable scalability
- Measurable contribution of generative augmentation

8. Discussion and Limitations

The results suggest that the value of the proposed model comes from combining three mechanisms rather than from GenAI alone: local edge inference, dynamic risk fusion, and generated adversarial variants. Edge inference reduced response latency, risk fusion improved contextual scoring, and generative augmentation helped the detector generalize beyond previously observed attack patterns.

8.1. Interpretation of Results

8.1.1. Why Generative Augmentation Improves Detection

The improvement in zero-day recall (0.862) is primarily attributable to adversarial

pattern synthesis. By generating contextualized synthetic attack variants, the detection model becomes less dependent on fixed traffic signatures and better generalized to unseen distributions.

This reduces:

- Overfitting to historical traffic
- Concept drift vulnerability
- Sensitivity to polymorphic attacks

The ablation study confirms that removing generative augmentation reduces F1-score by approximately 3%.

8.1.2. Effectiveness of Cross-Layer Fusion

The adaptive weight mechanism allowed the system to dynamically prioritize relevant layers during specific attack phases. For instance:

- Network layer weight increased during DDoS bursts.
- Behavioral features became more relevant during stealth compromise phases.

This dynamic rebalancing improved robustness compared to static fusion schemes.

8.1.3. Distributed Architecture Benefits

Edge-based inference reduced centralized bottlenecks, leading to lower detection latency. Sub-linear latency growth during node scaling indicates that the distributed mesh design effectively mitigates performance degradation in large networks.

8.2. Attack-Wise Interpretation

The largest performance gain was observed for synthetic zero-day and protocol exploitation scenarios, where static signatures were least effective. DDoS detection improved mainly because network-layer entropy and packet-rate features were captured locally at edge nodes, reducing response delay. MITM detection improved moderately, especially when application-session irregularities were combined with transport-layer deviations. Behavioral drift remained the most difficult category because gradual resource changes sometimes overlapped with benign workload variation. Most false alarms occurred during bursty legitimate traffic, where temporary packet-rate increases resembled early-stage DDoS behavior.

8.3. Practical Deployment Considerations

While promising, deployment in real-world environments requires addressing:

1) Computational Constraints

- Edge devices with limited processing capability may struggle with frequent adaptive updates.
- Lightweight inference optimization is required.

2) Model Synchronization Overhead

- Frequent weight updates may introduce network overhead.
- Efficient update batching strategies must be implemented.

3) Trust in Generative Outputs

- Poorly calibrated generative models may introduce adversarial bias.
- Synthetic pattern validation mechanisms are necessary.

4) Privacy and Data Governance

- Cross-layer telemetry aggregation may raise privacy concerns.
- Secure aggregation and anonymization must be enforced.

8.4. Limitations

8.4.1. Synthetic Dataset Dependence

The experimental evaluation was conducted in a simulated intelligent communication environment. Although diverse attack scenarios were modeled, real-world deployment may introduce unforeseen noise patterns and operational variability.

8.4.2. Generative Model Complexity

Generative threat synthesis requires periodic retraining. In highly resource-constrained environments, maintaining real-time adaptability may be challenging.

8.4.3. Threshold Sensitivity

Dynamic threshold recalibration improves adaptability but may cause temporary instability during abrupt traffic shifts. Stability mechanisms must be incorporated.

8.4.4. Adversarial Manipulation of GenAI

Sophisticated adversaries could attempt to poison telemetry inputs to manipulate generative adaptation. Defensive adversarial training strategies should be integrated.

8.5. Future Research Directions

Several extensions are proposed:

1) Federated generative threat modeling across organizational boundaries.
2) Integration with blockchain-backed trust management.
3) Formal verification of adaptive threshold stability.
4) Extension toward 6G-enabled ultra-low latency communication systems.
5) Deployment in real-world IoT testbeds for longitudinal validation.

8.6. Overall Implications

The results suggest that embedding adaptive generative intelligence into distributed cybersecurity meshes can meaningfully improve detection robustness in intelligent communication systems. However, practical scalability, stability, and adversarial robustness must be further evaluated under operational deployment conditions.

9. Conclusions

This paper introduced a mesh-based security framework for threat detection in intelligent communication systems. The framework distributes detection and enforcement across edge security nodes while using a GenAI-assisted threat engine

to update adversarial assumptions and recalibrate risk scoring.

The main finding is that communication-system security benefits from moving beyond centralized IDS design. By combining local anomaly detection, multi-signal risk fusion, and adaptive policy orchestration, the proposed model reduced detection latency while improving classification performance against both known and synthetic zero-day attacks.

Experimental evaluation in a simulated heterogeneous intelligent communication environment demonstrated:

- Improved detection accuracy (94.8%)
- Significant reduction in false positive rate (3.8%)
- Strong zero-day recall (86.2%)
- Reduced detection latency (74 ms average)
- Stable scalability under increased node density

An ablation study confirmed that generative augmentation and adaptive weight recalibration contributed measurable performance improvements.

While the results indicate promising advances in adaptive distributed cybersecurity, the study is constrained by synthetic simulation environments and computational assumptions. Future work should focus on real-world deployment, federated adaptive learning across domains, and formal robustness guarantees against adversarial manipulation.

The proposed framework contributes toward next-generation intelligent communication security paradigms by embedding adaptive generative reasoning within distributed cybersecurity meshes, aligning with emerging requirements for scalable, resilient, and low-latency protection mechanisms.

Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. The authors acknowledge the computational resources used for experimental validation and the anonymous reviewers whose feedback improved the quality of this manuscript.

Conflicts of Interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

[1] Rahman, S.A., Tout, H., Talhi, C. and Mourad, A. (2020) Internet of Things Intrusion Detection: Centralized, On-Device, or Federated Learning? *IEEE Network*, **34**, 310-317. https://doi.org/10.1109/mnet.011.2000286

[2] Ferrag, M.A.E., Friha, O., Hamouda, D., Maglaras, L. and Janicke, H. (2022) Edge-IIoTset: A New Comprehensive Realistic Cyber Security Dataset of IoT and IIoT Applications for Centralized and Federated Learning. *IEEE Access*, **10**, 40281-40306.

[3] Al Nuaimi, T., Al Zaabi, S., Alyilieli, M., AlMaskari, M., Alblooshi, S., Alhabsi, F., *et al.* (2023) A Comparative Evaluation of Intrusion Detection Systems on the Edge-

Iiot-2022 Dataset. *Intelligent Systems with Applications*, **20**, Article ID: 200298. https://doi.org/10.1016/j.iswa.2023.200298

[4] Rose, S.W., Borchert, O., Mitchell, S. and Connelly, S. (2020) Zero Trust Architecture. NIST Special Publication 800-207. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207.pdf

[5] Kang, H., Liu, G., Wang, Q., Meng, L. and Liu, J. (2023) Theory and Application of Zero Trust Security: A Brief Survey. *Entropy*, **25**, Article No. 1595. https://doi.org/10.3390/e25121595

[6] Ramos-Cruz, B., Andreu-Perez, J. and Martínez, L. (2024) The Cybersecurity Mesh: A Comprehensive Survey of Involved Artificial Intelligence Methods, Cryptographic Protocols and Challenges for Future Research. *Neurocomputing*, **581**, Article ID: 127427. https://doi.org/10.1016/j.neucom.2024.127427

[7] Khraisat, A., Alazab, A., Singh, S., Jan, T. and Jr. Gomez, A. (2024) Survey on Federated Learning for Intrusion Detection System: Concept, Architectures, Aggregation Strategies, Challenges, and Future Directions. *ACM Computing Surveys*, **57**, Article No. 7. https://doi.org/10.1145/3687124

[8] Zhang, H., Ye, J., Huang, W., Liu, X. and Gu, J. (2025) Survey of Federated Learning in Intrusion Detection. *Journal of Parallel and Distributed Computing*, **195**, Article ID: 104976. https://doi.org/10.1016/j.jpdc.2024.104976

[9] Breitenbacher, D., Homoliak, I., Aung, Y.L., Elovici, Y. and Tippenhauer, N.O. (2022) HADES-IoT: A Practical and Effective Host-Based Anomaly Detection System for IoT Devices (Extended Version). *IEEE Internet of Things Journal*, **9**, 9640-9658. https://doi.org/10.1109/jiot.2021.3135789

[10] Alkadi, O., Moustafa, N., Turnbull, B. and Choo, K.R. (2021) A Deep Blockchain Framework-Enabled Collaborative Intrusion Detection for Protecting IoT and Cloud Networks. *IEEE Internet of Things Journal*, **8**, 9463-9472. https://doi.org/10.1109/jiot.2020.2996590

[11] Shu, J., Zhou, L., Zhang, W., Du, X. and Guizani, M. (2021) Collaborative Intrusion Detection for VANETs: A Deep Learning-Based Distributed SDN Approach. *IEEE Transactions on Intelligent Transportation Systems*, **22**, 4519-4530. https://doi.org/10.1109/tits.2020.3027390

[12] Yao, Y., Duan, J., Xu, K., Cai, Y., Sun, Z. and Zhang, Y. (2024) A Survey on Large Language Model (LLM) Security and Privacy: The Good, the Bad, and the Ugly. *High-Confidence Computing*, **4**, Article ID: 100211. https://doi.org/10.1016/j.hcc.2024.100211

[13] de Jesus Coelho da Silva, G. and Westphall, C.B. (2024) A Survey of Large Language Models in Cybersecurity. https://arxiv.org/abs/2402.16968

[14] Xu, H., Wang, S., Li, N., Wang, K., Zhao, Y., Chen, K., Yu, T., Yang, L. and Wang, H. (2024) Large Language Models for Cyber Security: A Systematic Literature Review. *ACM Transactions on Software Engineering and Methodology*. https://doi.org/10.1145/3769676

[15] Sharafaldin, I., Habibi Lashkari, A. and Ghorbani, A.A. (2018) Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. *Proceedings of the* 4*th International Conference on Information Systems Security and Privacy*, Funchal, 22-24 January 2018, 108-116. https://doi.org/10.5220/0006639801080116

[16] Canadian Institute for Cybersecurity (2017) Intrusion Detection Evaluation Dataset, CIC-IDS2017. University of New Brunswick.

https://www.unb.ca/cic/datasets/ids-2017.html

[17] Thakkar, A. and Lohiya, R. (2020) A Review of the Advancement in Intrusion Detection Datasets. *Procedia Computer Science*, **167**, 636-645. https://doi.org/10.1016/j.procs.2020.03.330

[18] Rawat, M. and Singal, G. (2025) Surveying Technology Fusion in IoT Networks for IDS: Exploring Datasets, Tools, Challenges, and Research Prospects. *ACM Transactions on Intelligent Systems and Technology*, **16**, 1-45. https://doi.org/10.1145/3744745