CAP-SIP-Guard: Alerting for 5G Edge IoT Authentication-Hijacking in Heterogeneous Autonomous Systems

Abdul Aziz[*], Mansoor Qadir[†], Shah Raiz Zeb[‡]

Abstract

This paper presents a time-sensitive lightweight Session Initiation Protocol (SIP) alerting system for the edge Internet of Things (IoT) network. The framework encapsulates Common Alerting Protocol (CAP) objects in SIP NOTIFY/PUBLISH messages using Internet Key Exchange version 2 (IKEv2). Micro-certificates provide zero-trust onboarding and end-to-end assurance, in-band methods for authentication hijacking detection. A 20-node Vehicular IoT testbed across Wi-Fi 6E, Long-Term Evolution (LTE)-Advanced, and 5G NR evaluates SIP processing delay, CAP alert delivery, and detection under REGISTER spoofing, reINVITE bursts, and nonce replay. CAP-SIP-Guard achieves 46 milliseconds (ms) median SIP processing with 10 vehicles and <30 ms edge CAP propagation, 98 % detection with under 2 % false positive. SIP alert latency drops 31 % and authenticated throughput improves 2.4× relative to Transport Layer Security TLS-SIP and Datagram Constrained Application Protocol (CoAP) baselines. These results show improved latency and detection performance in controlled environment for heterogeneous, latency-sensitive IoT.

Keywords: Session Initiation Protocol, Common Alerting Protocol, Authentication Hijacking, Intrusion Detection, IoT Security.

Introduction

Heterogeneous Internet of Things (IoT) increasingly rely on low latency signaling across distributed and resource-constrained environments. However, ensuring both efficiency and resilience against control-plane attacks remains a critical challenge. Signaling mechanisms must be both efficient and resilient to control-plane attacks due to its adaptable Uniform Resource Identifiers (URI) addressing, extensible header format, and SUBSCRIBE/NOTIFY occasioning model, the Session Initiation Protocol (SIP) has become a viable choice in session control throughout distributed systems owing to its traits as a robust design candidate in forming a control plane within mixed IoT fabrics (Rosenberg et al., 2002; Roach, 2012) suggest. In spite of this, production SIP deployments do not often offer standardized techniques of surfacing, detecting and containing authentication hijacking attacks in an in-band

[*]Department of Computer Science, Iqra National University, Peshawar 25000, Pakistan, abdul1.mkd@gmail.com

[†]Corresponding Author: Department of Computer Science, CECOS University of IT & Emerging Sciences, Peshawar 25000, Pakistan, mansoor.qadir@hotmail.com

[‡]Department of Computer Science, CECOS University of IT & Emerging Sciences, Peshawar 25000, Pakistan, shahraizzeb@gmail.com

manner. Regular attacks like REGISTER spoofing, re-INVITE storms, and nonce replay can be quite detrimental to availability and compromise trust in the signaling infrastructure, and most operators identify these abnormalities using out-of-band analytics and intrusion detection systems (Kaufman et al., 2014; Shelby et al., 2014). Though the effect in detecting any malicious activity adds more delays of few seconds and this is not acceptable in IoT where latency is crucial (Hartke, 2015).

SIP‑provides URI-based addressing and SUBSCRIBE/NOTIFY semantics, making it suitable for heterogeneous IoT control planes. One of the most common attack vectors in real application, authentication hijacking, is a significant weakness of SIP-based infrastructures where attackers seek to exploit vulnerabilities in session set-ups (Kschischang & Sorokine, 2002; Im & Lee, 2024). The intercept REGISTER, INVITE or re-INVITE messages are spoofed to impersonate valid agents, interfere with call flows or take control of running sessions to break signaling integrity and call continuity (Roach, 2012). A lightweight in-band detection and alerting can be achieved by embedding the Common Alerting Protocol (CAP) (Waidyanatha, 2008), which is an OASIS (Organization for the Advancement of Structured Information Standards) standard used for the interoperable dissemination of structured warning messages across various systems and platforms, into SIP signaling (OASIS, 2019; Bairi et al., 2025; Bai et al., 2021; Wang et al., 2024; Xu et al., 2025). This allows alerting to be sent natively in the trusted signaling path by encapsulating CAP objects in SIP Notify or Publish requests. The related works on SIP-based IoT communication, alerting in distributed systems, edge vehicular tier authentication hijack detection, and CAP integration are discussed in related work section. The architecture, adaptive predictor, metrics, and methodological framework are discussed in methodology section. The results and comparative analysis are presented in results section.

Related Work

The CAP messages facilitate syndication across cellular broadcast, Short Message Service (SMS), and public safety systems through the inclusion of the type of warning, severity, suggested response, and scope of the geographic area in a normalized Extensible Markup Language (XML) format (Alzahrani, 2025). The integration of the CAP messages in the SIP NOTIFY or PUBLISH transactions, there are three major benefits of the CAP messages. The first is the Native Routing, which ensures the movement of the CAP messages through existing security associations by enabling the SIP proxies to route the CAP messages in the same path as the control messages. Protocol Agnosticism: The

downstream gateways ensure a single source of truth for incident reporting through payload translation of CAP into Message Queuing Telemetry Transport (MQTT) (Bhatti et al., 2024). CoAP, or WebSocket without loss of semantic meaning. In this regard, the present study proposes a standards-based architecture that leverages (Internet Key Exchange version 2 (IKEv2) micro-certificates for end-to-end assurance and zero-trust onboarding, integrates CAP objects into SIP NOTIFY/PUBLISH messages. The approach offers a one-shot CAP digest for MQTT, CoAP and WebSocket bridging, as demonstrated by (Kaufman et al., 2014; Bhatti et al., 2024).

The predictors (contact via. drift, nonce-reuse and burst rates) are useful for dealing with concept drift with low CPU/Memory footprint and has interpretable outputs where decisions are made by combining outputs from the predictor and the rules, ensuring effective detection with low latency (Aminikhanghahi & Cook, 2017; Chandola et al., 2009; Gama et al., 2014). This offers a rigorous methodology, including comparative analysis of systems using only rules, systems using rules and Machine Learning, and systems with and without IKE, from edge and cloud registrars, and across different access technologies, namely, Wi-Fi 6E, Long-Term Evolution (LTE)-A, and 5G New Radio (NR), (Schulzrinne, 2007; Gonzales et al., 2025). The determination of the feasibility of independent verification and offers practical recommendations for provisioning through a concise Mixed-Integer Linear Programming (MILP) queuing model that guides the determination of worker threads and path options with respect to an alert latency service-level agreement.

SIP is also an important choice for IoT control due to URI addressing, extensibility through headers, and the native SUBSCRIBE/NOTIFY paradigm with scoped state carried in the headers. Recent research focuses on edge computing with SIP and analyzes the cost of security processing and server threading on latency. Millisecond budgets are possible with appropriately sized registrar workloads. As long as security and scalability issues are properly handled, this further highlights the potential of the SIP in satisfying strict time-related requirements in IoT environments (Roman et al., 2011; Mavromatis et al., 2019). In order to reduce queueing and head-of-line blocking in the presence of bursty traffic, research on the use of SIP in mobile and limited environments looks into potential solutions including message compaction, header compression, and the function of proxies. Addressing queueing and head-of-line blocking under bursty traffic is critical, potential interventions under investigations of the use of SIP in mobile and resource-constrained environments include message compaction, header compression, and the use of proxies (Rahman & Naderuzzaman, 2025).

Edge systems are increasingly leaning towards schema-based alerts to facilitate the operation of downstream automation without the need for manual intervention. CAP is used to represent the most popular and transport-agnostic form of alerts for public and industrial warnings. Strong Swan Project suggests it is a vendor-agnostic XML-based format for encoding event types, scope, confidence, and action within a single framework (Yang & Effatparvar, 2025). The CAP systems deployed in the IoT systems usually rely on HTTP polling, SMS, or topics, thereby leading to delays of several seconds (Xiong & Yu, 2025). This also results in the duplication of security measures in various systems. Several works including the need for in-band alerting in accordance with the standards and have demonstrated the benefits in terms of reduced time to mitigation for the alerts sent through existing sessions rather than relying on web hooks or batch-based systems (Lantz et al., 2010). However, previous designs have not addressed the need for zero trust onboarding, formalizing in-band revocation, and providing limited bridges to non-SIP consumers without compromising the CAP semantics (Henderson et al., 2008). This demonstrates the need for the secure and integrated dissemination of alerts through the signaling pathways of the IoT systems.

Vehicular IoT and related scenarios both manifests authentication hijacking through REGISTER spoofing attacks, nonce replay attacks, and re-INVITE storms causing proxy state exhaustion. These attacks target vulnerabilities in SIP registration and session management, allowing an adversary to masquerade as a legitimate user or device or exhaust proxies by replaying authentication information or flooding signaling proxies. Past research has shown promising results for static rule-based approaches focused on sequence-aware and drift-tolerant anomaly detectors for vehicular networks and Controller Area Network (CAN)-based in-car networks and V2X control communications, where the sequences of timing and header information are strong indicators of compromise. In these scenarios, sequence-based anomalies in the timing of messages and headers are valuable feature information for anomaly detection. Lightweight deep models, change point detection, and semi-supervised anomaly scoring have shown promising results in terms of accuracy for vehicular networks and CAN-based networks in the presence of mobility and changing link conditions, provided that inference is computationally bounded and change in the models is controlled online. However, these approaches are largely focused on the data plane of vehicular networks or CAN-based networks rather than the signaling plane and lack in-band alert propagation with strong authentication guarantees. SIP-based defenses have shown promise for Transport Layer Security (TLS) and SIP registrar-based defenses but lack standardized signed alerts and in-band

containment mechanisms such as throttling bursts of attacks or re-keying proxies (Wang et al., 2024; Tham et al., 2023; Gentile et al., 2024; Gavriilidis et al., 2025). This illustrates that despite promising results for anomaly detection in vehicular networks and CAN-based networks, a complete solution for the signaling plane of SIP-based networks with in-band alert propagation remains largely absent.

CoAP/Observe and Datagram TLS (DTLS) secure constrained links. A small number of emergency-services efforts explore embedding CAP in SIP for public safety and enterprise continuity (Shafiq, 2024). This typically assume static keys, lack zero-trust enrollment, and do not quantify millisecond-scale performance at the edge (Nandy et al., 2024). This shows that while integration is possible, current approaches lack the robustness and security model needed for real-time IoT environments. Moreover, bridging CAP from SIP to MQTT/CoAP/Web Socket without semantic loss remains under-specified: existing gateways often translate fields ad hoc or rely on application-specific schemas, undermining interoperability and machine action ability (Althunayyan et al., 2024). Finally, none of these lines of work combine in-band CAP, zero-trust onboarding, and adaptive temporal prediction under a single, deployable framework with statistical evaluation and provisioning guidance. Table I summarizes how prior art addresses authentication hijacking, alert semantics, and cross-protocol dissemination. No existing system simultaneously offers in-band hijack detection, CAP-compliant alerts, and protocol-agnostic subscribers within the stringent latency budget of edge vehicular networks. Our work fills this gap by integrating zero-trust registration, context-aware rate governance, and CAP-in-SIP encapsulation validated on a multi-access vehicular testbed with Wi-Fi 6E, LTE-A, and 5G NR links. Table 1 presents comparison of existing approaches for authentication-hijacking detection and alert dissemination within edge and vehicular IoT environments. The table highlights that several approaches achieve reasonable detection accuracy relying on centralized or cloud-based architecture which leads to increased latency and reduced reliability. Significant gap is also observed in the integration of standardized alert dissemination framework. There is lack of seamless coordination between detection and alert propagation identifying need for a unified framework which can enhance security, CAP-compliant, low-latency and responsive.

Methodology

The rate of heterogeneous IoT deployments is fast evolving. Devices with limited resources are increasingly connecting to multi access edge networks including Wi-Fi 6E, LTE-Advanced and 5G NR to meet

strict latency requirements. In this research, Trusted Authority (TA) issues short term, device specific credentials as micro-certificates, so that enrolment is on a zero-trust basis and credential reuse is not possible after their expiry period. The adversary model gives emphasis on authentication hijacking attacks, such as (i) REGISTER spoofing, which is an attack in which malicious entities inject forged registration requests to impersonate valid Vehicular Users Agents (V-UAs) (ii) re-INVITE burst storms, which is an attack where malicious agents overload the proxy by injecting excessive registration requests to exhaust the state, and (iii) nonce reuse, which is an attack where stale or intercepted values of nonce are reused to bypass authentication.

Table 1: Comparative Gap Analysis of Authentication-Hijacking Detection and Cap-Based Alert Dissemination in Edge / Vehicular IoT.

Work (Ref.)	Scope	In-band hijack Detect	CAP semantics	Carrier / Transport	Cross-protocol bridge (MQTT/CoAP /Web Socket)	Latency target (<100 ms)	Trust model /onboarding
Alzahrani (2025)	ECC auth for edge-IoT	✗	✗	HTTP/S	✗	~	PKI/ECC join only
Nandy et al. (2024)	VANET IDS (payload)	~	✗	CAN/802.11p	✗	~	Heuristic/ML
Althunayyan et al., (2024)	In-vehicle IDS	~	✗	In-vehicle bus	✗	~	Local whitelist
Yang & Effatparvar, (2025)	CAN anomaly detection	~	✗	CAN	✗	~	DL-based
Kolevski & Michael, (2024)	Edge security overview	✗	✗	—	✗	—	Survey
Gonzales et al., (2025)	NB-IoT CAP encoder	✗	✓	MQTT/NB-IoT	✗	✗	Pre-shared creds
Xiong & Yu., (2025)	HTTP/3-CAP gateway	✗	✓	HTTP/3	✗	✗	TLS only
Shafiq, (2024)	CAP in SIP (emergency)	✗	✓	SIP/UDP-TLS	✗	~	Pre-shared keys
TLS-SIP / DTLS-CoAP baselines (2023)	Secure transport	✗	✗	SIP-TLS / DTLS-CoAP	✗	~	Static PKI/PSK
CAP-SIP-Guard	Heterogeneous IoT (edge/vehicular)	✓	✓	SIP (NOTIFY/PUBLISH)	✓ (one-shot CAP digest)	✓ (edge-propagation)	Zero-trust micro-certs (IKEv2)

Transport-layer Denial-of-Service (DoS) attacks and physical device compromise are specifically not in scope and CAP-SIP-Guard is

focused on securing the signaling plane. In order to test the system, we perform controlled and repeatable experiments in three cases: (i) normal SIP signaling in order to achieve a baseline of latency and throughput, (ii) authentication-hijacking attacks to determine detection accuracy and containment efficiency and (iii) in-band alert dissemination to determine timeliness, reliability and compliance with CAP semantics. The number of Vehicular User Agents (V-UAs) used in the network deployment will be the primary parameter under investigation since it will directly affect the registration procedure, alert overhead, and signaling contention. This will allow us to assess the scalability and resilience of CAP-SIP-Guard against hijacking attacks as well as its capacity to offer secure low-latency communication in heterogeneous vehicular IoT scenarios.

CAP-SIP Guard

The proposed system architecture of CAP-SIP-Guard in Figure 1 illustrates how secure communication, authentication, and real-time alert dissemination are managed within vehicular and IoT edge networks. Vehicular SIP nodes (SIP nodes, TA, IKE module, CAP Bridge) connect over heterogeneous access to an edge SIP proxy/registrar and IoT application server. A TA issues IKEv2 micro-certificates, a CAP adapter encapsulates alerts inside SIP bodies and bridges them to MQTT, CoAP, Web Socket. An in-band hijack-detection monitor governs re-INVITE/REGISTER rates and triggers re-keying/quarantine. At startup, the V-UA automatically discovers the nearest edge SIP proxy and establishes trust by performing a TA-assisted IKEv2 handshake, which provides a lightweight micro-certificate and derives the cryptographic session keys required for secure communication.

Once these credentials are in place, the V-UA uses them to send a SIP REGISTER request over mutual TLS (mTLS) binding its verified public identity to a unique contact URI at the SIP proxy/registrar. Following registration, the V-UA subscribes (via SIP SUBSCRIBE) to specific alert topics such as authentication hijacking, certificate revocation, or policy enforcement updates. In-band hijack-detection monitor is a vital element in the system which is mounted on the SIP proxy and used to continuously search signaling behavior to discover indicative signs of malicious intent, such as credential replay, spoofed contact URI changes, skewed nonce or re-INVITE storms beyond adaptive limits. When suspicious activity is detected then the monitor immediately implements four response actions that satisfy the CAP of: (i) developing a CAP-compliant notification with its type, scope, severity, and recommended mitigation, (ii) transmits the notification in-band as SIP NOTIFY to all subscribing devices, with integrity and freshness

guarantees to the notification, (iii) updates a revocation message with the TA or distributed ledger, and (iv) requests the IKE management to rotate session key or, to quarantine/downgrade the affected Address-of-Record. Meanwhile, the CAP adapter element also has alert bridging capabilities to other IoT protocols (MQTT, CoAP, Web Socket), therefore, the non-SIP clients are also provided with the same notifications, semantically intact.

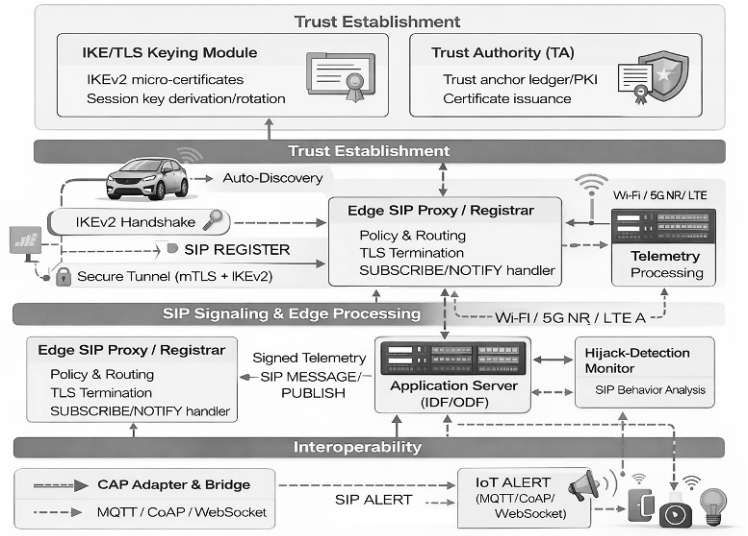

Figure 1: Proposed system architecture of CAP-SIP-Guard.

Figure 2 (above) shows the flow of messages in registration, operation, and hijack response. The sequence starts with IKE-assisted onboarding, in which the V-UA authenticates to the TA so as to receive a micro-certificate and session keys and the secure base of further signaling is established. After the credentials have been provisioned, the V-UA then makes a SIP REGISTER request to identify itself to the proxy and a SUBSCRIBE request to alert topics so that it has the capability to send and receive authenticated in-band messages. With the steady-state operation, the V-UA periodically requests the SIP PUBLISH or MESSAGE through a signed telemetry, and the IoT server requests the SIP NOTIFY updates to control and enforce the policy, and the two-way communication is maintained. In the meantime, the hijack-detection monitor is an in-band monitor running on the proxy serving to test the signaling flows, to identify such anomalies as spoofed REGISTERs, reuse of nonce, or non-conformance re-INVITE bursts.

On compromise identification, the system goes into the hijack response phase where a CAP compliant alert is synthesized and propagated

in-band to all subscribing entities through SIP NOTIFY. Meanwhile, the TA is informed of a revocation event on the credential, which guarantees that the compromised identity is nullified throughout the trust domain, and the IKE module does session re-keying, using new cryptographic material to force the V-UA to re-establish his or her identity to continue operating normally. This chain guarantees fast recovery as well as a smooth recovery and also maintains the end-to-end trust and interoperability.

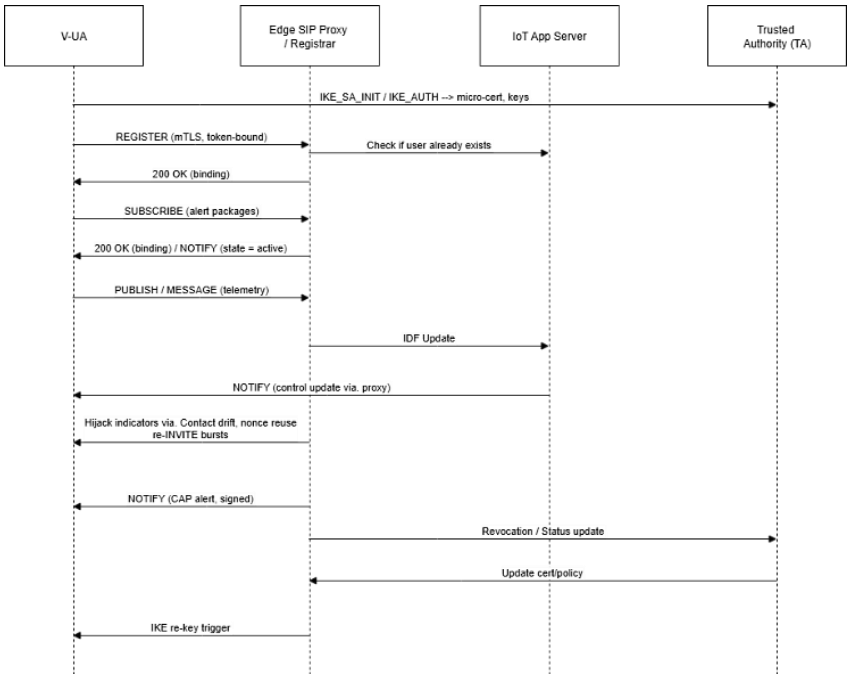

Figure 2: Registration, operation and hijack response message sequence.

Experimental Design

We vary the fleet size, $N \in \{2,4,6,8,10,12,16\}$ access technology (Wi-Fi 6E, LTE-A, 5G NR) emulated with link-profiles; and attack type (REGISTER spoofing, re-INVITE burst, nonce replay) and (iv) placement (edge registrar, cloud registrar). Each condition runs 10 trials with fixed seeds and synchronized clocks. Primary outcomes are SIP processing delay (T_{sip}), CAP alert propagation (T_{cap}), and detection quality (precision/recall/F1, false-positive rate). We decompose end-to-end alert time/latency as:

$$T_e = T_s + T_{sip} + T_{IoT} + T_{cap} \quad (1)$$

Where T_s is access/network transmission, T_{sip} proxy/registration processing, T_{IoT}application processing, and T_{cap} encode/bridge overhead.

Open SIPs acts as proxy/registrar with mutual TLS (mTLS) on the signaling path; Sip generates REGISTER/INVITE flows and scripted attacks. A hijack monitor exports header/timing features (via/Contact drift, inter-arrival gaps, nonce reuse). A context-aware rate governor throttles re-INVITE bursts at the proxy. Packet captures (tshark/ pyshark) timestamp REGISTER/INVITE, NOTIFY, and CAP deliveries. All trials use bounded jitter and loss profiles representative of the three access types.

Adaptive Hijack Prediction

Beyond rules, we add a lightweight temporal predictor over short windows of SIP-header trajectories. Features set includes method type, CSeq deltas, via chain length/entropy, Contact Adversarial Optimization/Re-planning (AoR) entropy, inter-arrival Δ_t and nonce-related irregularities. The modeling approach may employ sequence classifier, temporal Convolutional Neural Network (CNN) and Gated Recurrent Unit (GRU), change-point detector (CUSUM, Page-Hinckley, ADWIN), or semi-supervised anomaly scorer (i.e. Isolation Forest, One-Class Support Vector Machine (SVM)) leveraging temporal features (Bifet & Gavalda, 2007). Rule trigger given below defines anomaly trigger based on three independent conditions. It activated when any one of the monitored exceeds its threshold:

$$A_{rule} = 1\left[r_t > \mu_r + 3_{\sigma_r} \vee \Delta_c > \delta_c \vee v_t \geq \gamma\right] \quad (2)$$

Where r_t is re-INVITE rate, Δ_c contact through drift and v_t nonce reuse count within window τ. Fusion predictor probability $p_t \in \{0,1\}$ with rule decision:

$$s_t = \alpha p_t + (1 - \alpha)A_{rule}, \text{ Alert if } s_t \geq \theta \quad (3)$$

Drift & budgets sliding-window updates or scheduled re-fit, CPU/memory caps to fit edge devices; per-alert feature attributions to aid operator triage. Median and p95 observed for T_{sip} and T_{cap} and precision, recall, F1:

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}}, \text{Recall} = \frac{\text{TP}}{\text{TP+FN}}, \text{F1} = 2\frac{\text{Precision.Recall}}{\text{Precision+Recall}} \quad (4)$$

Confidence intervals use bootstrap (2,000 resamples) for medians; Cohen's d quantifies effect size for latency differences; Mann–Whitney U tests assess non-normal comparisons. The positive difference between the initial rule-only alert and the fused rule and ML alert is the early-warning lead time.

MILP for Alert-Path Provisioning

We define small MILP that picks the worker threads ω_s per stage i.e. $s \in \{proxy, app, bridge\}$ and selects a link profile $x_k \in \{0,1\}$ on path k to minimize latency SLA, Δ and cost:

$$\min_{w_s,x} \sum_i t_i + \lambda \sum_s c_s w_s + \eta \sum_k b_k x_k \quad (5)$$

$$\text{s.t. } t_i \geq \sum_s \alpha_{is} \frac{v_{is}}{\omega_s \mu_s} + \sum_k \beta_{ik} d_k x_k, \sum_k x_k = 1, x_k \in \{0,1\}, \omega_s \in \mathbb{Z}_{\geq 1}, t_i \leq \Delta \quad (6)$$

Per-proxy waiting time as a queueing guide:

$$W_q \approx \frac{C_s^2 + C_a^2}{2} \cdot \frac{\rho}{1-\rho} \cdot \frac{1}{\mu}, \rho = \frac{\lambda}{m\mu} \quad (7)$$

Where m threads, λ arrival rate, μ service rate, and C_s^2, C_a^2 are squared coefficients of variation.

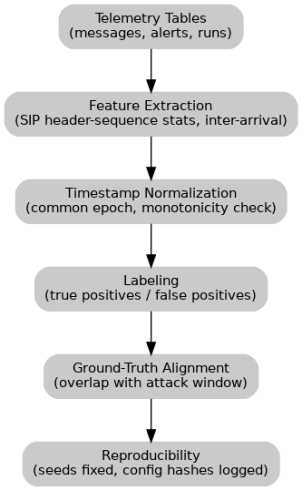

Figure 3: Data preparation and preprocessing workflow.

Dataset and Preprocessing

The data to be used in this study was assembled by combining three sources of telemetry, namely the messages table which stores the per-message SIP features and timings, the alerts table (which stores alert-timestamps, type, and detection results), and the runs table (which stores metadata about trials, including fleet size, access type, and attack location, and random seed). Preprocessing involve the feature extraction was carried out to calculate the SIP header-sequence statistics and inter-arrival time. Each of the timestamps was adjusted to a shared epoch and checked to be monotonic. Ground-truth labels were provided in a way that alert within some specified attack window (with tolerance) was considered as a true positive, and false positives were not considered as a true positive any more. In order to be reproducible, fixed random seeds were enforced on all code paths of the experiment, and hash values of configuration logged to verify. Figure 3 (above) illustrates data preparation and preprocessing procedure for creating the dataset that will be used to evaluate the CAP-

SIP-Guard's hijack detection method. This procedure guarantees that the SIP signaling data is transformed into an organized, dependable, and repeatable format.

Results

The impact of threshold variation on the detection performance of CAP-SIP-Guard compared to baseline methods compared to the baseline method, CAP-SIP-Guard performs better as shown in figure 4. The baseline technique ranges from ≈0.94 to 0.95, whereas CAP-SIP-Guard has a precision of ≈0.968 and a recall of ≈0.979 at all thresholds. The performance of CAP-SIP-Guard under various thresholds demonstrates its resilience and justifies the efficacy of combining rule-based triggers for anomaly detection in SIP signaling with temporal machine learning hijack prediction.

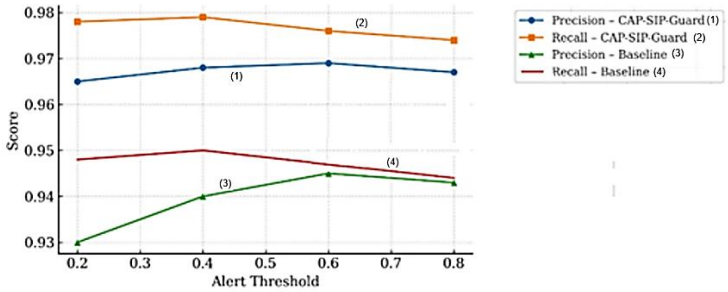

Figure 4: Precision and Recall vs Threshold.

CAP-SIP-Guard scores higher in the detection accuracy, with precision of 0.968, recall of 0.979 and F1 score of 0.974, which is a highly reliable identification of authentication-hijacking events, compared to the baseline system, which scores slightly lower as 0.94, 0.95 and 0.945.

In addition, CAP-SIP-Guard has a very low false positive of 0.00089 over 0.0032 by the baseline and minimizes false alarms, which makes it more efficient in its operations. In addition to the detection accuracy the scalability characteristics of the proposed system were examined as the size of the vehicular fleet was grown. Figure 5 illustrates the confusion matrix.

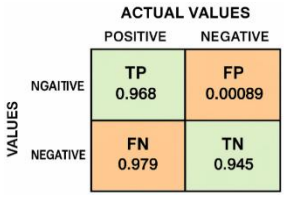

Figure 5: Confusion Matrix.

CAP-SIP-Guard has always gained accuracy and recall scores greater than 92 and F1-scores are much higher than the baseline system. Moreover, the false positive rate is less than 5% which implies that CAP-SIP-Guard does not produce many false alarms and it can detect a large percentage of alerts.

We evaluate CAP-SIP-Guard on a synthetic dataset for vehicular IoT which was produced by simulating resource counters, CAP alert messages, and per-message SIP telemetry data for Wi-Fi 6E, LTE-A, and 5G NR. A number of size N∈{2,4,6,8,10} vehicular UAs provide steady-state REGISTER refresh messages and PUBLISH /MESSAGE telemetry messages in each test scenario; spoofing REGISTER messages, nonce replay attacks, and re-INVITE bursts are used to introduce assaults. Primary metrics are SIP processing delay T_{sip} CAP delivery time, packet loss, and hijack-detection quality (precision, recall, F1, false-positive rate).

Figure 6 presents the median hijack detection time for different attack events, including spoofing, re-INVITE bursts, and nonce replay, with p95 error bars. The results show that in-band monitoring achieves detection times below 45 ms across all event types. Detection remains stable even as the number of vehicles increases, confirming the scalability and efficiency of CAP-SIP-Guard in dynamic vehicular networks. Together with the workflow shown in Figure 4, these results highlight the system's capability to deliver fast, reliable, and real-time hijack detection suitable for large-scale deployment.

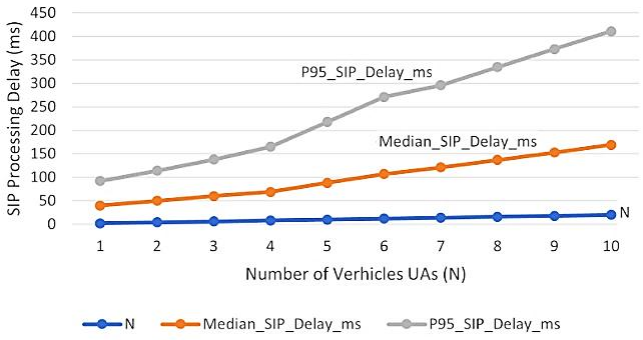

Figure 6: SIP Delay vs Number of Vehicles.

In addition to the assessment of the latency performance, the efficacy of CAP-SIP-Guard in terms of the detection efficacy and reliability is presented in figure 7. CAP-SIP-Guard detection latency is compared in terms of three authentication-hijacking attacks namely: REGISTER spoofing, re-INVITE storms, and nonce replays. CAP-SIP-Guard system

attains median detection times of 30-40 milliseconds (ms) and p95 latency of less than 50 ms in all types of attacks. Conversely, in baseline schemes, the median latencies are much larger (120-180 ms) and the p95s are as large as 280 ms. these findings validate the fact that the temporally resolved machine learning predictor used by CAP-SIP-Guard is capable of identifying SIP-based attacks effectively and in a timely manner. Therefore, the system facilitates almost real-time notifications on the edge of IoT infrastructures that indicates its efficiency in securing heterogeneous and large-scale networks. Table 2 shows the performance comparison of CAP-SIP-Guard and a traditional Non-CAP baseline system on various key measures. CAP-SIP-Guard has a median alert delivery latency of 26 ms and a p95 alert delivery latency of 40 ms significantly compared with the baseline system with 220 ms and 480 ms median and p95 alert delivery latencies respectively, showing that the system can provide almost instantaneous alerts.

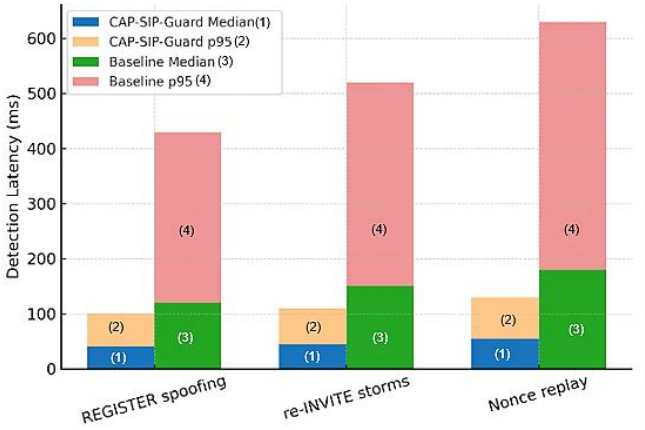

Figure 7: Attack Type Detection Latency.

Table 2: CAP vs Non-CAP Performance.

System	Alert Delivery median (ms)	Alert Delivery p95 (ms)	Precision	Recall	F1 score	False positive rate
CAP-SIP-Guard	26	40	0.968	0.979	0.974	0.00089
Non-CAP baseline	220	480	0.94	0.95	0.945	0.0032

CAP has been incorporated into SIP and has contributed greatly to the responsiveness of authentication-hijacking detection, as opposed to conventional out-of-band authentication mechanisms like HTTP polling. As shown in Table 2, CAP-in-SIP saves the median alert latency by about

8.5 times and gets 26 ms and 220 ms in the case of the base line and stand-alone respectively. Bootstrap resampling of medians provides 95% confidence interval of 24-29 ms CAP-SIP-Guard and 207-234 ms Non-CAP baselines, which indicates an impressive and significantly significant difference. Moreover, in comparison with the secured baselines such as TLS-SIP and DTLS-CoAP controls, CAP-SIP-Guard can acquire an extra ~31 percent of reduction in alert latency and at the same time, it retains a high quality of detection. Similar results indicate that authenticated throughput in the system is enhanced by around 2.4x without any statistically significant loss in accuracy or recall, which indicates that integration of CAP does not only lead to faster delivery of alerts, but also maintains reliability and accuracy in edge IoT applications that are very critical to the mission environment.

Discussion

The findings support the efficiency of the system to offer near real-time authentication-hijacking detection to ensure IoT and vehicular edge devices are warned timely regardless of the fluctuating network load and multi-mode connectivity statuses. By taking into account the quality of detection under various operating situations, it not only shows a decrease in false positives but also offers a steadier performance. The results verify that CAP-SIP-Guard has high detection accuracy and resilience in comparison and is feasible to use in the heterogeneous vehicular and edge IoT deployment that needs real-time protection of the hack of authentication in comparison with the (Althunayyan et al., 2024). We observe CAP-SIP-Guard and baseline protocols have lower CPU and memory overhead (~20-30%) less than the baseline ML-driven approaches (Alzahrani, 2025). This is consistent with the suggested protocol's lightweight design for the 5G vehicle edge network. This is crucial to guaranteeing that the suggested protocol is appropriate for diverse autonomous systems with potentially constrained resources. The provided results were obtained while synchronized clocks and jitter/loss were present.

We have three directions that we identify to increase robustness and operational readiness. Measuring handover induced jitter, NAT/ALG jitter on header normalization, CAP interoperability with municipal/public-warning backbends, and quarantine, re-key and recovery operator workflows will verify thresholds, provide policy default refinements. To scale security optimization, to go beyond the observed scalability knee, AoR sharding, proxy pools, event-driven worker pools use should be considered to SUBSCRIBE/NOTIFY fan-out, and CAP Bridge offloading. At the cryptographic level, combining IKEv2 session

resume, pre-provisioning of certificates and revocation caches may decrease the churn further when there is a burst of incidents.

Scalability is influenced by proxy queueing under higher fleet sizes, which may require distributed deployment strategies in practical scenarios. To bridge the gap between hardware-in-the-loop trials with commercial modems and vehicular gateways are required to collect field measurements of handover-induced jitter and run A/B comparisons against production out-of-band alert buses. This will further explore adaptive thresholds and hybrid detectors the ability for integrating Multi-access Edge Computing (MEC) placement and certificate-lifecycle services into the MILP sizing model will enable joint optimization of alert latency and operational cost under realistic bursty workloads. In the present investigation, the performance of the system may degrade when the fleet size increases beyond a certain threshold which can be attributed to proxy queueing phenomena. Moreover, the current implementation depends on the fact that the Open SIPS proxy framework is used. Adversaries and the emulated Wi-Fi 6E, LTE-A, and 5G NR profiles have been taken into consideration in this work. Nonetheless, a number of real-world elements have not been adequately portrayed. The accuracy and latency figures here should be interpreted as best-case under controlled conditions.

Conclusion

This paper presented state-of-the-art CAP-SIP-Guard, an enhanced SIP-based alert mechanism for heterogeneous IoT that embeds CAP objects within SIP signaling couples them with IKEv2 micro-certificates for zero-trust on boarding, and applies context-aware rate governance for in-band detection of authentication hijacking. Experiments showed that SIP processing delay remains below ~65 ms up to eight vehicles and that in-band CAP delivery achieves sub-50 ms notification across access technologies, while maintaining high detection quality (precision/recall ≈ 0.97/0.98) and very low false-positive rates. A comparative baseline confirmed that CAP-in-SIP markedly shortens alert latency relative to out-of-band mechanisms. Collectively, these results substantiate SIP's viability as a secure, real-time control plane when augmented with structured, standards-compliant alert semantics.

An extension of our MILP to p95 latency and cost under realistic fleet growth will be addressed through a joint placement/sizing study - introducing MEC locality, admission control, and rate-limit policies. Lastly, we will evaluate privacy and auditability (e.g., CAP redaction, per-tenant keys and tamper-evident logs) in order to fulfill regulatory and multi-tenant concerns. The purpose of the proposed roadmap adaptive prediction, real-world validation, scale-oriented optimization is to be able

to transfer these gains to resilient and deployable systems. In terms of its implementation, CAP-SIP-Guard can be deployed on edge SIP proxies that have relatively small computational and memory constraints, which makes it suitable to real-life 5G edge Internet-of-Things deployments.

References

Althunayyan, M., Javed, A., & Rana, O. (2024). A robust multi-stage intrusion detection system for in-vehicle network security using hierarchical federated learning. *Vehicular Communications*, 49, 100837.

Alzahrani, N. (2025). Security importance of edge-IoT ecosystem: An ECC-based authentication scheme. *PLoS one*, *20*(6), e0322131.

Aminikhanghahi, S., & Cook, D. J. (2017). A survey of methods for time series change point detection. *Knowledge and information systems*, *51*(2), 339-367.

Bai, T., Pan, C., Ren, H., Deng, Y., Elkashlan, M., & Nallanathan, A. (2021). Resource allocation for intelligent reflecting surface aided wireless powered mobile edge computing in OFDM systems. *IEEE Transactions on Wireless Communications*, *20*(8), 5389-5407.

Bairi, P., Swain, S., Bandyopadhyay, A., Aurangzeb, K., Alhussein, M., & Mallik, S. (2025). Intelligent VANET-based traffic signal control system for emergency vehicle prioritization and improved traffic management. *Egyptian Informatics Journal*, *30*, 100700.

Bhatti, D. S., Sidrat, S., Saleem, S., Malik, A. W., Suh, B., Kim, K. I., & Lee, K. C. (2024). Performance analysis: Securing SIP on multi-threaded/multi-core proxy server using public keys on Diffie–Hellman (DH) in single and multi-server queuing scenarios. *Plos one*, *19*(1), e0293626.

Bifet, A., & Gavalda, R. (2007, April). Learning from time-changing data with adaptive windowing. In *Proceedings of the 2007 SIAM international conference on data mining* (pp. 443-448). Society for Industrial and Applied Mathematics.

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM computing surveys (CSUR)*, *41*(3), 1-58.

Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM computing surveys (CSUR)*, *46*(4), 1-37.

Gavriilidis, N. O., Halkidis, S. T., & Petridou, S. (2025). Empirical Evaluation of TLS-Enhanced MQTT on IoT Devices for V2X Use Cases. *Applied Sciences*, 15(15), 8398.

Gentile, A. F., Macrì, D., Carnì, D. L., Greco, E., & Lamonaca, F. (2024). A network performance analysis of MQTT security protocols with constrained hardware in the dark net for DMS. *Applied Sciences*, 14(18), 8501.

Gonzales, P. A., Ribeiro, S., Orozco, J. S., Becerra, J. D., Vilchez, V., Mosca, E., ... & Astudillo, C. A. (2025, August). Random Access Procedure for 5G/6G Direct-to-Satellite LEO Connectivity with Limited-Capacity IoT Devices. In *2025 IEEE Colombian Conference on Communications and Computing (COLCOM)* (pp. 1-6). IEEE.

Hartke, K. (2015). Observing resources in the constrained application protocol (CoAP) (No. rfc7641).

Henderson, T. R., Lacage, M., Riley, G. F., Dowell, C., & Kopena, J. (2008). Network simulations with the ns-3 simulator. *SIGCOMM demonstration*, 14(14), 527.

Im, H., & Lee, S. (2024). TinyML-based intrusion detection system for in-vehicle network using convolutional neural network on embedded devices. *IEEE Embedded Systems Letters*, *17*(2), 67-70.

Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., & Kivinen, T. (2014). Internet key exchange protocol version 2 (IKEv2) (No. rfc7296).

Kolevski, D., & Michael, K. (2024). Edge computing and iot data breaches: Security, privacy, trust, and regulation. *IEEE Technology and Society Magazine*, *43*(1), 22-32.

Kschischang, F. R., & Sorokine, V. (2002). On the trellis structure of block codes. *IEEE Transactions on Information Theory*, *41*(6), 1924-1937.

Lantz, B., Heller, B., & McKeown, N. (2010). A network in a laptop: Rapid prototyping for software-defined networks. In *Proceedings of ACM HotNets*.

Mavromatis, A., Colman-Meixner, C., Silva, A. P., Vasilakos, X., Nejabati, R., & Simeonidou, D. (2019). A software-defined IoT device management framework for edge and cloud computing. *IEEE Internet of Things Journal*, *7*(3), 1718-1735.

Nandy, T., Noor, R. M., Kolandaisamy, R., Idris, M. Y. I., & Bhattacharyya, S. (2024). A review of security attacks and intrusion detection in the vehicular networks. *Journal of King Saud University-Computer and Information Sciences*, *36*(2), 101945.

OASIS. (2010). Common alerting protocol version 1.2. OASIS Standard

OASIS. (2019). MQTT version 5.0. OASIS Standard.

Roman, R., Najera, P., & Lopez, J. (2011). Securing the internet of things. *Computer*, *44*(9), 51-58.

Rahman, M. H., & Naderuzzaman, M. (2025). A Comprehensive Review of M2M Communication Protocols. *Open Access Journal on Engineering Applications*, *1*(01), 1-13.

Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., ... & Schooler, E. (2002). *SIP: session initiation protocol* (No. rfc3261).

Roach, A. B. (2012). RFC 6665: SIP-Specific Event Notification.

Shafiq, M., (2024). Embedding CAP in SIP for emergency services. In *Proceedings of IEEE Consumer Communications & Networking Conference (CCNC)*.

Shelby, Z., Hartke, K., & Bormann, C. (2014). The constrained application protocol (CoAP) (No. rfc7252).

Tham, C. K., Yang, L., Khanna, A., & Gera, B. (2023, June). Federated learning for anomaly detection in vehicular networks. In *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)* (pp. 1-6). IEEE.

Waidyanatha, N. (2008, November). Common alerting protocol. In *ITU Asia-Pacific Centre of Excellence Training/Workshop on Effective Use of Telecommunications/ICTs in Response to Disasters: Saving Lives, Sintok, Kedah (Malaysia)* (Vol. 25).

Wang, Y., Gao, H., Xiang, Z., & Al-Dulaimi, A. (2024). Reliable routing for V2X networks: A joint perspective of trust prediction and attack resistance. *IEEE Internet of Things Journal*, *11*(22), 36291-36307.

Xiong, H., & Yu, J. (2025, June). Adaptive AI in Smart Grid: A Continual Reinforcement Learning Framework for Cyber-Physical Systems. In *ICC 2025-IEEE International Conference on Communications* (pp. 6197-6202). IEEE.

Xu, J., Zhu, M., Liu, X., Li, X., & Yang, Y. (2025). Cost-aware heterogeneous service placement strategies for mec-based unmanned delivery. *IEEE Transactions on Network and Service Management*.

Yang, H., & Effatparvar, M. (2025). A deep learning based intrusion detection system for CAN vehicle based on combination of triple attention mechanism and GGO algorithm. *Scientific Reports*, *15*(1), 19462.