ONLINE ISSN: 3007-3197 | PRINT ISSN: 3007-3189

http://amresearchreview.com/index.php/Journal/about

Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 3, Issue 5 (2025)

Efficient Artificial Intelligence-based Constrained Application Protocol (CoAP) Protocol for IoT-Enabled Machine Learning Decision and Security System

[1*]Muhammad Atif Imtiaz, [2]Salheen Bakhet, [3]Saleha Yousaf, [4]Hira Siddique, [5]Syed Muhammad Rizwan

Article Details

Keywords: Machine Learning, Deep Neural Network, CNN, Prediction models, Internet Of Things, Threat Detection, Internet Of Things Networks, Wi-Fi Security

Muhammad Atif Imtiaz
School of Electrical, Computer and Telecommunications Engineering, University of Wollongong, NSW 2522, Australia. Department of Electronics Engineering, University of Engineering and Technology Taxila, 47050, Pakistan. Corresponding Author Email: matif@uow.edu.au
Salheen Bakhet
Department of Computer Science, University of Engineering and Technology, Lahore. salheen@ieee.org
Saleha Yousaf
The University of Lahore. saleha.yousaf567@gmail.com
Hira Siddique
School of Mathematics and Applied Statistics, University of Wollongong, NSW 2522, Australia. hira@uow.edu.au
Syed Muhammad Rizwan
Department of Computer Engineering, University of Engineering and Technology Lahore, Pakistan. rizwan.naqvi@ieee.org

ABSTRACT

The acquisition of data happens through devices with software-based internet capabilities. The Constrained Application Protocol (CoAP) provides application layer communication services for resource-constrained devices in current use. The current lightweight data protection approaches continue to experience attacks and require high-speed network links. The novel Artificial Intelligence-based (AI) hybrid lightweight data security method with CoAP enhancements functions as FPL-GLCoAP for providing security in IoT systems. The method consists of three essential functions, which include registration and authentication alongside lightweight data security. A new device obtains its credential registration as its first step when joining a network. Feister Substitution Permutation Block Cipher establishes the authentication process as part of its AI-based authentication mechanism. We have implemented the Feister Substitution Permutation Block Cipher as a lightweight AI data security mechanism that uses small-sized unique keys at each round to reduce data transmission to the cloud and preserve real-time analysis while saving bandwidth. The Light Weight Data Security (LWDS) receives evaluation through cryptanalysis that performs diffusion property tests (Galois Field multiplication, one-to-one matrix linear permutation within the diffusion layer). The main objective of incorporating Galois Field multiplication, one-to-one matrix linear permutation within the diffusion layer, is to safeguard the statistical relationships between plain text and ciphertext. Analysis confirms that proposed CoAP achieves higher accuracy together with data confidentiality, while reducing latency through minimal bandwidth utilization across the DS2OS traffic traces dataset. The proposed hybrid lightweight data security detection method provides enhanced performance results. Proposed CoAP exhibits better precision and accuracy levels of 13% and 15%, respectively as compared to CNN and ANN and SCOAP protocols..

INTRODUCTION

The lightweight packet exchange between constrained IoT devices required the development of CoAP by the IETF. CoAP emerged as an IETF-developed minimalistic application layer protocol to support IoT device communication. CoAP is based on the REST framework architecture. CoAP implements GET, POST, PUT, and DELETE operations [1, 2]. The CoAP header measures only 4 bytes in length with an optional section that includes token options and payload. An illustration of the CoAP header can be seen in Figure 2. The CoAP client requests a server through RESTful methods to receive server responses. The client uses observation by subscribing to server resources throughout a designated period, after which the server sends updates [3, 4]. The client receives updates from the server about resources only when those resources experience changes. Such methodology enables power savings during operation. The resource state requests from IoT clients can be minimized through this approach in multiple situations. CoAP operates with reliable and non-reliable transmission modes [5]. Each node transmits CON messages through reliable transmission, using which the originator demands ACK responses during communication. Non-ACK messages form the foundation of unreliable communication because receivers lack any required responses. The transmission of reliable messages will require sender confirmation when an acknowledgment response fails to arrive during the predefined time interval [6]. The sender automatically retransmits the packet through the receiver when the retransmission timeout (RTO) reaches its defined period. These retransmissions do not raise the MAX_RETRANSMIT setting because it generally uses the value 4. The representation of examples using CON and NON messages can be observed.

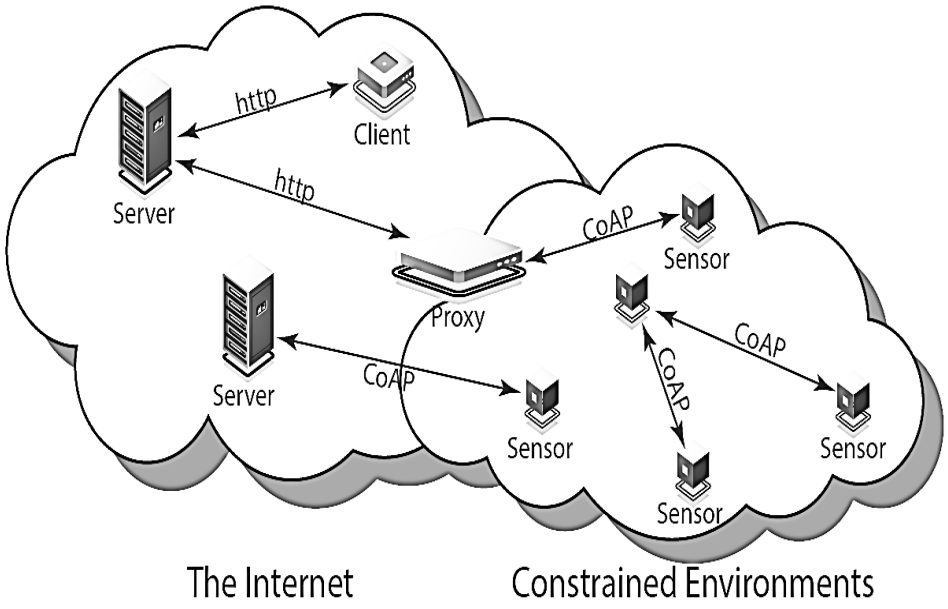


FIGURE 1: GENERALIZED COAP FRAMEWORK [7]

The researcher proposed a proxy prototype through CoAP implementation that provides direct communication compatibility between medical sensors and internet systems [8]. The system allows medical sensors to connect to the internet and interact with other nodes through the RESTful communication standard. Patient medical data automatically transfers from proxy applications on their smart devices. The medical sensor data possesses the capability of being transferred between both medical centers and doctors [9, 10].

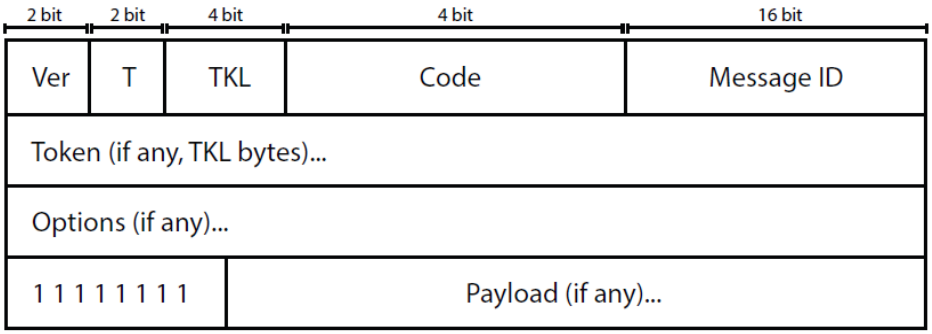
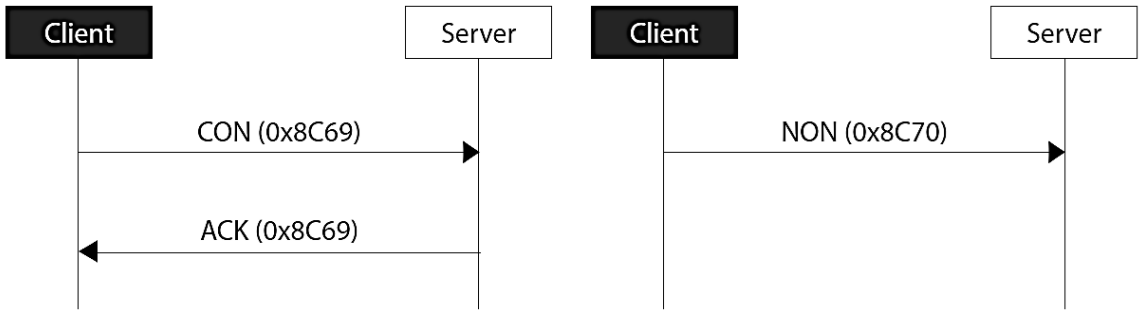


FIGURE 2: HEADER OF GENERALIZE COAP FRAMEWORK [11]

In Figure 3, the default jamming system that implements CoAP as a standard BEB limitation approach. BEB. The RTO value ranges from 2 to 3 seconds but it doubles after each timeout

expiration. The sender waits for the expiration of each retransmission cycle before reaching the total number of retransmissions.



(a) Confirmable Message (b) Non-confirmable Message

FIGURE 3: (A) CONFIRMABLE MESSAGES OF COAP FRAMEWORK (B) NON-CONFIRMABLE MESSAGES OF COAP FRAMEWORK [12]

RELATED WORK

The internet operates between client smartphones and medical workers through smartphones acting as communication proxies. The system integrates HTTP and CoAP conversion on the doctor's smartphone, which enhances compatibility with server(doctor's smartphone) functions [13]. The observed functionality within CoAP technology decreases the need for continuous server-client data transmissions. The server operates in combination with the client to obtain medical sensor information. The system functions by getting periodic system responses rather than continuous ones. This design benefits from CoAP as an IoT protocol that requires minimal computational resources [14]. This implementation operates successfully without draining excessive CPU power and memory, along with minimal power consumption. The paper by Oryema et al. in [10] builds and implements an IoT messaging solution intended for healthcare applications. The protocol works well with numerous IoT devices through dual communication and maintains a minimal system burden. The continuous request waiting period reduces the available rest time for devices, negatively impacting these restricted devices [15].

ACCESS CONTROL IN COAP

The information transfer protocol for Smart Grid devices depends on the communication protocol, OSGP cannot integrate with constrained IoT devices that use CoAP. Most IoT devices utilize communication through CoAP, but they do not incorporate this protocol with their IoT devices running CoAP. The solution named CoAP and OSGP Integration serves as a data packet

mapping between CoAP and OSGP according to [16, 17]. The mapping function operates on every request-response interaction individually. The GET method executes through the MicroCoAP library before mapping this message to continue.

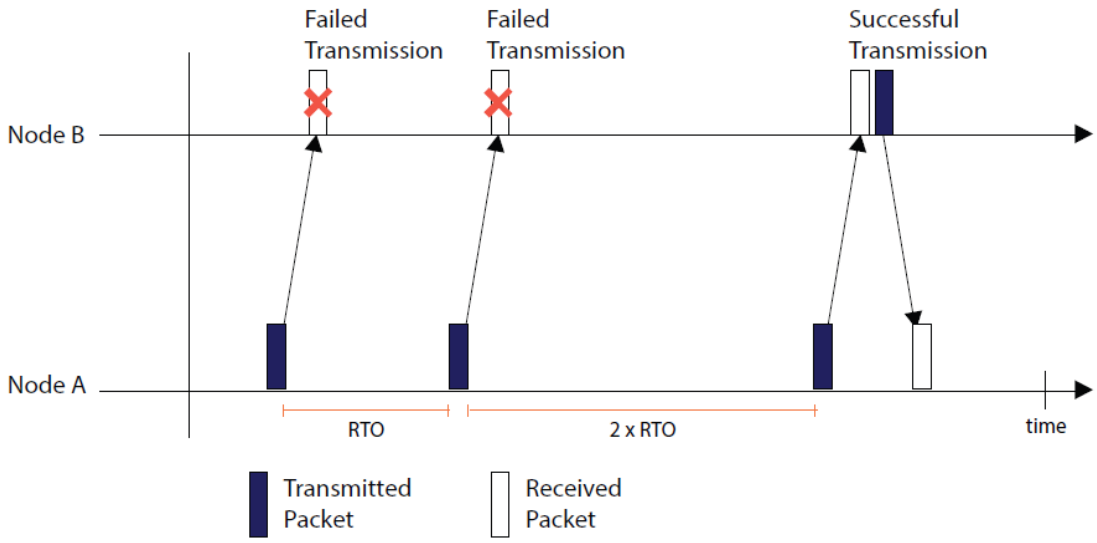


FIGURE 4: (A) CONGESTION OF MESSAGE PACKETS IN THE COAP FRAMEWORK (B) NON-CONFIRMABLE MESSAGES OF THE COAP FRAMEWORK [18]

An IoT device utilizes OSGP partial read requests for data packets received through CoAP. The program retrieves the request type alongside message ID and packet size parameters from the CoAP packets. The CoAP packets undergo inspection to determine if their content matches between CoAP requests and OSGP requests. The data packet of OSGP contains the CoAP message payload that fills its count and offset fields. The offset field receives the message contents while the message size goes into the count field. The CoAP packet contains a software field that receives its information through a count value from the OSGP packet and the data field uses values from the OSGP packet's offset [19, 20].

COAP FRAMEWORK USING LAYERS

The payload field of CoAP contains the OSGP packet values, including both offset and data fields. However, performing mapping Performing individual mapping from request/response may increase latency and overhead between nodes, so an optimization procedure is needed. Access control framework functions as a proposed system for IoT security operations [21, 22]. The authors introduce a power-efficient security design for main servers while achieving service-based access control. CoAP serves as the implementation method for communication through

packets. [23, 24].

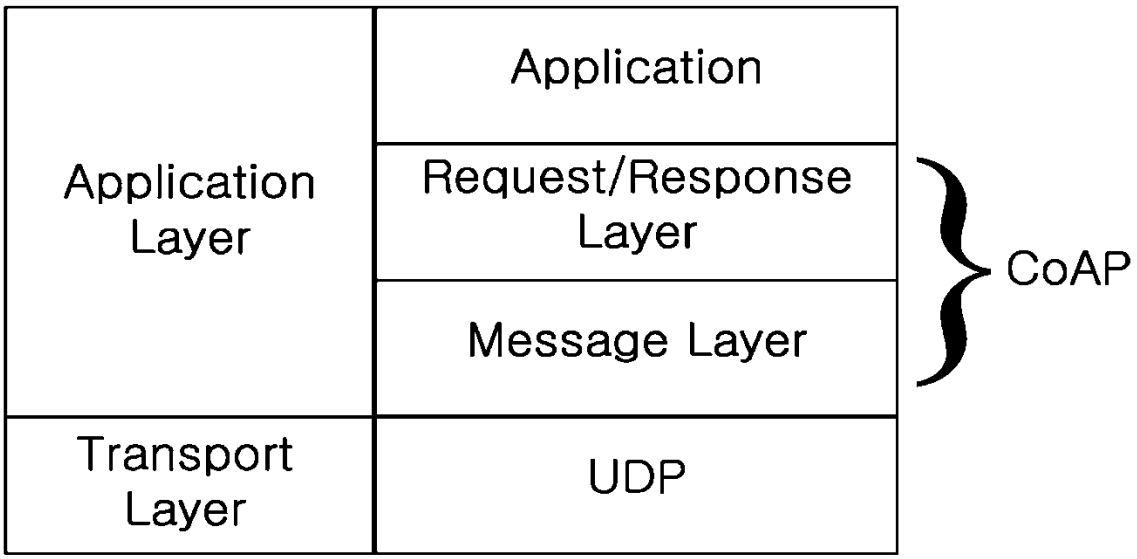


FIGURE 5: COAP FRAMEWORK BASED ON LAYERS FUNCTIONALITY [25]

The system adopts the MQTT messaging system through CoAP implementation. The system architecture divides functions into three sections, consisting of publisher, subscriber and message broker. The proposed system implements the messaging system, which utilizes CoAP clients as subscribers and the CoAP server functions as a broker [26, 27]. The system employs the CoAP observer method as its communication approach. The implementation of MQTT subscriptions along with GET, POST, PUT and DELETE methods through MQTT occurs in this architecture. The system relies on MQTT to discover topics, register resources and distribute medical data records through measured publications [28, 29].

ENHANCEMENTS AND REMOTE MONITORING IN COAP

The authors demonstrate another utilization of CoAP to build a remote healthcare monitoring system in their study [30]. The proposed system enables real-time vital monitoring of patients through a browser display. The CoAP protocol functions in the Mozilla Firefox web browser and works as both an endpoint client and a server for sensors. A CoAP server function operates through patient body sensors that work together with the Web browser as the client. Two testing frameworks, Erbium and Copper, were applied to simulate the proposed method during implementation [31, 32]. An add-on for the Mozilla browser known as Copper serves as the second implementation, while the Erbium REST service operates on Contiki OS[33].

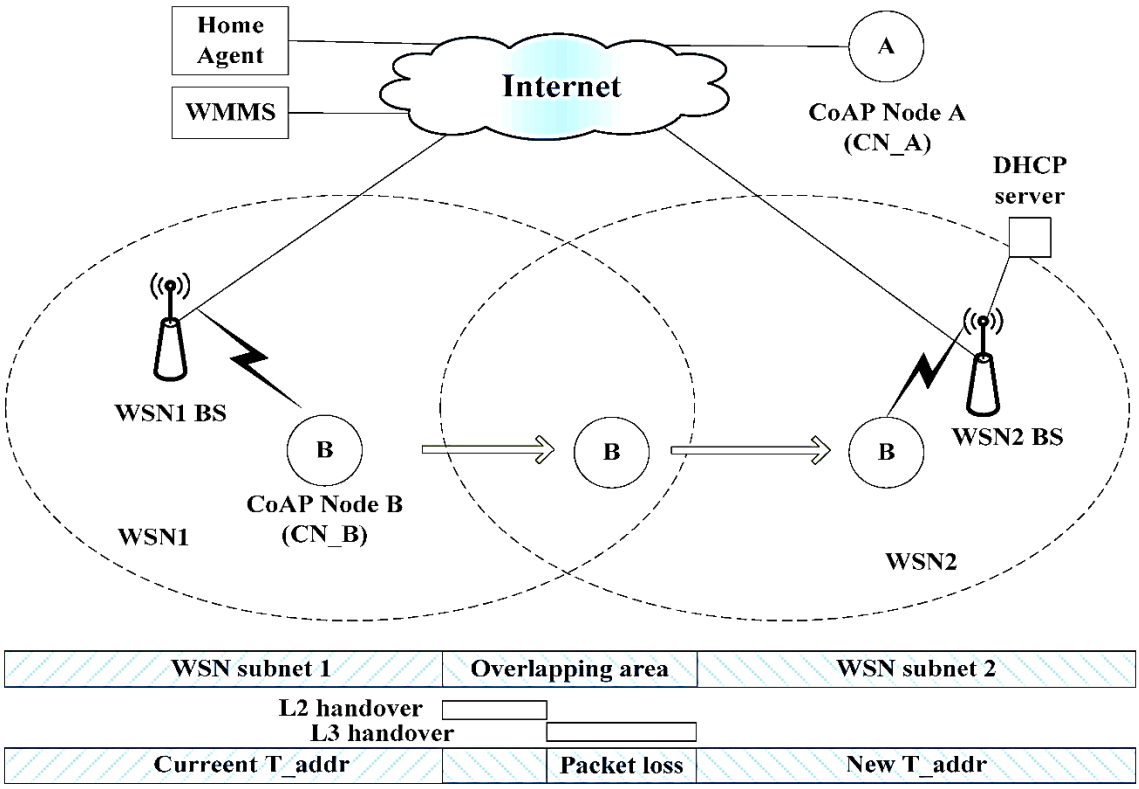


FIGURE 6: MOBILITY EVALUATION OF GENERALIZED COAP FRAMEWORK [34]

Beyond medical applications, CoAP enables various domains to use it for interoperability and protocol integration. The integration of CoAP operates with other protocols while providing streaming services along with security and resource discovery and observation features [35]. The basic structure of CoAP requires additional development to fulfill its functional requirements. For example, the basic CoAP control mechanism exhibits poor performance in dynamic network settings, along with its inefficient operation [36]. Numerous congestion control schemes have been developed by researchers to address network dynamic situations that emerged due to these conditions. CoAP could not implement cross-protocol authentication through interoperability between CoAP and HTTP. The original CoAP design operates using UDP together with DTLS for securing data transmission. The DTLS encryption system remains vulnerable to hacking attacks while also being an insecure transmission protocol. In addition, high cost and computational [37, 38]. The high power requirements of DTLS act as a major drawback to its applicability. Multiple security enhancements need to be implemented in CoAP to establish better security features. This proposal introduces security improvements for CoAP

through enhancements to the DTLS protocol operation [39, 40].

PROPOSED FRAMEWORK BASED ON COAP USING MACHINE LEARNING

The research paper addresses the key management issues found in current CoAP through static, dynamic and hybrid ML methods while developing an advanced ECC-CoAP protocol by minimizing the communication steps from the LESS protocol. A CoAP protocol emerges by enhancing the performance of the LESS trajectory protocol while ECC persistence and the new ECC-CoAP protocol develop. The LESS protocol underwent improvements, which led to a reduction in its step count. The proposed scheme fits best into resource-limited IoT environments. The system provides secure IoT device-server communication within resource-limited IoT networks with reduced communication overhead. The proposed method consists of four sequential stages:

Stage 1: Protocol initiation, Stage 2: DHCP challenge phase, Stage 3: Client Request phase, Stage 4: Authentication and DHCP response phase.

The protocol starts with a session initiation phase that is followed by server challenge, then client response and challenge, before client authentication and server response, followed by key negotiation and server authentication. The five protocol phases consist of session initiation, followed by server challenge phase and client response and challenge phase, followed by client authentication and server response phase, before key negotiation and server authentication. The detailed implementation sequence of ECC-CoAP shows how the server and client devices exchange messages. The sequence of operations between the user/IoT device and server can be observed in the following figures. Users authenticate to remote servers using their valid inputs under the presumption that the servers can be trusted. Security experts have observed occasional instances where an insider from the remote server detects malicious activities. Once an opponent acquires essential user credentials stored in the server platform, they assume the role of an adversary. Proposed ECC-CoAP implements crucial HU and DIDU storage on the server. The IoT device requires additional authentication credentials, which the server stores during the process. The proposed system requires HU and DIDU to function. The below Eq (1) shows that the input feature vector is $\boldsymbol{s}^{(\boldsymbol{b},\boldsymbol{t})}$. As the proposed COAP, $\boldsymbol{i}$ represents the random unit of $\boldsymbol{b}$ layer and $\boldsymbol{y}$ represents the total units of $\boldsymbol{b}$ layer.

$$S_i^{(b,t)} = \sum_{z=1}^{E} p_{iz}^{(b)} J_z^{(b-1,t)} + \sum_{i'}^{y} x_{ii'}^{(b)} J_{i'}^{(b,t-1)} \quad \text{Eq (1)}$$

$$J_i^{(b,t)} = \beta^{(b)}(S_i^{(b,t)}) \quad \text{Eq (2)}$$

$$J^{(b,t)} = \beta^{(b)} \times (W^{(b)} \times J^{(b-1,t)} + W^{(b)} \times J^{(b,t-1)}) \quad \text{Eq (3)}$$

STATIC THREAT RECOGNITION FRAMEWORK BASED ON COAP USING MACHINE LEARNING

Extraction of binary file structural elements combined with function calls analysis makes up static detection. Static detection serves its main purpose during the development and testing process of software applications. The current static detection procedures follow the processes illustrated in Fig. 7. Modern cybersecurity faces significant threats from APT incidents. These sophisticated attacks operate in a very stealthy manner. Researchers have studied different approaches as solutions to address these difficulties. Researchers have integrated three main approaches in their work, including machine learning (ML), deep learning (DL) and Explainable AI (XAI). A review of essential contributions made by these technologies follows.. The implementation of Decision Trees and Bayesian Networks, among other Machine learning (ML) and deep learning (DL) methods, detects known security threats effectively. The autoencoders and reinforcement learning techniques of DL offer solutions by filling the identified gaps in security detection. These systems study intricate, evolving assault methods. The autoencoder demonstrates excellence when extracting features while performing dimension reduction. Relying on deep reinforcement learning (DRL) enables systems to respond to developing malware signatures. These tools enable the identification of malware associations to particular APT group actions. Explainable AI (XAI) enhances the readability of machine learning and deep learning models in interpretation. The black box challenge receives direct attention through this methodology. Feature importance analysis can be displayed through LIME and SHAP techniques. XAI techniques enable analysts to develop confidence in AI decision outputs. High-demand situations require transparent insight systems for gaining practical conclusions. XAI frameworks tend to boost both system complexity and operational expenses. Achieving better cybersecurity requires the elimination of these gaps. APT evolution becomes easier to combat

through the implementation of these systems. The real-time threat identification capability represents a strong suit of ML systems.

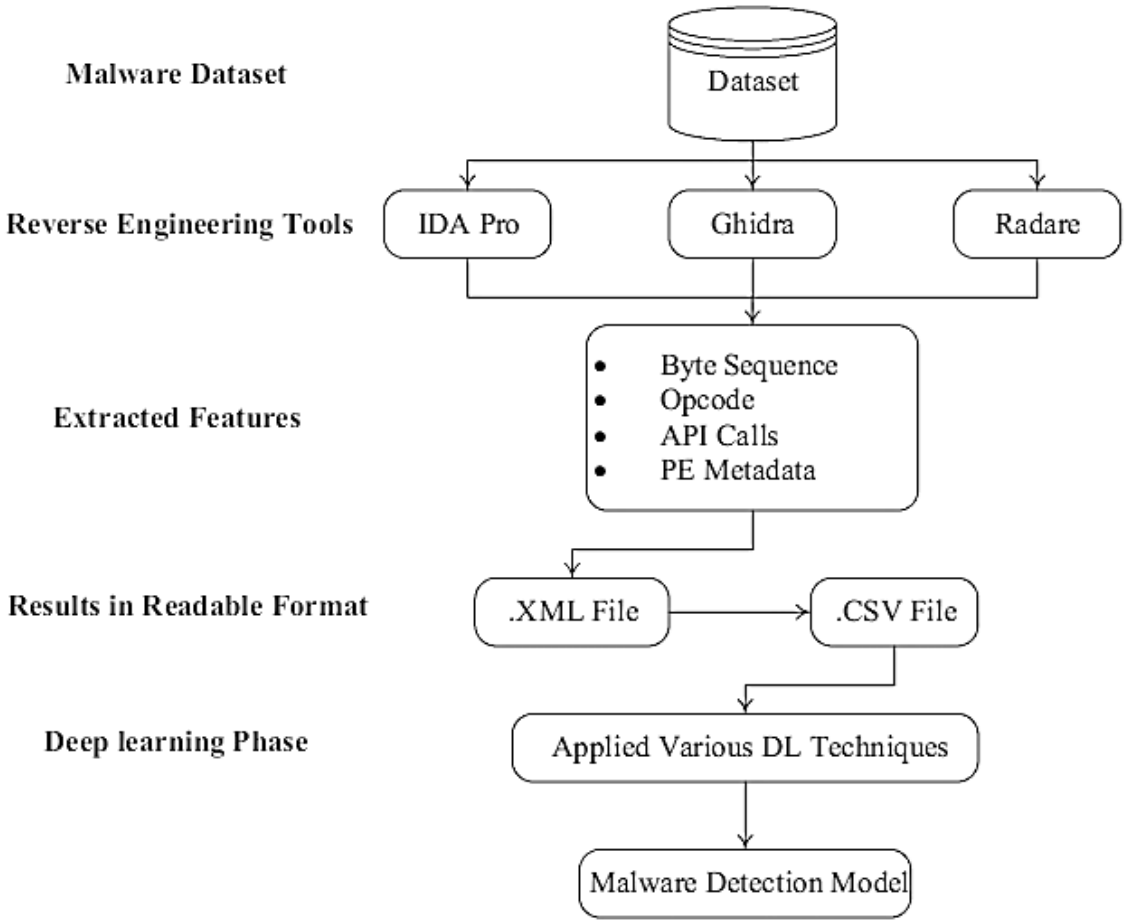


FIGURE 7: MACHINE LEARNING BASED STATIC INTRUSION DETECTION FOR IOTS USING COAP

DYNAMIC FRAMEWORK BASED ON COAP USING MACHINE LEARNING

Dynamic detection lets analysts operate potentially harmful software files within restricted virtual machine emulators and sandboxes. Analyzing resource consumption within the sample provides indications about the behavioral patterns of malicious programs. The detection technique delivers its primary utility during maintenance operations of running software programs. The dynamic detection method has its core operational steps shown in Figure 8 as part of the current detection methodology.

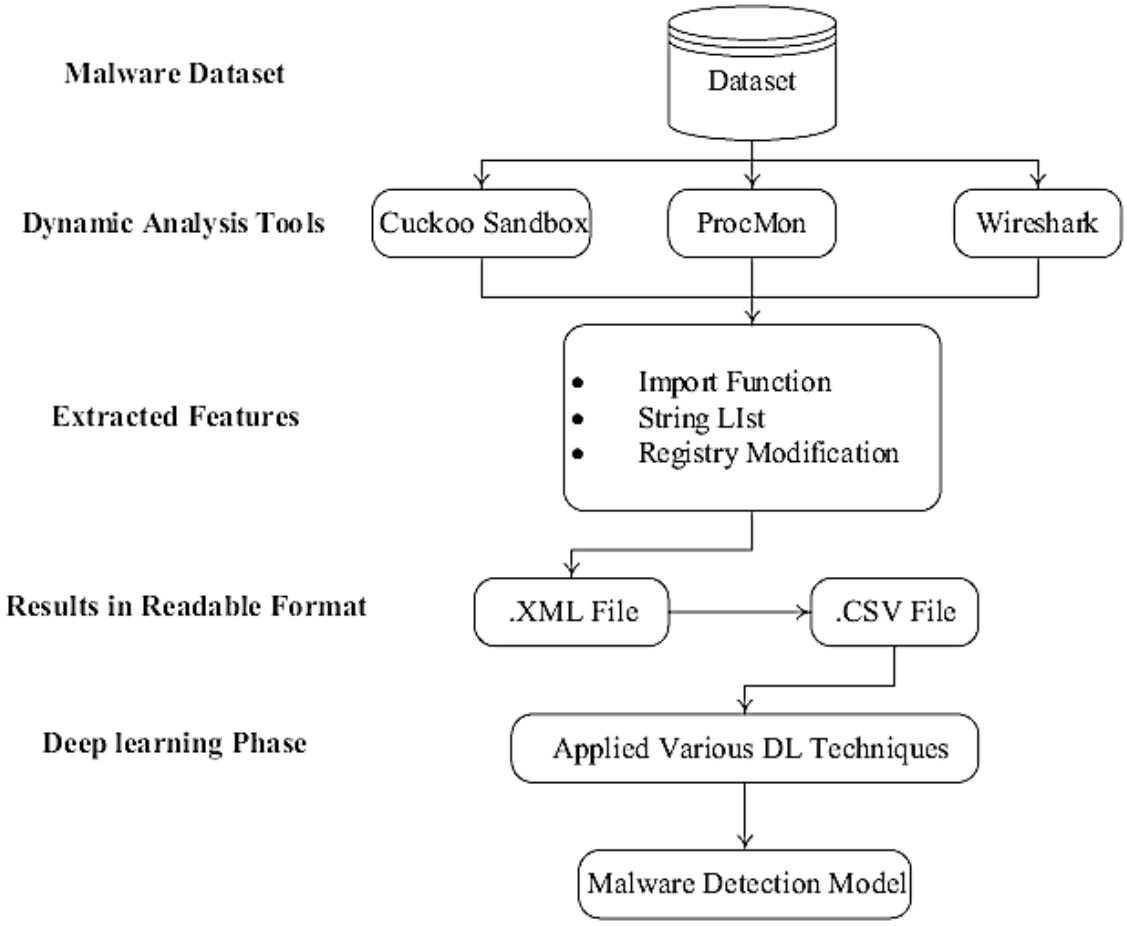


FIGURE 8: MACHINE LEARNING-BASED DYNAMIC INTRUSION DETECTION FOR IOT'S USING COAP

HYBRID MALWARE DETECTION TECHNIQUES BASED ON DL

Hybrid analysis detection of shell malware occurs at an extremely challenging difficulty level. The processes of dynamic analysis give researchers essential knowledge regarding hidden functions of code during emulation operations. The future potential exists in this technique even though it produces functional limits and security weaknesses. The team created a detection system that combined features to address previous system flaws.

$$\ln f l_{it}^{+} = \Sigma_{j=0}^{t} \Delta \ln w^{T} x + \quad b_{it}^{+} = \Sigma_{j=0}^{t} \max(\Delta w^{T}{}_{ij,0}) + \epsilon_{it} \qquad \textbf{Eq (4)}$$

$$\ln f l_{it}^{+} = \Sigma_{j=1}^{t} \Delta \ln w^{T} x + \quad b_{it}^{+} = \Sigma_{j=1}^{t} \max(\Delta w^{T}{}_{ij,1}) + \epsilon_{it} \qquad \textbf{Eq (5)}$$

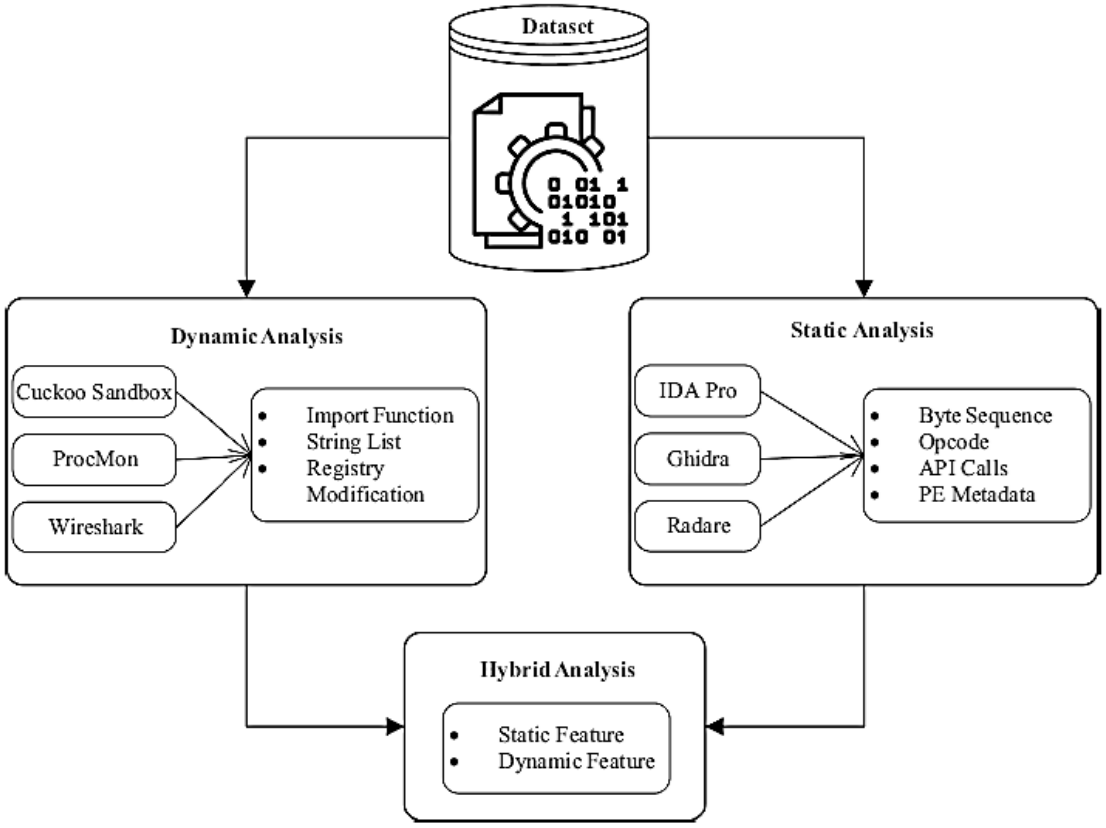


FIGURE 9: MACHINE LEARNING-BASED HYBRID INTRUSION DETECTION FOR IOTS USING COAP

SIMULATION TESTING

OpenSSL Software Foundation operates as one of the most experienced and feature-complete cryptographic libraries available in existence today. TLS and DTLS protocols with certificate handling services form part of the extensive collection of features, along with other functionalities in this framework. Parts of the code foundation exist in their original form. In theory, this library operates at the maximum speed because its core components use direct assembly implementation. The library was designed for desktop computer installation. The devices used for this project do not benefit from optimal requirements designed for limited resources. The license used by OpenSSL failed to comply with GPL projects until GnuTLS created a library to resolve this issue. It provides almost every functionality found in OpenSSL, together with the capability to operate in assembly. The performance speed of GnuTLS falls below that of OpenSSL since it does not receive the same level of optimization. The library supports TLS and DTLS versions combined with certificates and serves desktop computer

devices. Table 1 represents the XML Analysis of ML-IDS using the LRDDoS Dataset (CoAP) Using Multiple Nodes as mentioned below.

TABLE 1: XML ANALYSIS OF ML-IDS-COAP USING LRDDOS DATASET (COAP) USING MULTIPLE NODES

Data Set	Library Fold 1	Server Fold 2	Client Fold 3	H4-CoAP Fold 4	H5-CoAP Fold 5	Node1-CoAP Fold 6	Node2-CoAP Fold 7	Node3-CoAP Fold 8	Node4-CoAP Fold 9	Node2-CoAP Fold 10
LRDDoS Dataset (CoAP)	196,23	206,12	211,01	200,25	196,23	181,33	300,12	156,13	220,3	202,21
	196,23	181,33	300,12	181,33	196,23	211,01	200,25	196,23	216.71	214.11
	196,23	206,12	211,01	200,25	196,23	211,01	200,25	196,23	519.89	217.92
	216,15	206,12	216,15	206,12	211,01	211,01	200,25	196,23	911.34	60.21
	216,15	196,23	181,33	300,12	156,13	196,23	181,33	196,23	217.11	213.68
	89.65	196,23	211,01	200,25	196,23	196,23	211,01	156,13	220,3	202,21
	196,23	181,33	300,12	156,13	196,23	181,33	300,12	196,23	216.71	214.11
	196,23	211,01	200,25	196,23	196,23	211,01	200,25	196,23	519.89	217.92
	196,23	211,01	200,25	196,23	196,23	211,01	200,25	196,23	911.34	60.21
	211,01	211,01	200,25	196,23	211,01	211,01	200,25	196,23	217.11	213.68
	156,13	196,23	181,33	196,23	156,13	196,23	181,33	181,33	300,12	317.97
	196,23	181,33	300,12	156,13	196,23	181,33	300,12	156,13	220,3	202,21

Below mentioned Table 2 shows the Comparative Analysis of ML-IDS-CoAP -CoAP using LRDDoS Dataset (CoAP) with Multiple Classifiers while Table 3 represents the Comparative Analysis of Numerous Parameters Active Attack IDS for IoTs.

TABLE 2: COMPARATIVE ANALYSIS ML-IDS-COAP USING LRDDOS DATASET (COAP) WITH MULTIPLE CLASSIFIERS

Classifier	Data Set	Congestion Time	Delay	RAM Time	H4-CoAP Fold 4	H5-CoAP Fold 5	Node1-CoAP Fold 6	Node2-CoAP Fold 7	Node3-CoAP Fold 8	Node4-CoAP Fold 9	H6-CoAP Fold 10	H7-CoAP Fold 11
RNN	LRDDoS Dataset	0.011	62.19	3211	56.71	62.19	90.18	61.28	62.19	90.18	56.71	57.92
DT		0.041	57.34	3244	59.89	57.34	62.19	52.34	57.34	62.19	59.89	60.21

NBB	0.121	57.78	3364	54.32	57.78	57.34	54.32	57.78	57.34	54.32	53.68
RF	0.034	62.19	3057	52.34	57.34	62.19	52.34	57.34	62.19	56.71	57.92
SCOAP	0.025	62.19	3158	54.32	57.78	57.34	54.32	57.78	57.34	59.89	60.21
ML-IDS	0.018	39.65	3021	53.37	62.19	90.18	53.37	62.19	90.18	54.32	53.68

$$\ln fl_{it}^{+} = \Sigma_{j=2}^{t} \Delta \ln W^{T} x + \quad b_{it}^{+} = \Sigma_{j=2}^{t} \max(\Delta W^{T}{}_{ij,2}) + \epsilon_{it}$$

Eq (6)

Table 3 shows the Comparative Analysis of Numerous Parameters based Active Attack IDS for IoTs, while Table 4 represents the Comparative Analysis of Client Server using LRDDoS Dataset (CoAP) Using Multiple Nodes. On the other hand, Figure 10 represents the Latency packet loss using CoAP.

$$B_{m,n}(q+1)(1 - \frac{1 - X(0,1) - X(-1,1)}{1 - c_{m,n} \times f_{mn}(q)}) = X(0,1) \times R_{s,n}$$

Eq (7)

TABLE 3: COMPARATIVE ANALYSIS OF NUMEROUS PARAMETERS BASED ON DOS ATTACK

Attack	Parameters	ANN	RNN	CNN-LSTM	DT	CNN	SCOAP	Proposed DL-IDS
DOS Attack	Loss	0.343	0.331	0.3411	0.3411	0.3411	0.343	0.511
	Specificity	0.132	0.3411	0.531	0.531	0.531	0.132	0.331
	Accuracy	0.453	0.531	0.453	0.531	0.453	0.453	0.3411
	F-1 Score	0.541	0.121	0.541	0.3411	0.3411	0.541	0.531
	R^2 Score	0.343	0.331	0.343	0.531	0.531	0.343	0.121
	Sensitivity	0.132	0.3411	0.3411	0.121	0.121	0.132	0.531

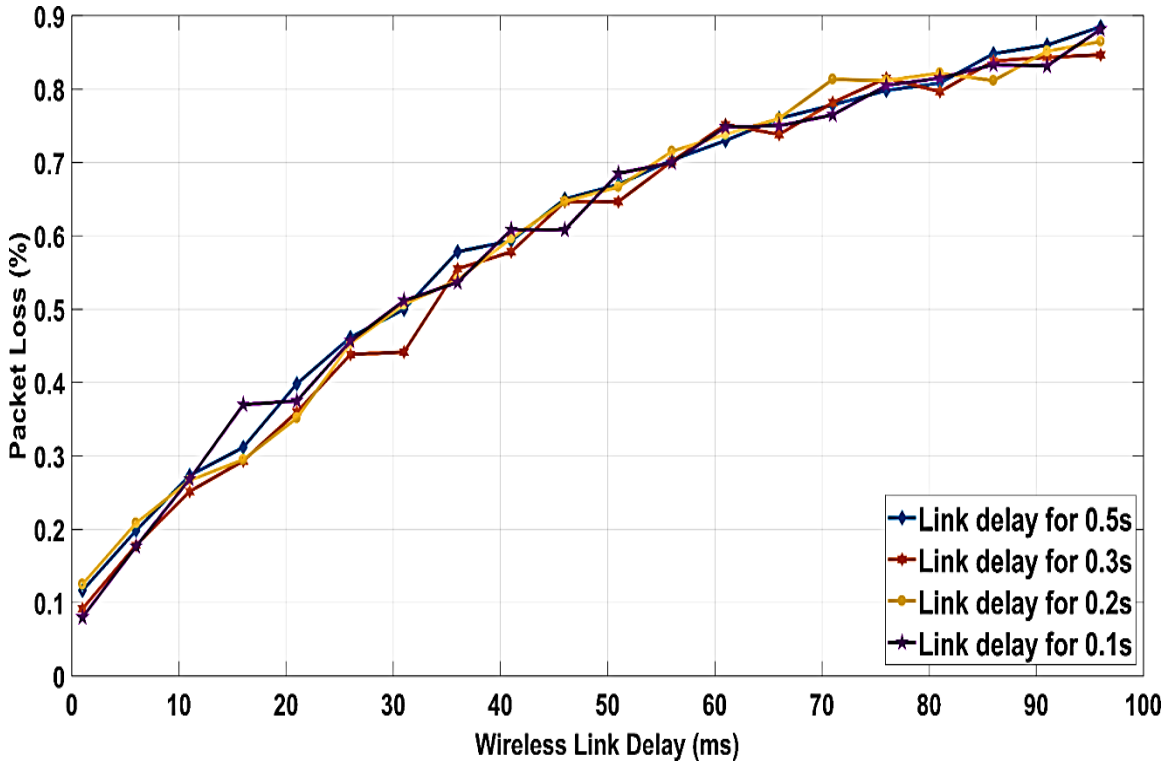


FIGURE 10: LATENCY PACKET LOSS USING COAP

TABLE 4: COMPARATIVE ANALYSIS OF CLIENT SERVER USING LRDDOS DATASET (COAP) USING MULTIPLE NODES

Data set			Server (Folds)	Mean (M)	Standard deviation (Sd)	Latency	Frame Drop (T)	Data Rate (bps)	Average Throughput
LRDDoS Dataset (CoAP)	AI awareness (AIA)	Intention to use virtual influencers	Server H1-CoAP Fold 1	3.198	2.581	3.581	0.3411	0.64 bps	3.916
			Server	2.118	3.1	3.198	0.5431	0.61 bps	1.5

	H2-CoAP Fold 1 Server	3.4	2.581	1.41	0.6321	0.72 bps	1.1
	H3-CoAP Fold 1 Server	3.198	3.5	3.198	0.7531	0.83 bps	1.51
	H3-CoAP Fold 1 Server	1.41	5.1	2.1	0.8451	0.97 bps	3.1
	H4-CoAP Fold 1 Server	1.51	1.21	3.1	0.6751	0.86 bps	2.51
	H5-CoAP Fold 1 Server	2.41	5.1	3.581	0.3411	0.64 bps	3.916
	H1-						

	CoAP Fold 2 Server	6.55	4.21	4.3	0.5431	0.61 bps	5.5
	H2-CoAP Fold 2 Server	7.11	5.1	3.581	0.6321	0.72 bps	5.1
	H3-CoAP Fold 2 Server	3.8	3.198	2.581	0.7531	0.83 bps	1.51
	H4-CoAP Fold 2 Server	4.1	2.118	3.1	0.8451	0.97 bps	3.1
	H5-CoAP Fold 2 Server	4.128	3.4	2.581	0.6751	0.86 bps	1.51
	H1-CoAP						

	Fold 3						
	Server	5.7	5.1	3.1	0.3411	0.64 bps	7.11
	H2-CoAP Fold 3						
	Server	6.55	4.21	4.3	0.5431	0.61 bps	3.8
	H2-CoAP Fold 3						
	Server	7.11	5.1	3.581	0.6321	0.72 bps	4.1
	H3-CoAP Fold 3						
	Server	3.8	3.198	2.581	0.7531	0.83 bps	4.128
	H4-CoAP Fold 3						

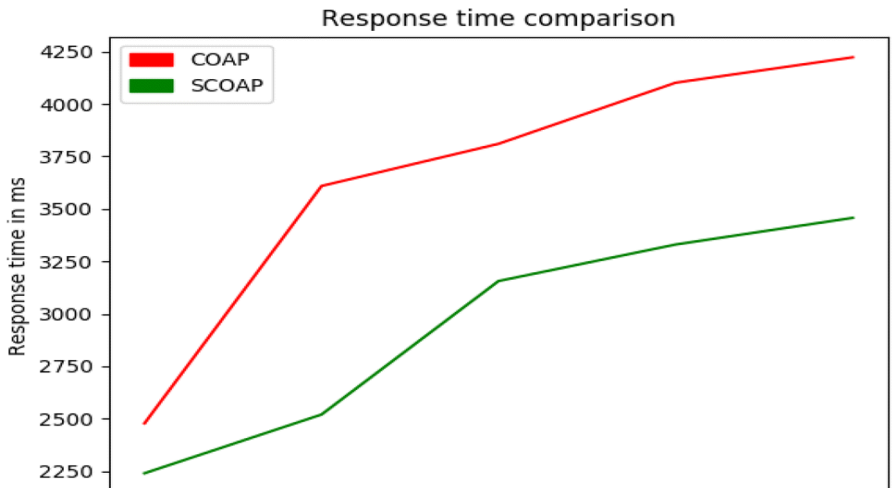


FIGURE 11: RESPONSE TIME COMPARISON OF COAP AND SCOAP

Figure 11 shows the response time (RT) for COAP and SCOAP. RT is the time instant between the initial RD requests to the last response (R) by the client. Proposed IDS-COAP (RT) is 18% higher than regular SCOAP.

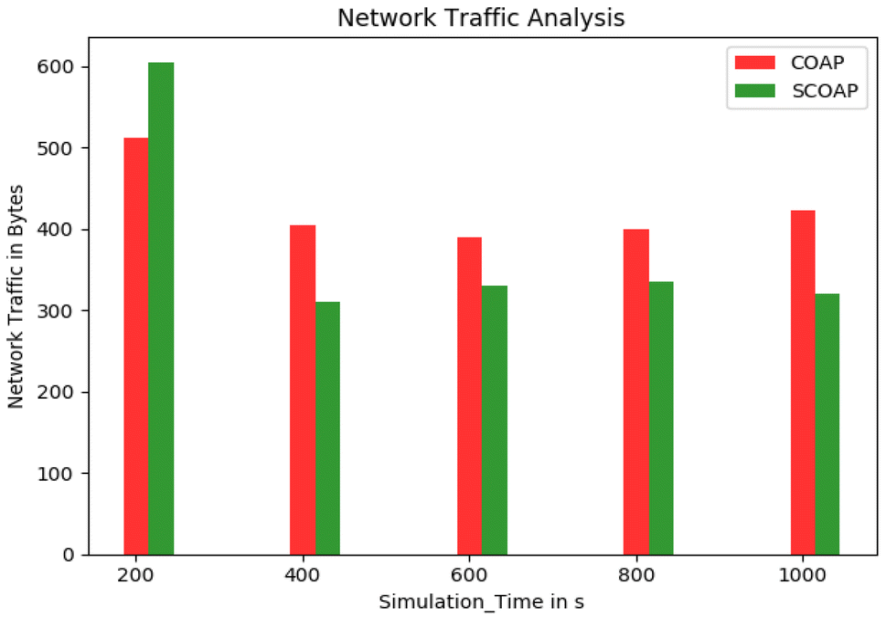


FIGURE 12: NETWORK TRAFFIC COMPARISON

CONCLUSION AND RECOMMENDATIONS

This research has presented the design and development mechanism and framework, enhanced CoAP protocol for IoT security. This paper implements the CoAP protocol based on Machine Learning with IoT techniques to establish request-response exchanges between devices using UDP protocols. The CoAP design approach specifically supports sensors in such low-power environments that run their operations from constrained resources. The user protocol establishes itself as a fundamental control system that provides reliability protection for CoAP framework operations. At low packet loss rates, CoAP deals with reduced data amounts. The capability of CoAP to handle larger amounts of transmitted data increases as the message size expands. Both signal strength analysis and data volume assessment confirm that the mobile user agent can

process large amounts of data and recognize devices through its capabilities. Packet data loss reaches 1.0% when the CoAP protocols maintain continuous data performance at 99%. CoAP protocols enable fast data storage on the cloud server to assess IOT system performance in environmental monitoring. Implementation of IoT systems through CoAP demonstrates the future capabilities of secure next-generation network systems. We evaluated two major aspects of various CoAP implementations regarding their characteristics as well as behavioral traits. Among the examined libraries, only CoAPy does not fulfill the requirements of our evaluation. The fast speed of backbone systems allows heavyweight implementation clients because these systems have sufficient resources, but lightweight libraries serve devices with limited resources. The selection of an appropriate library for systems rests upon designers who should identify the most suitable tool from our enumerated evaluations.

FUNDING STATEMENT: The authors received no specific funding for this study.

CONFLICTS OF INTEREST: The authors declare that they have no conflicts of interest to report regarding the present study.

REFERENCES

[1] Li, W.; Jung, C.; Park, J. IoT Healthcare Communication System for IEEE 11073 PHD and IHE PCD-01 Integration Using CoAP. KSII Trans. Internet Inf. Syst. 2018, 12. [CrossRef]

[2] service for low-power wide area networks: LO-CoAP-EAP. Sensors 2017, 17, 2646. [CrossRef]

[3] Tamboli, M.B.; Dambawade, D. Secure and efficient CoAP based authentication and access control for Internet of Things (IoT). In Proceedings of the 2016 IEEE International Conference on Recent Trends in Electronics,

[4] Information Communication Technology (RTEICT), Bangalore, India, 20–21 May 2016; pp. 1245–1250.

[5] Krawiec, P.; Sosnowski, M.; Batalla, J.M.; Mavromoustakis, C.X.; Mastorakis, G. DASCo: Dynamic adaptive streaming over CoAP. Multimed. Tools Appl. 2018, 77, 4641–4660. [CrossRef]

[6] Rahman, W.U.; Choi, Y.; Chung, K. Quality Adaptation Algorithm for Streaming over CoAP. In Proceedings of the 2018 International Conference on Information and Communication Technology Convergence (ICTC),

[7] Guo, X., Aviles, G., Liu, Y., Tian, R., Unger, B. A., Lin, Y. H. T., & Kampmann, M. (2020).

Mitochondrial stress is relayed to the cytosol by an OMA1–DELE1–HRI pathway. Nature, 579(7799), 427–432.

[8] Jindal, A., Aujla, G. S., & Kumar, N. (2019). SURVIVOR: A blockchain-based edge-as-a-service framework for secure energy trading in SDN-enabled vehicle-to-grid environment. Computer Networks, 153, 36–48. Khoramshahi, M., & Billard, A. (2019). A dynamical system approach to task adaptation in physical human-robot interaction. Autonomous Robots, 43(4), 927–946.

[9] Lin, K., Li, Y., Sun, J., Zhou, D., & Zhang, Q. (2020). Multi-sensor fusion for a body sensor network in a medical human-robot interaction scenario. Information Fusion, 57, 15–26. Manogaran, G., Baskar, S., Hsu, C. H., Kadry, S. N., Sundarasekar, R., Kumar, P. M., & Muthu, B. A. (2021).

[10] FDM: Fuzzy-optimized Data Management Technique for Improving Big Data Analytics. IEEE Transactions on Fuzzy Systems, 29(1), 177–185. Manogaran, G., Shakeel, P. M., Priyan, R. V., Chilamkurti, N., & Srivastava, A. (2019). Ant colony optimization-induced route optimization for enhancing the driving range of electric vehicles. International Journal of Communication Systems, e3964. https://doi.org/10.1002/dac.3964

[11] Nasimi, M., Habibi, M. A., & Schotten, H. D. (2020). arXiv preprint. Platoon–assisted vehicular cloud in VANET: vision and challenges. arXiv preprint arXiv:2008.10928. Neto, P., Sim˜ao, M., Mendes, N., & &Safeea, M. (2019).

[12] Preeth, S. S. L., Dhanalakshmi, R., & Shakeel, P. M. (2020). An intelligent approach for energy efficient trajectory design for mobile sink based IoT supported wireless sensor networks. Peer-to-Peer Networking and Applications, 13(6), 2011–2022.

[13] Priyan, M. K., & Devi, G. U. (2018). Energy-efficient node selection algorithm based on node performance index and random waypoint mobility model on the Internet of vehicles. Cluster Computing, 21(1), 213–227.

[14] Ramprasad, L., & Amudha, G. (2014, February). Spammer detection and tagging based user generated video search system—A survey. In International Conference on Information Communication and Embedded Systems (ICICES2014) (pp. 1–5).

[15] Chung, J., Gulcehre, C., Cho, K. & Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. Preprint at arXiv: 1412. 3555 (2014).

[16] Ahsan, M. M. et al. Enhancing monkeypox diagnosis and explanation through modified transfer learning, vision

[17] H. Khan, I. Uddin, A. Ali, M. Husain, "An Optimal DPM Based Energy-Aware Task Scheduling for Performance Enhancement in Embedded MPSoC", Computers, Materials & Continua., vol. 74, no. 1, pp. 2097-2113, Sep. 2023

[18] Khan, S., Ullah, I., Khan, H., Rahman, F. U., Rahman, M. U., Saleem, M. A., ... & Ullah, A. (2024). Green synthesis of AgNPs from leaves extract of Saliva Sclarea, their characterization, antibacterial activity, and catalytic reduction ability. Zeitschrift für Physikalische Chemie, 238(5), 931-947.

[19] U. Hashmi, S. A. ZeeshanNajam, "Thermal-Aware Real-Time Task Schedulabilty test for Energy and Power System Optimization using Homogeneous Cache Hierarchy of Multi-core Systems", Journal of Mechanics of Continua and Mathematical Sciences., vol. 14, no. 4, pp. 442-452, Mar. 2023

[20] Y. A. Khan, F. Khan, H. Khan, S. Ahmed, M. Ahmad, "Design and Analysis of Maximum Power Point Tracking (MPPT) Controller for PV System", Journal of Mechanics of Continua and Mathematical Sciences., vol. 14, no. 1, pp. 276-288, May. 2019

[21] Ali, M., Khan, H., Rana, M. T. A., Ali, A., Baig, M. Z., Rehman, S. U., & Alsaawy, Y. (2024). A Machine Learning Approach to Reduce Latency in Edge Computing for IoT Devices. Engineering, Technology & Applied Science Research, 14(5), 16751-16756.

[22] Nasir, M. S., Khan, H., Qureshi, A., Rafiq, A., & Rasheed, T. (2024). Ethical Aspects In Cyber Security Maintaining Data Integrity and Protection: A Review. Spectrum of engineering sciences, 2(3), 420-454.

[23] Khan, A. Ali, S. Alshmrany, "Enery-Efficient Scheduling Based on Task Migration Policy Using DPM for Homogeneous MPSoCs", Computers, Materials & Continua., vol. 74, no. 1, pp. 965-981, Apr. 2023

[24] Abdullah, M., Khan, H., Shafqat, A., Daniyal, M., Bilal, M., & Anas, M. (2024). Internet of Things (IoT's) in Agriculture: Unexplored Opportunities in Cross–Platform. Spectrum of engineering sciences, 2(4), 57-84.

[25] Y. A. Khan, M. Ibrahim, M. Ali, H. Khan, E. Mustafa, "Cost Benefit Based Analytical Study of Automatic Meter Reading (AMR) and Blind Meter Reading (BMR) used by PESCO (WAPDA)", In 2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), IEEE., pp. 1-7, Aug. 2020

[26] Y. A. Khan, "A high state of modular transistor on a 105 kW HVPS for X-rays tomography Applications", Sukkur IBA Journal of Emerging Technologies., vol. 2, no. 2, pp.

1-6, Jun. 2019

[27] Khan, S. Ahmad, N. Saleem, M. U. Hashmi, Q. Bashir, "Scheduling Based Dynamic Power Management Technique for offline Optimization of Energy in Multi Core Processors", Int. J. Sci. Eng. Res., vol. 9, no. 12, pp. 6-10, Dec. 2018

[28] H. Khan, M. U. Hashmi, Z. Khan, R. Ahmad, A. Saleem, "Performance Evaluation for Secure DES-Algorithm Based Authentication & Counter Measures for Internet Mobile Host Protocol", IJCSNS Int. J. Comput. Sci. Netw. Secur., vol. 18, no. 12, pp. 181-185, July. 2018

[29] Y. A. Khan, "Enhancing Energy Efficiency in Temperature Controlled Dynamic Scheduling Technique for Multi Processing System on Chip", Sukkur IBA Journal of Emerging Technologies., vol. 2, no. 2, pp. 46-53, Jan. 2019

[30] Khan, K. Janjua, A. Sikandar, M. W. Qazi, Z. Hameed, "An Efficient Scheduling based cloud computing technique using virtual Machine Resource Allocation for efficient resource utilization of Servers", In 2020 International Conference on Engineering and Emerging Technologies (ICEET), IEEE., pp. 1-7, Apr. 2020

[31] Rahman, M. U., Khan, S., Khan, H., Ali, A., & Sarwar, F. (2024). Computational chemistry unveiled: a critical analysis of theoretical coordination chemistry and nanostructured materials. Chemical Product and Process Modeling, 19(4), 473-515.

[32] Naz, H. Khan, I. Ud Din, A. Ali, and M. Husain, "An Efficient Optimization System for Early Breast Cancer Diagnosis based on Internet of Medical Things and Deep Learning", Eng. Technol. Appl. Sci. Res., vol. 14, no. 4, pp. 15957–15962, Aug. 2024

[33] Akmal, I., Khan, H., Khushnood, A., Zulfiqar, F., & Shahbaz, E. (2024). An Efficient Artificial Intelligence (Al) and Blockchain-Based Security Strategies for Enhancing the Protection of Low-Power loT Devices in 5G Networks. Spectrum of engineering siences, 2(3), 528-586.

[34] H. Khan, M. U. Hashmi, Z. Khan, R. Ahmad, "Offline Earliest Deadline first Scheduling based Technique for Optimization of Energy using STORM in Homogeneous Multi-core Systems", IJCSNS Int. J. Comput. Sci. Netw. Secur., vol. 18, no. 12, pp. 125-130, Dec. 201

[35] Waleed, A. Ali, S. Tariq, G. Mustafa, H. Sarwar, S. Saif, I. Uddin, "An Efficient Artificial Intelligence (AI) and Internet of Things (IoT's) Based MEAN Stack Technology Applications", Bulletin of Business and Economics (BBE)., vol. 13, no. 2, pp. 200-206, July. 2024

[36] Shah, S. Ahmed, K. Saeed, M. Junaid, H. Khan, "Penetration testing active reconnaissance

phase–optimized port scanning with nmap tool", In 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), IEEE., pp. 1-6, Nov. 2019

[37] Y. A. Khan, "A GSM based Resource Allocation technique to control Autonomous Robotic Glove for Spinal Cord Implant paralysed Patients using Flex Sensors", Sukkur IBA Journal of Emerging Technologies., vol. 3, no. 2, pp. 13-23, Feb. 2020

[38] Hassan, H. Khan, I. Uddin, A. Sajid, "Optimal Emerging trends of Deep Learning Technique for Detection based on Convolutional Neural Network", Bulletin of Business and Economics (BBE)., vol. 12, no. 4, pp. 264-273, Nov. 2023

[39] Khan, A. Yasmeen, S. Jan, U. Hashmi, "Enhanced Resource Leveling Indynamic Power Management Techniqueof Improvement In Performance For Multi-Core Processors" ,Journal of Mechanics of Continua and Mathematical Sciences., vol. 6, no. 14, pp 956-972, Sep. 2019

[40] Javed, M. A., Anjum, M., Ahmed, H. A., Ali, A., Shahzad, H. M., Khan, H., & Alshahrani, A. M. (2024). Leveraging Convolutional Neural Network (CNN)-based Auto Encoders for Enhanced Anomaly Detection in High-Dimensional Datasets. Engineering, Technology & Applied Science Research, 14(6), 17894-17899.