Cyber-XAI-Block: An End-to-End Cyber Threat Detection & FL-based Risk Assessment Framework for IoT Enabled Smart Organization Using XAI and Blockchain Technologies

Omar Abboosh Hussein Gwassi, · Osman Nuri Uçan, · Enrique A. Navarro

Abstract – **Due to the changes and developments, IoT systems are vulnerable to cyberattacks, which may result in threatening incursions. One of the most common cyberattacks on IoT-enabled smart organizations is Social Engineering Attacks (SEA). The existing works detect the SEA using DL and ML methods whereas they are limited to validating the user behaviors. The major drawbacks in existing work include high false positive rates, attack reoccurrence, and increases in numerous attacks. This study's main objective is to use explainable deep learning techniques to increase cyber security in an IoT-enabled smart organization environment. In this paper, the CapsNet algorithm is used to process the collected fingerprint and blink patterns of the employee. Additionally, this study implemented the QKSCP to reduce communication channel vulnerabilities like MITM, reply attacks, etc. The DQN-A3C algorithm is used for attack detection and prevention. This research used the explainable DQN-A3C model combined with the SILT transformer for natural language explanations to assure the trustworthiness between the AI model and humans to increase the security level against social engineering attacks. To accurately categorize harmful external traffic flows, we developed Hopping IDS and IPS based on an explainable HGN model combined with SHAP and SILT explanations. We have implemented a knowledge-sharing system using FL technology between CTR and cloud servers, also known as global risk assessment, to improve the understanding of cyberattacks on global levels. The proposed model can be analyzed using the following metrics malicious traffic, detection rate, Understandability Analysis, Secure Communication Analysis, False Positive Rate, and Prevention Accuracy.**

Index Terms- **IoT, Quantum Key Secure Communication Protocol, Dual Q Network based Asynchronous Advantage Actor-Critic algorithm, Siamese Inter Lingual Transformer, Harmonized Google Net, Cyber Threat Repository.**

I. INTRODUCTION

Internet of Things (IoT) systems is complex and includes several related components. In recent years, IoT has become one of the most advanced technologies. However, since the devices represent such a large quantity of data, insufficient security safeguards are often the biggest obstacle to IoT adoption. In an IoT system with a large attack surface, it is difficult to meet security needs. As a result, to satisfy the requirement for security, solutions must adopt a holistic approach. Additionally, certain small IoT devices lack memory management units (MMUs) [1]. Businesses with extensive websites are targeted by hackers and intruders. They may attack using a variety of techniques, such as spyware, malware, worms, fraudulent logins, and viruses. Security software is necessary for organizations to protect their networks against damaging assaults and unauthorized usage [2]. One of the key technologies used to facilitate smart factories is the IoT, which links all industrial assets, including equipment and control systems, with information systems and business processes [3]. As there are more IoT devices in smart factories, securely storing, collecting, and transferring data will be the main concern. Therefore, in such cases, private, industrial, and personal data are in danger. Due to the widespread usage of the internet, in recent years, cyberattacks on political organizations, financial institutions, and energy industries have risen dramatically. [4]. As the number of linked devices continues growing, the IoT industry must be reliable for users. Cybersecurity is a methodology for preventing unwanted access to networks, devices, and data as well as for preserving the confidentiality, integrity, and availability of such data. Artificial intelligence, particularly machine learning (ML) and deep learning (DL), has been extensively used in the disciplines of cyber security, including intrusion detection, virus detection, and spam filtering [5]. eXplainable Artificial Intelligence (XAI) is attracting more and more interest from a wide range of applications due to its numerous advantages, including exceptionally transparent, reliable, and interpretable system development. The management of privacy and data protection in IoT devices, however, may not be adequate depending on how the AI models arrive at their conclusions. Particularly in the healthcare and military applications that might considerably benefit from XAI, some IoT application justifications must be made clear. IoT and its supporting technologies have the potential to depend on XAI technology, which has this guarantee. Cyber security models must be created using XAI to provide more understandable models while retaining high accuracy and enabling human users to grasp trust and manage the next generation of cyber defense systems [6].

A blockchain is a kind of distributed ledger technology (DLT) made up of an expanding collection of data blocks that are safely connected [7]. One of the developing concepts acknowledged by researchers and industry is blockchain technology, which has six key properties: decentralization, immutability, transparency, autonomy, anonymity, and open source. IoT is a similarly promising technology area where several smart applications are being created. Applications for the IoT are implemented using sensors, intelligent devices, and controllers [8]. Many industries, including cryptocurrency, healthcare, energy, digital documents, and IoT, make use of blockchain technology. The development of IoT-based applications now depends significantly on blockchain technology. The usage of blockchain in IoT will benefit both people and society. IoT integration with blockchain may enable peer-to-peer messaging, real-time data and file sharing, and autonomous communication between IoT nodes without the requirement for a centralized client-server approach. Blockchain is vulnerable to a variety of attacks while being verifiable and unchangeable. The stand-alone IoT applications have been improved by the IoT and blockchain integration, which has grown significantly [9,10].

Cybersecurity is a popular field of study that focuses on protecting networks from ambiguous online criminals. There are several different types of cyberattacks carried out by bad actors that target ordinary people's private information and cause several negative problems. Artificial intelligence (AI) is a current and potential technology that complements cyber security to intelligently fight against unknown cyberattacks. To tackle cyber threats, it is obvious that AI technology uses ML and DL technology, which can intelligently identify and forecast hazardous undiscovered attack patterns. Many IoT-enabled organizations are now experiencing severe security threats. Therefore, the danger that an organization faces from cyber assaults grows every year [11,12].

Recent research has focused on employing AI technology to protect IoT-enabled smart enterprises to address these issues. Many research projects use ML and DL-based techniques to conduct an Intrusion Detection & Prevention System (IDS/IPS). IDS/IPS now in use technically come under the category of signature- and anomaly-based methods. While the anomaly approach used DL and Deep Reinforcement Learning (DRL) techniques to identify unknown attack patterns, the signature-based methods used rule-based ML classifiers to identify trained and known assault patterns. However, although such techniques provide enhanced protection against hostile cyberattacks, they have limitations when taking into account issues with privacy and validity. Another major problem with that is the degree to which intelligent firms blindly accept the judgments of their AI models, which might result in increased false positive rates. To increase the trust between AI models and people, XAI was developed. XAI offers explanations for the decisions made by AI models. Only a small number of research studies have used XAI for cyber security, and those that have tended to explain traditional ML and DL methods like support vector machines (SVM), random forests (RF), convolutional neural networks (CNN), etc. To further justify research on XAI-based cyber security, new AI algorithms must be used [13,14].

Social engineering attacks (SEA) are one of the most common attacks against IoT-enabled smart organizations. To ambiguously get any person's or organization's confidential information, the SEA primarily manipulates human characteristics such as greed, fear, and curiosity. There are several ways that SEA occurs, including phishing (using bogus emails and URLs), vishing (using misleading phone calls), smishing (using harmful SMS), etc. Using DL and ML techniques, the current works identify and reduce SEA, however, they are only capable of verifying user behavior. We proposed an end-to-end secure research project for IoT-enabled smart organizations employing XAI, blockchain, and Federated Learning (FL) technologies after taking into account the issues currently present [15].

The main goal of this research is to use explainable deep learning techniques to increase cyber security in an IoT-enabled smart organization context. In this research, to process the fingerprint and blink patterns of the employee, the CapsNet algorithm is used. To reduce communication channel vulnerabilities QKSCP is implemented. This study used the DQN-A3C algorithm for attack detection and prevention. To accurately categorize harmful external traffic flows, we have developed Hopping IDS and IPS based on an explainable HGN model combined with SHAP and SILT explanations. To improve the global cyber-attack knowledge, we have performed a knowledge-sharing mechanism using FL technology among CTR and cloud servers.

A. Motivation and objectives

Amplifying security towards the IoT-enabled smart organization in cyberspace is still a major concern. The state-of-the-art works even though espouse AI and blockchain technology for warranting security to the IoT-enabled infrastructures whereas the secure, effective, scalable, and robust mechanism was not yet researched. Some of the major drawbacks of the state-of-the-art works are listed below,

- **Amplified Cyber Threats**- The validity and authenticity of the connected organizations were not taken into account in the majority of current works in IoT-enabled environments, which results in undesired harmful traffic and imposes many threats. On the other hand, the old methods of authentication utilized traditional metrics such as user name, ID, password, etc., which were quite vulnerable to impersonation and shoulder surfing attacks, respectively. Additionally, communication sniffing threats might easily exploit the current works.
- **Trust and Credibility Issues on AI Models**- The majority of current works blindly depend on AI models and their outcomes, but since they are unaware of how the AI model decided on a specific

piece of data or entity, they have credibility and trust issues. However, although some current works use eXplainable AI (XAI), they only give explanations for traditional AI algorithms, and the explanations they supply are difficult for a common user to understand.

- **High Attack Reoccurrence Possibility**- Almost all of the efforts that are already in existence focus only on the detection of cyberattacks, while also exploiting the risk and preventative measures connected to the attacks. There is a considerable possibility of attack recurrence if none of the work is provided to them in one picture.
- **Elevated False Positive Rates**- Modern works used autoencoders as the primary deep learning model, which restricts the accuracy of attack detection and categorization since machine learning models cannot handle large amounts of data without experiencing high false positive rates.

The foremost objective of this research is to ensure end-to-end cyber security and privacy in the smart organization environment by integrating cutting-edge technologies such as cognitive computing, AI, blockchain, and federated learning. Some of the other sub-objectives in this research are provided below:

- To limit the risks associated with malicious traffic and communication by using the three-strand authentication technique and quantum communication, both of which are resistant to the main vulnerability threats.
- To mitigate the impact of social engineering attacks, execute explainable DRL-based attack detection and dynamic policy generation.
- To defend against and stop harmful network attacks by using DL-based IDS/IPS that can be understood and that also lowers the rate of attack recurrence.
- To execute FL-based global risk assessment and policy modification to reduce the risk to global security while maintaining local organization privacy.

B. Research contributions

This study's main goal is to increase cyber security in an IoT-enabled smart organization setting utilizing transparent deep learning techniques. Some of the major research highlights are listed below,

- We have presented a three-strand authentication architecture that verifies the authenticity on three different levels to tackle the key vulnerability concerns offered by fraudulent IoT users. Additionally, to prevent hacking and man-in-the-middle attacks, the organization's information is securely sent via QKSCP.
- We used the explainable DQN-A3C model combined with the SILT transformer for natural language explanations to assure the trustworthiness between the AI model and humans to increase the security level against social engineering attacks. The XDRL-generated against social engineering policies successfully reduced the incidence rate of assaults.
- To reduce dangerous external network traffic, we developed the Hopping IDS and IPS based on the explainable HGN model, SHAP, and SILT explanations, which properly characterize dangerous external traffic flows. Additionally, to strengthen the security, we also carried out a local risk assessment.
- We have implemented a knowledge-sharing system using FL technology between CTR and cloud servers, also known as global risk assessment, to improve our understanding of cyberattacks on a worldwide scale.

C. Paper organizations

The rest of the research is organized as follows; existing research shortcomings are evaluated in Section II. Section III describes the specific problem statement. The proposed work is explained fully in Section IV along with diagrams and pseudocode. The experimental setup is provided in Section V along with thorough explanations of the simulation setup, comparative analysis, and research summary. Section VI provides a thorough discussion of the suggested works' conclusion.

II. LITERATURE SURVEY

Additionally, this section describes the research shortcomings of previous studies. The author in [16] performed cyber security in an IoT environment using blockchain and federated learning technology. The organizations collaborating in this work include a base station, a blockchain network, and several clusters of smart factories. The cluster leader in each smart factory cluster was chosen to participate in blockchain operations. The cluster leader was also in charge of gathering information from the cluster members to execute local model training. Local model in this context refers to anomaly detection data utilizing a variety of AI methods, including neural autoencoder, isolation forest, K-means, cluster-based local outlier factor, and principal component analysis. The local model was then combined on the global server with blockchain support and safely sent to the local clusters. Anomaly identification in this case was carried out using artificial intelligence by the smart industry clusters head. However, failing to take into account the accuracy of smart industrial devices damaged local models.

The main goal of this study [17] was to perform intrusion detection using an efficient DL system. In this study, the author employed a deep learning technique that has been developed and is based on layered deep polynomial networks called spider monkey optimization. The dataset's collected

flows were made available for data pre-processing techniques including Minkowski distance. For feature extraction and selection utilizing spider monkey optimization, the pre-processed features were provided. Staked deep polynomial networks were used to categorize the chosen characteristics as anomalous and normal. This study solely uses optimal deep learning for intrusion detection; however, since preventive measures for intrusion detection were not taken into consideration, there is a significant chance that an attack would occur again. This study's [18] objective is to use deep learning technology for intrusion detection in a 5G IoT context. In this work, a deep learning system by the name of autoencoder is used, and it effectively combats anomalous data. Filtering was done as part of the pre-processing of the dataset's data. The significant characteristics were chosen and scaled from the pre-processed data. The autoencoder method divides the chosen and scaled features into normal and anomalous categories. The performance of the classifiers was also tested using several machine and DL classifiers, and the findings show that the autoencoder outperforms the other classifiers. In this case, anomaly detection in the 5 G-enabled IoT was performed using the deep autoencoder model. Deep autoencoder adoption, however, reduces efficiency when the rate of incoming packets is large.

In this study [19], the authors used generative artificial intelligence to find intrusions in an IoT environment. General adversarial networks served as the basis for the generative artificial intelligence technique. The parties that are participating in this endeavor are the data center, federated edge server, and social Internet of Things consumers. Users' network flows feature was pre-processed, monitored, examined, and chosen by federated edge servers. After that, the general adversarial network received the chosen features as input to carry out intrusion detection. This effort included training the general adversarial network to recognize single and multiple assaults, respectively. Here, the GAN model was used to identify network intrusion, however, the GAN models were severely limited in terms of unstable training and slow due to the limited training data.

DRL is the technique used by the authors in this study [20] to perform intrusion detection in an IoT environment. The entities involved in this effort included cloud servers, attacker-harmful servers, and IoT devices. Deep Deterministic Policy Gradient is a DRL method that operates on the current state and executes actions. Here, the network's present state is described as a statistical characteristic of the network. The action done was conducting traffic prediction, and the reward was a reduction in the error distance between harmful traffic that was predicted and malicious traffic that occurred. According to the findings, the deep deterministic policy gradient-based intrusion detection method works better than the other techniques. Here, when it comes to labeled intrusion data, data for training DRL models might be hard to come by in many IoT environments. The lack of accessible data may make DRL algorithms function poorly since these algorithms normally need a lot of data to perform well. In this study [21], authors used cognitive technologies to identify social engineering attacks. In this study, heuristics and intelligence algorithms were used to identify the behaviors associated with social engineering traps. Users' actions were encouraged by the attack features. The cognitive and intelligent model, which consists of attention and elaboration modules, was given such properties to identify social engineering assaults. Additionally, their technique detects multi-stage attacks on the phishing data. This study's methodology for detecting social engineering attacks was entirely based on cognitive activities, which was ineffective since it increased the possibility of false positives while providing improved attack detection. The deep learning model is used to identify social engineering attacks on the network.

The primary social engineering attack in this paper [22] was the phishing assault. The victims were duped into providing their personal information via bogus website pages, which allowed the attacker to access their data for malevolent purposes. The victim's personality, age, educational level, and age were the characteristics used for phishing assaults. Based on such characteristics, the susceptibility range of the victims was found to be two levels, such as less and very susceptible, respectively. This study uses many machine learning algorithms to ultimately identify social engineering attempts based on that vulnerability state. To identify phishing assaults, this paper used numerous ML techniques. However, issues with overfitting and interoperability restrict the use of machine learning algorithms.

To defend against social engineering assaults, the authors in this study [23] used an ensemble of ML algorithms. This study uses an ensemble ML model called stacking and voting classifiers to primarily identify fake phishing URLs. Initially, different natural language processing techniques were used to pre-process the URLs that were collected from the dataset. Several characteristics, including the IP of the URL, the HTTP token, the port, the rank of the page, the length of the URL, the age of the relevant domain, etc., were retrieved from the pre-processed URLs. To identify phishing attempts, the extracted characteristics were provided as input to the ensemble models known as random gradient classifier and voting classifier, respectively. Here, the fake URLs were found using the stacked ensemble machine learning algorithm. However, there were significant complexity problems with the ensemble model's use.

The authors in this study [24] used a deep learning system to identify fake phishing websites. Data samples were first gathered from a variety of sources, including Kaggle, Phish Tank, Phish Storm, etc. Natural language processing methods were used for the obtained samples to extract the different URL characteristics. Recurrent neural network, a DL model developed based on that characteristic, is used to identify fraudulent phishing websites. Recurrent neural network-based models outperform other works, according to machine learning models used to evaluate this work's performance. Here, the recurrent neural network was used to identify phishing URLs even though it lacked complicated training processes and had a high temporal complexity.

In this study [25], the authors used an ensembled classifier to protect the cyber-physical system from harmful attacks. hosts (which include both benign and malignant hosts), gateway, ensemble deep learning model, feature fusion module, and machine learning-based Meta classifier are some of the entities included in this study. The ensembled DL module used three DL techniques, Recurrent Neural Networks (RNN), long short-term memory (LSTM), and gated recurrent units (GRU), to extract features after receiving the input from the hosts. For dimensionality reduction and feature fusion, the kernel-based principal component analysis was used with the features that were derived from those models. The ensemble meta-machine learning classifiers, such as random forest and logistic regression, were then used to identify and categorize the intrusions using the fused feature as an input. This process included both good and bad hosts in the network, and since those users were permitted to communicate data without having their identity verified, impersonation attacks were made possible. The Internet of Things ecosystem was made more secure by using the fuzzy technique.

This study [26] proposes the Cognitive Spammer Framework (CSF) is used for web spam detection. In this study, the malicious websites were identified and eliminated using several cognitive techniques. The web pages were first gathered and cleaned up as part of the initial pre-processing of the dataset. Principal component analysis was used to extract characteristics from pre-processed data. The three machine learning classifiers—bagged multivariate adaptive splines, Bayesian generalized linear model, and boosted linear model—were then given the retrieved features. The data to identify the spammy websites was then sent to the fuzzy rule-based classifier. Many IoT devices have limited resources including memory and processing speed. Here, operating complex cognitive frameworks like CSF on these systems might be difficult or have serious performance issues.

To detect and prevent botnet attacks, the authors of this study [27] used an optimization technique based on deep learning. The dataset's data were pre-processed after being gathered. The deep neural network was given the pre-processed feature to process the network features, and the Local Global Bat Algorithm (LGBA) was used to optimize the deep neural network's hyperparameters. The neural network-based local global bat optimization method received the characteristics to help it identify and categorize botnet attacks. IoT networks are dynamic, and devices often join and leave the network. The actions of a botnet can change over time. It could be difficult for LGBA to continually adjust to these dynamic changes.

The authors of this paper [28] suggested employing deep learning algorithms to detect and resist jamming attacks in a cognitive radio-IoT context. Primary users, secondary users, and fusion centers are the organizations engaged in this work. Additionally, the attackers were active in destroying the malicious activities. The secondary users' detection associated with the main user's usage of the available spectrum was combined and sent to the autoencoder model, which categorizes whether or not the sensing data was jammed. In this article, the autoencoder model was used to make the ultimate determination of jamming attacks, but the deep learning model was not trusted, which resulted in greater false positive rates.

For identifying and categorizing botnet assaults in an IoT-enabled edge environment, the author of this research [29] presented federated learning and deep learning technologies. The worldwide server and IoT-enabled edge devices are the parties participating in this work. All IoT-enabled edge devices compute their local models using a deep neural network to update the weights from a global server. The global server received the locally created model and used the Federated Averaging (FedAvg) algorithm to collect it. The local model devices were then given the collected data so they could update their local models and increase their capacity to identify attacks. Federated learning was used in this instance to improve the accuracy of botnet attack detection rates. Although it protects the privacy of local parties, the lack of data privacy protection (i.e., communication channels with no security) results in man-in-the-middle and poisoning assaults, respectively.

In this research [30], the IoT ecosystem was protected against intrusions by using blockchain and federated learning technologies. IoT devices, a blockchain-based organization server, and a worldwide server were the participants in this study. Every IoT device uses a machine learning technique to train its local model. The intermediary blockchain server combined the locally created models, which lowers the attack vulnerability rate. After receiving the local model collected for global model aggregation, the global server distributed the updated model to the IoT devices. In this case, intrusions were effectively identified through federated learning based on machine learning and blockchain technology. However, the possibility of an attack recurrence is increased since the risk associated with them was not evaluated and rectified. In this study, the author [31] suggests a method for detecting and preventing DDoS attacks in an IoT network. Blockchain technology has the potential to transform how data and information are transferred between unreliable parties. A technical advance is the ability to establish trust in scattered settings without the need for central authority, and blockchain technology could be a suitable method for safeguarding distributed, limited-trusted Internet of Things systems. In this case, blockchain operations require a consensus process, which might cause a delay. Low-latency communication is often essential for real-time reaction to attacks in IoT systems. For applications that need almost immediate decision-making, blockchain's inherent latency might be an issue.

In this research [32] for addressing the risk and identifying attacks in IoT, an effective security system is suggested by the author. By combining Spider Monkey Optimization (SMO) and Hierarchical Particle Swarm Optimization (HPSO), the suggested hybrid optimization approach can handle a lot of intrusion data classification problems and improve detection accuracy by decreasing false alarm rates. In this situation, the entire system becomes increasingly complicated as a result of hybrid optimization

methods that often integrate various algorithms or methodologies. Understanding and maintaining the intrusion detection system may be challenging as a result of its complexity.

In this study [33] the author developed a novel framework for IoT network intrusion detection based on XAI. To quickly detect intrusions, a deep neural network model was first included in the architecture. Along with detecting IoT-based dangers, the XAI-based Framework also provides more details and an explanation of how and why our deep neural network model makes certain detection judgments. Due to the variety of sensors and devices, IoT data may be very complex, multidimensional, and heterogeneous. So, in this case, it may be difficult for XAI approaches to provide clear explanations for complex, high-dimensional data, making it difficult for analysts to understand the reasons for detection decisions.

For the IoT ecosystem, this article [34] developed a concept for an AI-based method for detecting harmful smart contracts and intrusions. To effectively identify unauthorized users (intruders) from the IoT ecosystem, AI-based algorithms such as the random forest, decision tree, logistic regression, MLP, and Bayesian model were first taken into consideration. In this scenario, malicious users are eliminated from the IoT ecosystem, and only genuine users are permitted to send data to the designated receiver. Further, to enhance safety this study uses deep learning methods to distinguish between malicious and benign smart contracts. In this case, missing values, noise, and consistency issues might affect IoT data. The performance of DL models may be affected by poor data. Although important, data pretreatment and cleaning can be resource-intensive. Here, Table I describes the research gap in the literature survey.

TABLE I
RESEARCH GAP IN LITERATURE SURVEY

Reference	Objectives	Methods or Algorithms Used	Limitations
[16]	To identify suspicious users and transactions on an IoT network built on blockchain that is particularly designed for smart factories.	Isolation Forest (IF), K-means, Principal Component Analysis (PCA).	The reliability of the smart industrial devices was not taken into consideration, which damaged local models.
[17]	Enhanced threat detection efficiency using a novel Deep Learning-based IDS (DL-IDS) architecture to safeguard IoT environments.	. Spider Monkey Optimization algorithm (SMO)	Since preventative methods for intrusion detection were not taken into consideration, this study performs intrusion detection using optimal deep learning, which increases the possibility of an attack reoccurrence.
[18]	Implementation of an efficient intrusion detection system capable of identifying assaults based on the IoT.	Autoencoder algorithm.	Using a deep autoencoder increases complexity when the rate of incoming data is high.
[19]	To tackle safety issues and improve edge network security.	Generative Adversarial Network (GAN) based intrusion detection method.	GAN models were severely limited by inconsistent training and delay due to the small amount of training data.
[20]	DRL for performing intrusion detection in an IoT environment.	Deep Deterministic Policy Gradient (DDPG) algorithm	Difficult to perform due to shortage of labeled data in IoT.
[21]	Cognitive technology for detecting social engineering attacks	Heuristics and intelligent methods	Not so effective, leads to higher false positive rates.
[22]	Detecting the network against social engineering attack using deep learning model.	Decision Forest, Logistic regression, support vector machine, boosted decision tree.	Interoperability and overfitting problems cause limitations.
[23]	An ensemble of machine learning techniques for defeating social engineering attacks.	Decision trees (DTs), random forests (RFs), k-nearest neighbors (KNN)	High complexity issues.
[24]	Phishing detection in real-time browsing environment using deep learning.	Recurrent Neural Networks (RNN), Random Forest (RF).	Complicated training processes and increased time complexity.
[25]	Ensemble classifier to protect the cyber-physical system from harmful intrusion.	RNN, LSTM, GRU, Convolution neural network (CNN).	The elaborate details of the learning process that took place in the intermediary neural network layers are hidden by DL approaches.
[26]	CSF is used for web spam detection.	Principal component analysis, RF	Since the suggested strategy focuses on the identification and prevention of spam images, IoT devices have difficulty processing a high number of images.
[27]	Deep learning-based optimization algorithm for detecting and mitigating botnet attacks.	Bat Algorithm (BA), Local-Global Best Bat Algorithm for Neural Networks (LGBA-NN).	Difficult to adjust to dynamic changes.
[28]	Detecting and resisting the jamming attacks using deep learning algorithm for CR-IoT environment.	Deep autoencoder (DAE)-TRUST approach.	Higher false positive rates.

[29]	Detecting and classifying the botnet attacks in IoT-enabled edge environment.	Federated Deep Learning (FDL), DL, Federated Averaging (FedAvg) algorithm.	No protection for the privacy of data, so attacks might occur.
[30]	To safeguard the IoT environment against intrusions.	Hierarchical Blockchain-based Federated Learning (HBFL)	Attack reoccurrence possibility.
[31]	Method for detecting and preventing DDoS attacks in an IoT network.	Consensus, Smart Contract	High latency.
[32]	An effective security system for dealing with threats and spotting IoT attacks.	SMO, Hierarchical Particle Swarm Optimization (HPSO), and RF.	High complexity.
[33]	To defend IoT-based networks against newly developing IoT-based assaults, a unique DL-based architecture utilizes Deep Neural Networks (DNNs).	RuleFit, Local Interpretable Model-agnostic Explanations (LIME), and SHapley Additive exPlanations (SHAP)	Difficult for XAI methods to understand complex IoT data.
[34]	Deep learning methods to distinguish between malicious and benign smart contracts.	LSTM, GRU, and artificial neural network (ANN)	Not flexible to low-quality data present in IoT.

III. PROBLEM STATEMENT

An ongoing key problem is improving security for IoT-enabled smart organizations worldwide. A few of the specific problems highlighted in the latest research are as follows:

To defend IoT devices against network cyber-attacks, federated learning, and deep learning technologies are used. In this study [35], a deep learning technique called deep autoencoder was utilized to filter unwanted network traffic data. In this research [36] to avoid social engineering attacks in smart organizations, blockchain technology-based double control is used. The organization recipient, attacker, verifier, blockchain, and key management server are the participants in this effort. Some of the problems detected in these papers are:

- The network traffic data was detected and analyzed using the deep autoencoder model. Adoption of a deep autoencoder model, however, has severe complexity limitations and is not appropriate for large data sets.
- The entities (i.e., IoT devices) participating in this effort weren't guaranteed to be legitimate; instead, they generated network traffic to the gateway, amplifying the main vulnerability concerns such as shoulder surfing attacks and cloning attacks, etc.
- Additionally, this work is particularly vulnerable to communication-related assaults like man-in-the-middle attacks, sniffing, etc., which creates a danger to the IoT users' and organizations' respective privacy concerns.
- Furthermore, the federated learning server's aggregated global model was left insecure, which allowed model poisoning attacks to occur, which had an impact on all local entities.
- This work uses blockchain technology to identify vishing, a sort of social engineering assault. However, human background verifiers were able to identify the phishing, which resulted in increased false positive rates.
- Along with false positive rates, the main concern with their detection findings was the trustworthiness of the results since a corrupted verifier may purposefully offer incorrect detection on genuine users.
- Finally, the two-stage verification for vishing allows for better outcomes, but brute force assaults are caused when the validity of entities like the organization receiver and verifier is not guaranteed.

The authors of this paper [37] used artificial intelligence technologies to identify potential cyberattacks in an IoT-based cyber environment. For attack detection and attack attribute capture, this study uses a dual-stage ensemble ML approach and a DL algorithm. In this study [38], explainable AI technology was used to identify and prevent distributed denial-of-service attacks in a cyber IoT environment. The autoencoder DL method is utilized in this case to identify distributed denial of service attacks. Problems faced by researchers are:

- This study implements a two-stage framework for detecting cyberattacks utilizing deep learning and machine learning, including the use of principal component analysis for feature selection. Principal component analysis, however, has poor results since it only takes into account the linear relationships between the features.
- The use of the decision tree and support vector machine ensemble machine learning algorithms is limited by their poor interoperability and increased complexity.
- The trust in the two-stage deep learning and ensemble machine learning models was not justified which leads to higher false positive rates and also imposes credibility issues.
- Here, it simply detects attacks however, since mitigation techniques and the risks associated with those risks were not supplied, the possibility of attacks reoccurring was increased.
- The primary goal of this research was to use an explainable artificial intelligence strategy to identify distributed denial of service threats in the IoT context. The effectiveness of a distributed denial of service assault, however, was not yet achieved without taking user authenticity into account.

- The use of a typical and standard deep learning model called an autoencoder could not withstand the effective intelligent detection of distributed denial of service assaults, which also has an impact on detection reasons.
- Here, the detection server was centrally located, allowing the attacker to cause a distributed denial of service on the central server, which resulted in a single point of failure and security vulnerabilities.

Decision tree and random forest ensemble multi-categorization models are used in this study [39]. The detection results were explained using the shapely adaptive explanation technique, which provides values to the classification and explanation findings as well as an explanation of the feature's importance. Some of the major problems in this study are:

- Here, an ensemble machine learning model was used to carry out the intrusion detection process. However, the use of machine learning algorithms is constrained by overfitting and underfitting as a result of excessive and insufficient training sample feeding.
- Only intrusion detection was carried out in this cyberspace, no preventative measures were implemented, and the risk connected with the assaults was not examined, which increased the chance of attack recurrence.
- This study also uses a shapely adaptive explanation tool to acquire explanations. However, only highly experienced experts might grasp their explanations, which has an impact on decision-making.

Research Solutions

To overcome data integrity, immutability, distributed nature, and irreversibility, blockchain technology is used. IoT Layer (IoTL), Cognitive Layer (CL), and Cyber Risk Analysis Layer (CRAL) are the three layers that make up the suggested work. The IoTL consists of workers with their IoT devices, while the CL is made up of edge and cloud servers, with the edge server being used for monitoring the human factors of the employees and the cloud server being used to validate network traffic as well as execute local risk assessments. The findings of the global risk analysis are finally updated to the underlying local parties using the CRAL. QKSCP is used to reduce communication channel vulnerabilities like Man in The Middle (MITM), reply attacks, etc. The explainable DQN-A3C model was used to increase security against social engineering attacks. The Shapely Adaptive Explanation (SAE) tool is used to increase trust in the AI model that was created. Hopping IDS and IPS were developed to reduce harmful external network traffic based on the explainable HGN concept.

IV. PROPOSED METHOD

The primary goal of this study is to strengthen the security and privacy of IoT-enabled smart organizations by merging XAI, FL, and blockchain technology. Adoption of XAI improves the accuracy of decisions on the outcomes and increases trust levels in AI. By using FL technology, participating smart organizations may maintain their privacy without revealing their true identities. Additionally, by using blockchain technology, the issues of data integrity, immutability, distributed nature, and unchangeability are resolved. Three layers make up the proposed work: the Cognitive Layer (CL), the IoT Layer (IoTL), and the Cyber Risk Analysis Layer (CRAL). Employees using IoT devices are part of the IoTL, and the cloud and edge servers that make up the CL are used to validate network traffic as well as execute local risk assessments. The edge server is used to monitor employee human factors. To update the underlying local parties on the findings of the global risk analysis, the CRAL is used. Figure 1 describes the proposed architecture. The processes involved in the proposed work are:

- Triple Strand Authentication
- XDRL-based Social Engineering Attack Detection & Prevention
- XDL-based Hopping IDS & IPS
- Federated Learning-based Global Cyber Risk Assessment

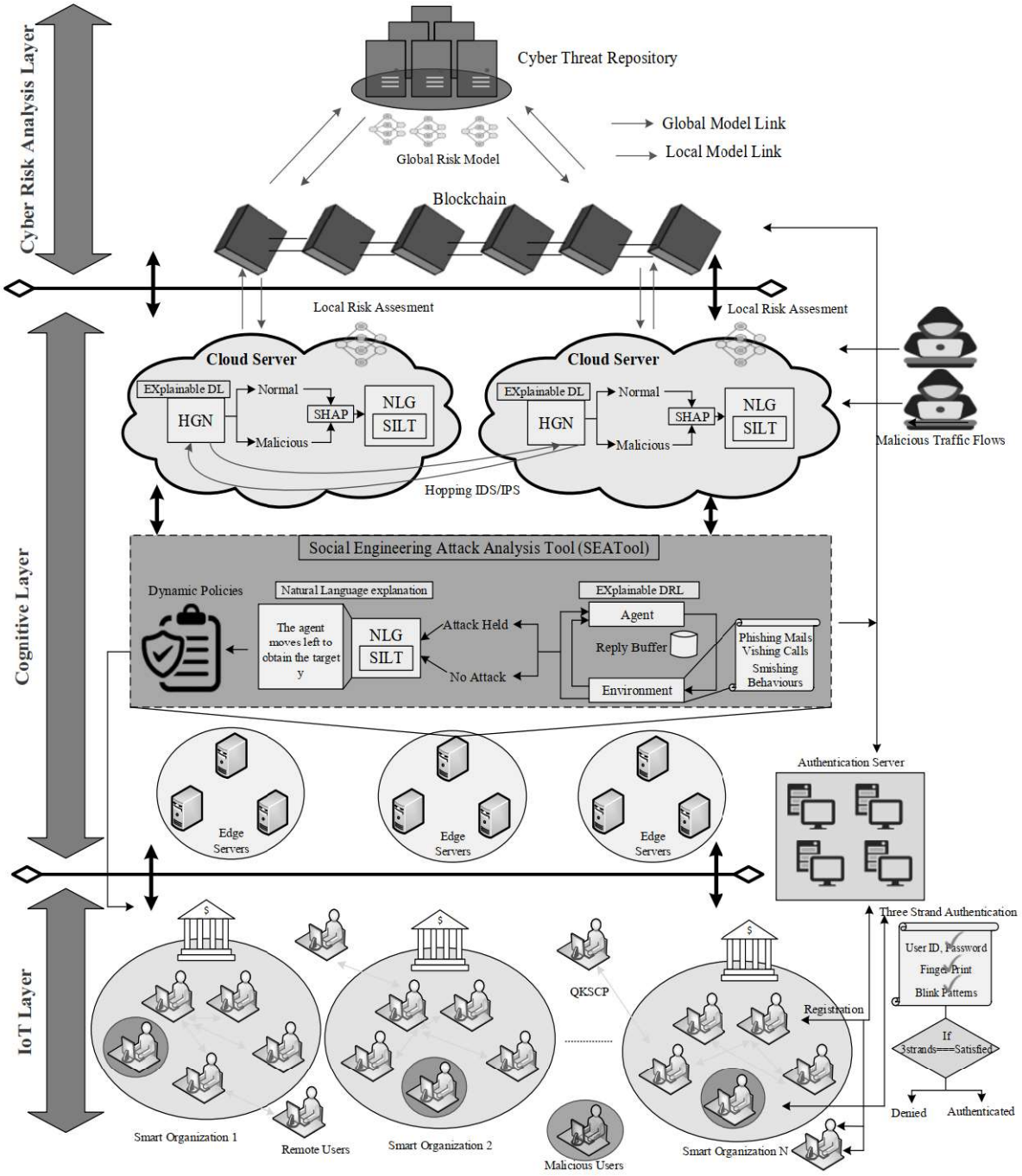

Fig 1. Proposed architecture

A. Triple Strand Authentication

Initially, the employees are registered and authenticated to the Authentication server (AS) with their respective organization IoT devices and biometrics. The blockchain helps the AS by giving the workers tamper-proof and secure authenticity. In our work, the employees are authenticated using three different methods: (1) In the first method, the employees' user names, IDs, and passwords are used to confirm their reliability; (2) In the second method, the employees' fingerprints are used to confirm their authenticity; (3) In the third method, the employees' unique eye blink patterns are registered and verified to complete the employee legitimacy. The Capsule Network (CapsNet) algorithm is used to process the obtained credentials, such as fingerprints and blink patterns.

a. Capsule Network algorithm

A sort of deep learning architecture is capsule networks, sometimes known as Caps Nets. Specifically in managing hierarchical and spatial connections between features, Caps Nets were created to solve some of the shortcomings of conventional convolutional neural networks (CNNs) in computer vision applications. For fingerprint identification jobs, Caps Nets may be used. In this case, the IoT device would use a camera or fingerprint sensor to take a picture of a person's fingerprint. A Caps Net model would then be used to process the acquired picture. In the context of the IoT, CapsNets may be used in the authentication process to increase security and the accuracy of user or device identification. Blink pattern authentication utilizing CapsNets may offer a higher level of security and user comfort, particularly in situations where conventional authentication techniques might not be feasible or safe enough.

The core architecture for CapsNet is shown in Figure 2. The input data is the preprocessed data. The convolutional operation processes the data in a layer with various filters after that. Data is then further processed by the conv with squash activation in the primary capsule layer before being replaced. The reshaped data is then handled by "Dynamic Routing" to produce feature capsules in the feature capsule (FC)layer. In the form of a one-hot encoder, an Auxiliary Layer finally substitutes each FC layer capsule with its length to correspond to the true label. The core operation in CapsNet is as follows:

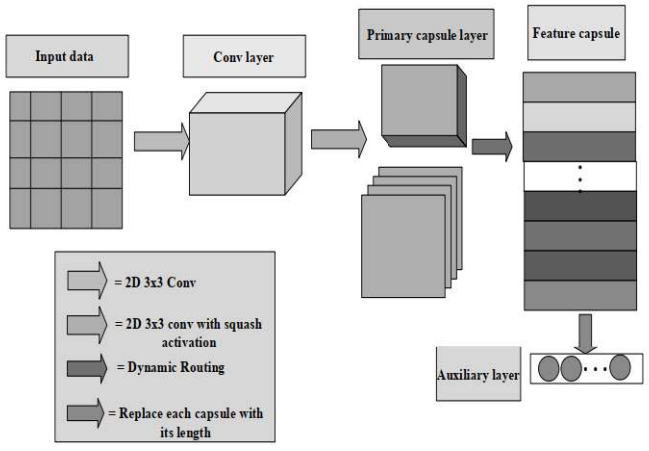

Fig 2. The architecture of CapsNets

i. Convolution operation

Let $x_i \in R$ be the one-dimensional data. The input data vector x^j of length n is represented as $x^j = [x_1^j, x_2^j, \ldots\ldots x_k^j]$, j=1,2,…..M, where M is the number of training points and n is the number of Aps. A convolution operation involves a filter $w^j \in \mathbb{R}^n$, which is applied to the vector x^j to produce a new feature. For instance, a feature c_i^j is generated from the vector x^j by:

$$c_i^j = \mathrm{f}(w^j . x^j + b^j) \quad (1)$$

Here, b^j is the bias term; f stands for the nonlinear activation function, which introduces non-linearities to CNN and is preferred for multi-layer networks to identify non-linear aspects of input data. The filter w^j is used on each vector to x^j where j is the number of vectors required to create a feature map as:

$$c^j = |c_1^j, c_2^j, \ldots\ldots c_n^j| \quad (2)$$

ii. Dynamic Routing

Capsules are neuron groups that indicate various metrics of activity for such neurons. The vector's length shows an indication of the existence of an entity. The levels of pooling are mostly responsible for CNN's issues. Therefore, these layers are replaced in the capsule networks by a better criterion known as "routing by agreement" based on this standard, however in the parent capsules in the next layer, their coupling coefficients differ. If the anticipated parent capsule output matches the actual parent capsule output, each capsule predicts the parent capsule output, and the cup coefficient between these two capsules is raised. Given m_x as capsule output x, its prediction for parent capsule y is as calculated.

$$m_{y|x} = N_{xy} u_x \quad (3)$$

where $m_{y|x}$ is the vector prediction of the y^{th} output of the capsule at a higher level, which is estimated by Capsule x in the lower layer, and N_{xy} is the weighting matrix that is required knowledge for the backward pass The coupling coefficients a_{xy} are calculated by using the following softmax function based on the conformation level between the capsules in the below and the parent capsules

$$a_{xy} = \frac{\exp(c_{xy})}{\sum_z \exp(c_{xy})} \quad (4)$$

where the log-likelihood (c_{xy}) is that if the capsule x and capsule y are connected and its initial routing setting is 0. As a result, the input vector for the parent capsule y is determined accordingly.

$$p_{y=} \sum_x a_{xy} \; m_{y|x} \quad (5)$$

The next nonlinear squash function, which determines each capsule's ultimate output based on its initial vector value described in Eq. (5), prevents exceeding one of the output vectors of capsules.

$$v_y = \frac{\|s_y\|^2}{1+\|s_y\|^2} \frac{s_y}{\|s_y\|} \quad (6)$$

where s_y and v_y are seen as the input and output vector of y. The log probabilities in the classification process should be changed based on the agreement between v_y and u_y, based on the fact that if the two vectors agree, a significant internal product will be provided. As a result, in order to update log probabilities and coupling coefficients, agreement is determined as follows.

$$q_{ij=v_j} \cdot m_{j|i} \quad (7)$$

Every capsule z in the final layer contains a loss function r_z that gives a high percentage value on capsules with lengthy output instantiation parameters in the absence of entity. The loss function r_z is computed accordingly.

$$r_{z=}S_z \max(0, t^+ - \|u_z\|)2 + \lambda(1 - S_z) max(0, \|u_z\| - t^-)^2 \quad (8)$$

S_z is 1 exists if z is present, or else S_z is 0. Before the learning process, the terminology or hyperparameters t^+, t^- and λ indicated.

In this paper, the use of three-strand authentication prevents the main authentical threats, such as brute force, impersonation, and shoulder surfing attacks, and lowers the amount of malicious communication in the IoT-enabled cyber environment. Only the first and third strands of authentication are used for distant workers. Additionally, we have created the QKSCP to lessen communication channel vulnerabilities like Man in The Middle (MITM), reply attacks, etc. To be more precise, QKSCP is used to facilitate communication between the organization's IoT device and sender and receiver, who can only encode and decode data, respectively.

b. Quantum Key Secure Communication Protocol (QKSCP)

IoT devices have several security problems that threaten the safety of the users, the devices, and the network. Thus, before quantum computers are available, we must develop quantum-secured protocols to guard against vulnerabilities in our systems. The development of a cryptosystem using quantum mechanics has been considered to be the most secure system, and quantum cryptography is a particularly fascinating field. Nobody can violate it without the sender or recipient of the communication being aware of it. One of the most fundamental methods used in quantum cryptography is the quantum key. The greatest feature of the quantum key protocol is the ability of the channel to identify any eavesdroppers in the IoT systems. QKSCP provides a high degree of protection, especially against assaults based on the usual algorithms used today. QKSCP may provide effective security for maintaining the confidentiality and integrity of data sent between IoT devices and other systems when it is used for IoT devices. Greater network scalability is provided by public key infrastructure (PKI), and QKSCP with PKI promises to provide robust network security. The initial connection just has to be secure for QKSCP authentication to be equivalent to public key infrastructure authentication. If the QKSCP manages to generate a small number of secret keys, QKSCP authentication may then be accomplished utilizing these secret keys. It doesn't matter even if Eve is successful in obtaining the first authentication keys after the first QKSCP conversation. The recovered keys from both rounds of the QKSCP may provide unmatched information security at the time of the subsequent communication. Below are the different steps that make up the technique:

1. **Initial phase**

The first step involves the following actions:

- To establish a "secure" connection, Alice and Bob submit their IDs.
- Information centers employ "public key authentication" to confirm that they are authorized users of the "public key" infrastructure. The information centers create random numbers when the public key authentication is successful. Additionally, the information centers deliver private key-encrypted, individually generated KEY POOLs. KPA belongs to Alice, whereas KPB belongs to Bob.
- The first information center creates a quantum communication link and sends copies of the key pools to both parties.
- A copy of the keys is shared between Alice and Bob during their first communication via the information center.
- Otherwise establishes a quantum communication channel without a POOL exchange.

2. **Mutual authentication**

- Alice publicly asks Bob for a key from the POOL KPA. Bob looks up that key in KPB. Only when the keys coincide does the transmission take place.
- Bob publicly requests a key from the POOL KPB from Alice. Alice validates that key using KPA. Only when the keys match does the transmission take place.
- For each transmission, this procedure is repeated. This prevents spying between the two as a result. This also ensures that mutual cent percent user authentication is completed as only Alice and Bob are aware of their key information from the KEY POOL.

Additionally, Alice and Bob store the replacement keys from the prior operation and delete the duplicate keys that were previously used in authentication from their KEY POOL. By doing this, it is made sure that the key cannot be duplicated and utilized for financial gain by anybody else. A higher rate of transmission success is attained as a result of the reciprocal authentication of the keys.

The most secure authentication is mutual authentication, a three-pass identity verification technique.

B. *XDRL-based Social Engineering Attack Detection & Prevention*

In the environment of the smart organization, only the authenticated workers will be permitted to access their job information (i.e., for both the kinds of employees who work in the company and also from home). In such a situation, it's crucial to protect sensitive information. To do this, we've used cognitive analysis with XDRL in the edge server to defend against social engineering attacks, including phishing, vishing, and smishing assaults, respectively. The Social Engineering Attack Analysis Tool (SEATool) on the edge server is based on the DQN-A3C, an XDRL algorithm. The DQN-A3C gathers the current state (i.e., employee behaviors on being vulnerable to social engineering assaults) and takes appropriate action according to that (i.e., attack or not). To be more specific, organizational devices are used with continuous monitoring for phishing emails, vishing phone calls, and smishing messages. Additionally, the behavioral responses of the workers to these threats are tracked and categorized. For example, if an employee opens a phishing email, the authenticity of the touching may be verified based on the employee's anxiety, curiosity, and urgency. The related email can be verified as well based on the sender's IP and uniqueness. These factors led to the classification of the DQN-A3C into two categories, such as attack and no attack.

a. Dual Q Network-based Asynchronous Advantage Actor-Critic algorithm (DQN-A3C)

A DQN is a DRL algorithm used in smart environments to find anomalies. IoT networks are prone to a variety of cyber threats, and DQN can be used to recognize and stop these kinds of attacks. It involves training an agent to take actions that improve network security based on identifiable states and rewards when utilizing DQN for attack detection in IoT environments. It is useful to the IoT deployments' cybersecurity toolset, particularly as IoT networks continue to become more complicated and vulnerable to assaults. States in DQN represent the information provided by the environment to an agent so that it can react. A decision made by the agent after analyzing the environment within a specific duration, such as after completing a minibatch, is referred to as an action. In DQN, a reward is the response from the surrounding environment to an agent's corresponding action. Based on the size of the mini-batch and the output values of DQN, a reward vector can be created. Our anomaly detector depends on the DQN algorithm, which constantly improves the performance of its detection.

i. DQN: Anomaly detector $\rightarrow \pi$

The policy π is followed by an anomaly detector. Equation (9) defines the policy π as a conditional probability distribution.

$$\pi := x(M \mid Z) \tag{9}$$

where Z is the set of the system's states and A is the series of its actions. π (z, m) = x (M = m | Z = z) denotes the probability for the action m in the provided state z.

ii. Detector Performance $\rightarrow V_\pi$

The performance of the anomaly detector is given by (10).

$$A_\pi = \sum_{z \in Z} y^\pi (s) \sum_{m \in M} Q(z, m) * \pi(z, m) \tag{10}$$

where y^π (s) equals the probability that the system will be in state z while functioning by the policy π. The cumulative reward from state z with action a is represented by Q (z, m). The performance of the anomaly detector is determined by the average cumulative reward under the policy.

iii. Optimal anomaly detector $\rightarrow \pi^*$

Maximizing performance is the goal of an optimal anomaly detector. The maximal performance is achieved by following the optimal policy as given by (11).

$$\pi^* = argmax_\pi A_\pi. \tag{11}$$

In the case where $y^\pi(z)$ is approximately equal for all $z \in Z$ and $|Z|$ is the number of states in Z, it follows that:

$$A_\pi^* = max_\pi A_\pi = \frac{1}{|Z|} \sum_{z \in Z} max_m Q(z,m) \tag{12}$$

Equation 12 demonstrates that the perfect anomaly detector that follows the optimal policy π^* is determined by the cumulative Q-value function Q(z, m). This holds under the assumption that the anomaly detection problem is deterministic and y^π (s) is approximately uniform.

b. A3C

A reinforcement learning method called the Asynchronous Advantage Actor-Critic (A3C) algorithm can be adapted to detect attacks in Internet of Things (IoT) systems. The A3C Algorithm is much faster and simpler. A global agency and several individual employees make up A3C. DNNs are used to represent both the global agent and individual learners. Each DNN has two outputs: one for the actor and one for the critic. The first output is a scalar reflecting the anticipated reward of a certain state V(y), and the second, is a vector expressing a probability distribution over all potential actions (y,d). Through the use of multi-processor computing, the suggested A3C architecture might enhance the effectiveness of the anomaly detector. One global network and multiple worker agents make up the architecture of an anomaly detector. The five stages of cyclic training are as follows. Each worker is initially reset to the global network, after which they interact with their respective environments, estimate value and policy losses, compute gradients from losses, average their gradients,

and then update the global neural network weights. They then go back to step one to repeat the training process until convergence or the time limit is reached. The batches y_{t_1} d_{t_1} y_{t_1}+1,y_{t_i} a_{t_i} y_{t_i}+1, from the database are sampled for each worker neural network, and the states are then loaded into the policy NN (parameter θ' for worker, θ for global) and state value NN ($\theta_{x'}$ for worker, θ_x for global). The difference between the estimated value function Rt and the real value function V_t (advantage value $A_t = R_t - V_t$) is employed to train the NN approximator of the policy function. If a favorable advantage is returned, increase the possibility of action d_t^n, else lower the possibility if a negative advantage is returned. Each worker updates their own θ' and $\theta_{x'}$ parameters, which are then submitted to the global. To avoid overestimating particular activities, the derivative to the global parameter pθ, pθ_x is calculated on the average of each work θ', $\theta_{x'}$. The policy's entropy (H(π ($d_j|y;\theta'$)) is included to promote exploration and avoid unnecessary, unsatisfactory convergence. The neural sampler provides batches of training samples y_{t_1} d_{t_1} y_{t_1}+1,y_{t_i} d_{t_i} y_{t_i}+1… The detector does the action at state Y_t and gathers the current rewardr_t and the next state y_{t+1} from the environment. The log probability of the chosen action $\hat{d}_t$ under the probability distribution and weighed based on the relevant advantage value D_t is used to represent the policy loss function in the form of D_t log $\pi(\hat{d}_t)$. By reducing both the value loss function and the policy loss function, the actor and critic are constructed and trained concurrently.

Here, algorithm 1 describes the anomaly detector based on A3C.

Algorithm 1: Anomaly detection based on A3C

Setup the global shared parameters $\theta_x$and counter N: =0;
Set the settings for the worker thread θ', θ_{x1} and time step t:=1;
while N<N_{max} **do**
pθ:=0, pθ_x:=0;
θ':=θ, θ'_x:=θ_v;
t_{start}:=t, get state s_t;
while (y_t is NOT terminal) or (t- t_{start}<=$t_{\max}$) **do**
perform actions $d_t = \pi(d_t | y_t ; \theta')$;
get reward r_t and transition to y_{t+1};
t:=t+1, N:=N+1
end while
if (y_t is end state) **then** R:=0; R:=V(y_t, θ'_x);
end if
for j ∈ (t − 1, . . .t_{start}) **do**
R:=r_j+γR;
pθ := pθ + ∂logπ(a_j |s_j; θ')(R − V(y_j ; θ'_x))/∂θ' + β∂H($\pi(d_j | y_j ; \theta')$)/∂θ' ;
pθ_x := pθ_x + ∂(R − V(y_j ; θ'_x))2/∂θ'
end for;
Update pθ and pθ_x to global asynchronously
end while

In this paper, the DQN-A3C algorithm is used as an anomaly detector, which is used to detect and identify the type of attack in IoT devices. Additionally, by using the SILT, we presented explanations for the actions performed using the Natural Language Generator (NLG). One can easily understand the classification outcomes of the XDRL with the use of natural language explanations, which increases confidence in the AI model. Fig 3, describes the anomaly detection architecture based on A3C.

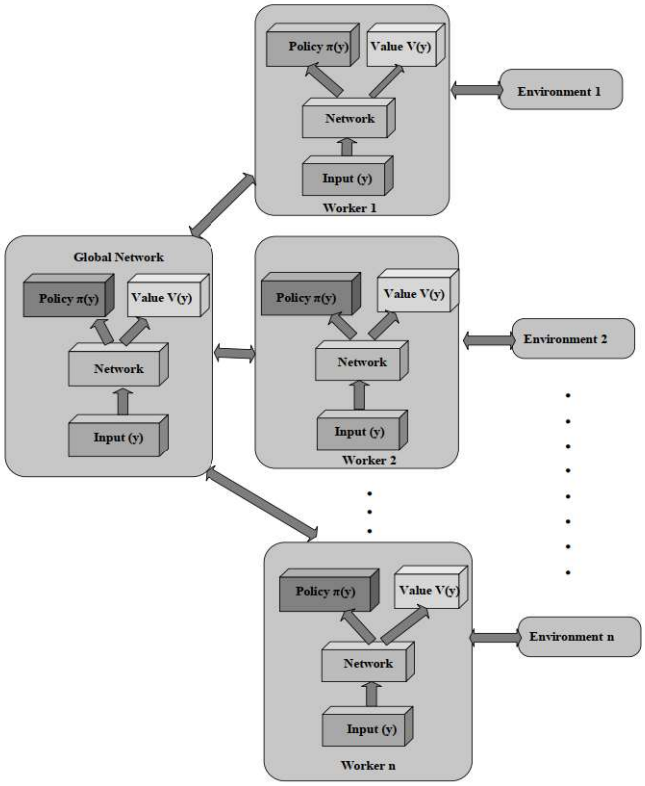

Fig 3. Anomaly detection architecture based on A3C

c. Siamese Inter Lingual Transformer (SILT)

The SILT appears to be a model that uses Siamese architecture and can handle several languages. Its specific use is to explain the actions carried out by an NLG system. These explanations could help debug and enhance the NLG model as well as make the activities of the NLG system more understandable or visible to users. In this context, the term "NLG" most likely refers to the process of generating text that is understandable by humans from structured data or other forms of non-linguistic input.

Additionally, to strengthen access to IoT devices even more, access policies are developed dynamically based on the explanations they provide. For instance, if a phishing assault is discovered and described, the edge server will block such emails and tell the user not to respond to them. The chance of

social engineering manipulation in the proposed smart organization setting is significantly decreased by examining both human and technological factors.

C. XDL-based Hopping IDS & IPS

In this study, the cloud server will additionally validate and examine traffic from external networks using the HGN XDL method. The network traffic characteristics (flow and packet features) collected by the cloud server include packet arrival time, packet location, time to live, flow length, Source IP and destination IP, Communication protocol, source port, and destination port. These characteristics are provided to the HGN, which offers two classifications into classes: normal and malicious.

a. Harmonized Google Net (HGN)

IoT network security may be improved or specific IoT data analysis tasks can be carried out by combining deep learning models, such as convolutional neural networks (CNNs) like HGN, with conventional techniques. The most popular choice among various deep learning architectures that may be used for network traffic analysis is HGN-based architecture. Implementing HGN for IoT network traffic monitoring is a difficult and continuing effort, but immediately identifying and avoiding assaults, may greatly increase the IoT ecosystem's security. The network is to be expanded via the HGN Model. The HGN Model aims to expand the network. The structure is shown in Figure 4. The inception structure is a crucial part of it and could improve the network's accuracy. The dimensionality reduction of the 11-convolution kernel may lead to a reduction in the number of parameters and an increase in network depth. Branching and merging may widen the network, which increases network accuracy.

The HGN network, the Inception-v2 structure, the Inception-v3 structure, and the Inception-v4 structure have all seen ongoing improvements. The Inception-v3 structure essentially replaces the one-dimensional convolution kernel, while the Inception-v4 structure principally adds the residual network concept. The Inception-v2 structure largely includes the batch normalization layer. The batch normalization layer is mostly added to the Inception-v2 structure. The use of the grouped convolution technique is required. Sometimes, the information flow between groups can be awkward in grouped convolution, which will also increase the model's size. The parameter values are lower in the channel shuffle convolution because it uses a convolution core that only applies to one channel. The channel shuffle is seen in Figure 5. Labels can be seen in the figure on the far left, where gconv stands for packet convolution. In Figure 5, there are three pictures. The most common method is shown by grouping convolution in Figure 5(a). Figures 5(b) and 5(c) show the information-sharing process and impact, respectively, after the addition of the improved idea.

An effective tool for evaluating network integrity and security may be obtained by using HGN for network verification, particularly when combined with additional network monitoring and security solutions. To evaluate the security and integrity of network traffic, HCN, CNN architecture, is often modified.

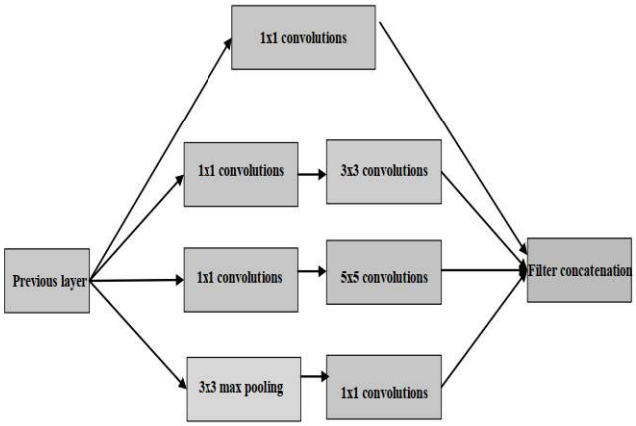

Fig 4. Google Net inception structure

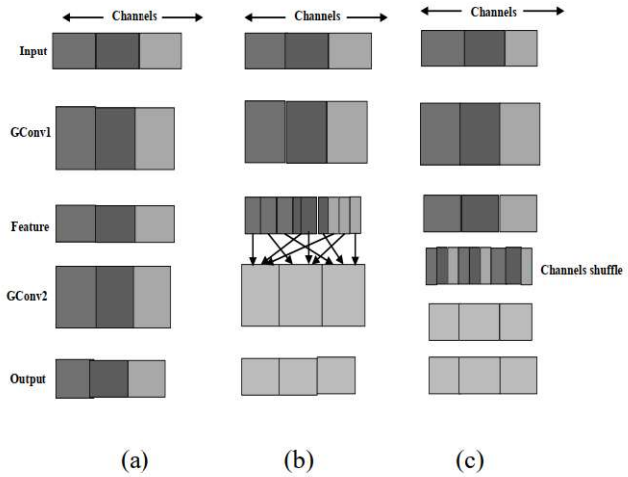

Fig 5. Channel shuffle schematic diagram. (a) Grouped convolution. (b) Apply shuffle. (c) Shuffle effect.

The Shapely Adaptive Explanation (SAE) tool is used to analyze the value of the characteristics and offer the explained classification results to increase confidence in the created AI model. We employed the NLG approach, which utilized the SILT model, to explain in understandable natural language to further increase the explanation's understandability. The cloud server conducts local risk assessment based on effect and feasibility levels into two categories, such as low and high risk, respectively, using the classification findings and natural language explanations. To limit the danger and probability of a repeat assault, the suggested work conducts intimation of analyzed findings to the surrounding IDSs (i.e., hopping the IDS result to close servers).

C. FL-based Global Cyber Risk Assessment

After analyzing and mitigating the local cyber risk, the proposed work tends to update and evaluate the total global risk to further enhance the security of the smart organization environment. To do this, we employed FL technology, in which the local risk brought about by smart organizations is handled as the local model, and the global model is produced in the CRAL layer by the Cyber Threat Repository (CTR). The CTR contains examples of various cyberattacks, along with information on the danger involved. By using blockchain technology, CTR secures weights and aggregates the locally created models. To strengthen the smart organization against the effects of global cybercrime, the aggregated findings are subsequently given to all local stakeholders. A cyber-risk-free environment may be created by conducting a global risk assessment, which makes it easy to identify and manage the risk of high-potential and malicious cyberattacks.

a. Federated Learning

Federated learning (FL) is used to train AI algorithms on a range of devices, such as smartphones, IoT devices, and other scattered cloud services. FL enables local data training and includes secure methods to preserve data from IoT devices. In IoT systems, FL may be a useful method for conducting global cyber risk assessments. FL can help in addressing some of the issues related to gathering and processing private and secure IoT data from various sources. FL is an effective approach for boosting the security of networked devices and systems since it can provide a privacy-preserving, scalable, and flexible solution to global cyber risk assessment in IoT ecosystems. Integrating blockchain and FL makes it possible to protect the trained models' integrity and prevent model poisoning attacks. Machine learning vulnerabilities come in two flavors: data poisoning and model poisoning. Due to FL's lack of information about the local models' creation, FL is more susceptible to model poisoning and model substitution assaults. Because they provide the attacker quick access to the features of the global model without needing access to every user's local model, model poisoning attacks are more effective than data poisoning attacks. To calculate the global model in the FL process, the central server combines the local models as shown below:

$$X^{s+1}=X^{s}+\frac{\eta}{q}\sum_{k=1}^{p}(Y_k^{s+1}-X^{s}) \quad (12)$$

where s is the communication round, p is the subset of chosen users, X^s is the existing global model, Y_k^{s+1} is the new local model, η is the learning rate, and q is the overall user count. In model replacement, the attacker tries to replace the global model X^{s+1} with a malicious model Z.

$$Z=X^{s}+\frac{\eta}{q}\sum_{k=1}^{p}(Y_k^{s+1}-X^{s}) \quad (13)$$

V. EXPERIMENTAL RESULTS

The experimental analysis of the suggested IoT cyber threat detection method is shown in this section. The results demonstrate the excellent efficiency of the proposed strategy. This sub-section includes the simulation setup, comparison analysis, and research summary.

A. Simulation setup

This sub-section explains the simulation setup and environment for cyber threat detection in IoT. Table II represents the system configuration.

Table II. System specification

Hardware specification	Hard disk	500GB
	RAM	minimum 2GB
Software specification	Simulation tools	ns-3.26
	Processor	2.5 GHz and above
	OS	ubuntu-16.04 LTS (64-bit)

The proposed work is simulated by ns-3.26. Table III represents the simulation configuration.

Table III. Simulation parameters

Parameters	Values
IoT devices	100
Authentication server	1
Edge server	2
Cloud server with blockchain	1
Cyber threat repository server	1
Number of rounds	300
Number of attacks	10
Interval	1.0
Packet size	1024
Distance	600
Number of nodes	100
Mobility Speed	1

B. Comparative analysis

The proposed model is contrasted with other existing methods like SHAP (SHapley additive exPlanations) algorithm, XGBoost, hybrid Cuda-Deep Neural Network Long short-term memory (CuDNNLSTM), and Cuda-Deep Neural Network Gated Recurrent Unit (CuDNNGRU) algorithms. The proposed model is analyzed by malicious traffic, detection rate, false positive rate, and prevention accuracy metrics graphically. In the proposed framework, the understandability analysis metric helps to identify and prevent cyber threat, thus securing the IoT devices. Examining the security and integrity of communication between IoT devices, gateways, and backend systems is the objective of the secure communication analysis metric. They are essential to guaranteeing the safety of IoT devices and the data they generate, transfer, and analyze.

a. Malicious traffic

Network security is provided by anomaly detection systems, which can correctly identify malicious network traffic. In the framework of IoT, malicious traffic refers to network communications or data transfers produced by corrupted or malicious IoT devices. These malicious activities can have a variety of harmful intentions and effects.

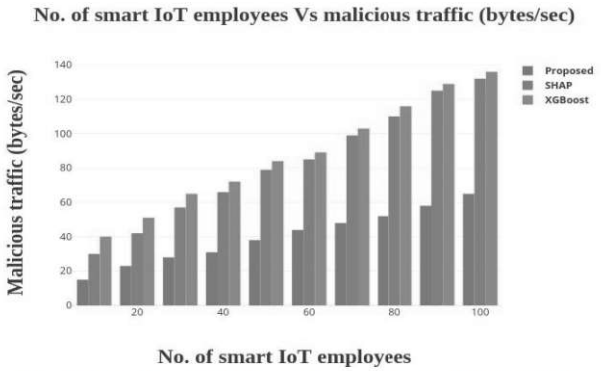

Fig 6. No. of smart IoT employees Vs malicious traffic

Fig.6 shows the malicious traffic comparison of the proposed method to existing methods. Malicious traffic is 132 bytes/sec for SHAP, 136 bytes/sec for XGBoost, and 65 bytes/sec for the proposed method. As a result, it is found that the proposed method has lower malicious traffic when compared to existing methods.

b. Detection Rate (DR)

It determines the model's harmful activity count by dividing it by the total number of activities identified in the test collection. It is sometimes referred to as recall. The equation (14) for determining DR is as follows:

$$DR=\frac{TP}{FN+TP} \tag{14}$$

The following are the different parameters that are utilized to calculate DR values: TP= True Positive, FN= False Negative.

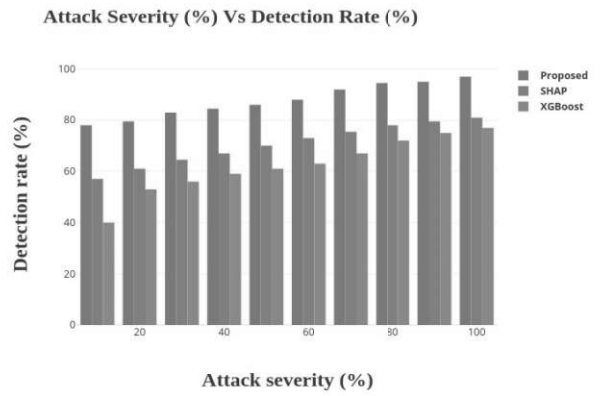

Fig 7. Attack severity vs. detection rate (%)

Figure 7 demonstrates the DR comparison between the suggested methodology with existing approaches such as SHAP and XGBoost. In terms of DR metric, the proposed method has acquired 97%, SHAP 81%, and XGBoost 77%. Overall, the suggested framework has demonstrated a higher detection rate.

c. False positive Rate (FPR)

The total number of legitimate network traffic observations that the model misclassifies as malicious activity is calculated. False positives may be an issue since they can cause unnecessary warnings or actions, interrupting business as usual and sometimes overloading security professionals with false alarms.

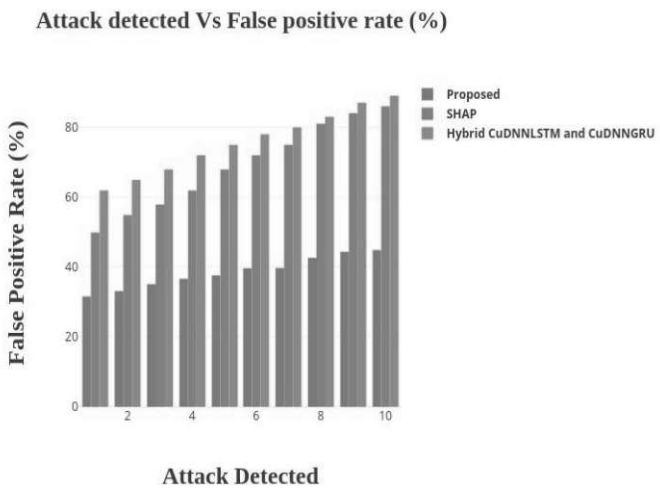

Fig 8. Attack detected vs. false positive rate (%)

The false positive rate is shown in Fig 8. The proposed techniques are compared with existing approaches. From Fig. 8 it is noticed that SHAP and hybrid methods show FPR of 86% and 89%, whereas the proposed method shown an FPR of 45% respectively. In IoT attack detection, a low false positive rate is a desirable result since it shows accurate threat identification. The proposed method is very effective in attack identification.

d. Prevention Accuracy (PAC)

It determines the proportion of successfully detected occurrences by the model to all of the observations included in the test collection. When determining the model's accuracy, TP and TN are both taken into consideration. The equation (15) for determining accuracy is as follows:

$$PAC=\frac{TP+TN}{TN+FN+TP+FP} \tag{15}$$

There are various parameters used to calculate PAC values that are given as follows: TN=True Negative, FP=False Positive, TP= True Positive, FN= False Negative.

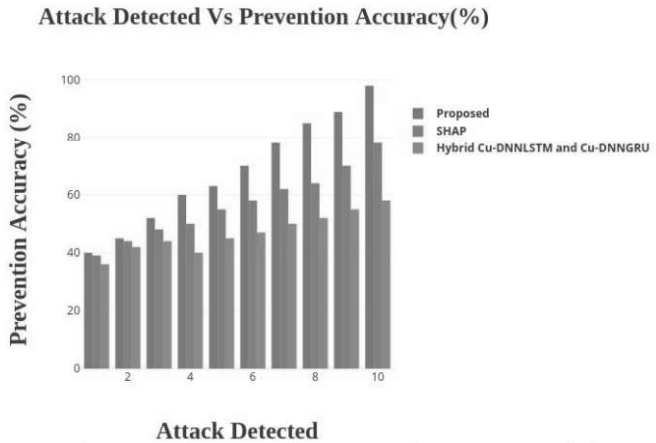

Fig 9. Attack Detected Vs Prevention Accuracy (%)

Fig 9 shows the result of the accuracy comparison. The proposed method will get the maximum accuracy of 98%, whereas existing methods such as SHAP and hybrid algorithms get an average of 39% and 36%. High prevention accuracy indicates excellent threat protection. Therefore, the proposed approach has the highest prevention accuracy compared to existing methods which helps reduce the attacks on IoT devices.

C. Research summary

In this research, we create a network topology. It consists of 100 - IoT devices,1-Authentication Server, 2- Edge servers, 1- Cloud server with blockchain and 1- Cyber Threat Repository server. Initially, the Authentication Server registers the employee's IoT devices with a user name, ID, password, fingerprint, and unique eye blink and is stored in the blockchain. During the authentication fingerprint and blink patterns are processed using the Capsule Network (CapsNet) algorithm and detect the attacks. Next, perform the data communication based on Quantum Key Secure Communication Protocol (QKSCP) to reduce the vulnerabilities in the communication channel. Next, in Edge server performs the attack classifications based on the Dual Q Network-based Asynchronous Advantage Actor Critic algorithm (DQN-A3C) process. Next, in the cloud server perform the network traffic classification by using Harmonized Google Net (HGN). Next, the Cyber Threat Repository (CTR) server generates the attack detection-based local model and stores it in the blockchain. Plot the results graph for No. of Smart IoT employees vs. malicious Traffic, Attack Severity vs. detection Rate, Attacks Detected vs. false Positive Rate and Attack Detected Vs Prevention Accuracy. This sub-section deals with the discussion of the performance of the proposed approach. Fig (6) – (9) shows the graphical depiction of comparative results and the below table IV provides the numerical results of the comparative analysis.

Table IV. Numerical outcomes

Performance metrics	Proposed	SHAP	XGBoost	Hybrid CuDNNLSTM + CuDNNGRU
Malicious traffic (bytes/sec)	65	132	136	-
Detection rate (%)	97	81	77	-
False positive rate (%)	45	86	-	89
Prevention accuracy (%)	98	39	-	36

VI. CONCLUSION

The main goal of our research is to increase the security and privacy of IoT-enabled smart organizations. Furthermore, we also address the problems of Amplified Cyber Threats, High Attack Reoccurrence Possibility, and Elevated False Positive Rates. Three layers, including an IoTL, CL, and CRAL, make up the proposed work. Employees using IoT devices are part of the IoTL. The cloud and edge servers that make up the CL are used to validate network traffic as well as execute local risk assessments. The edge server is used to monitor an employee's human aspects. Last but not least, the CRAL is used to inform the underlying local parties by the outcomes of the global risk analysis. We perform Triple triple-strand authentication, Social Engineering Attack Detection and prevention, Hopping IDS & IPS, and Global Cyber Risk Assessment. By using the simulation tools ns-3.26, and ubuntu 16.04 the proposed model was tested. The proposed strategy is evaluated by assessing the new technique to the existing ones in terms of malicious traffic with 65 bytes/sec, detection rate with 97%, false positive rate with 45%, and prevention accuracy with 98%. Numerical analysis is used to examine the performance of our strategy, and it can be shown that it performs better than other approaches in every measure.

REFERENCES

1. Tabassum, S., Parvin, N., Hossain, N., Tasnim, A., Rahman, R., & Hossain, M. I. (2022, December). IoT Network Attack Detection Using XAI and Reliability Analysis. In *2022 25th International Conference on Computer and Information Technology (ICCIT)* (pp. 176-181). IEEE.
2. Patil, S., Varadarajan, V., Mazhar, S. M., Sahibzada, A., Ahmed, N., Sinha, O., ... & Kotecha, K. (2022). Explainable artificial intelligence for the intrusion detection system. *Electronics*, *11*(19), 3079.
3. Huong, T. T., Bac, T. P., Ha, K. N., Hoang, N. V., Hoang, N. X., Hung, N. T., & Tran, K. P. (2022). Federated learning-based explainable anomaly detection for industrial control systems. *IEEE Access*, *10*, 53854-53872.
4. Yazdinejad, A., Dehghantanha, A., Parizi, R. M., Hammoudeh, M., Karimipour, H., & Srivastava, G. (2022). Block hunter: Federated learning for cyber threat hunting in blockchain-based iiot networks. *IEEE Transactions on Industrial Informatics*, *18*(11), 8356-8366.

5. Jagatheesaperumal, S. K., Pham, Q. V., Ruby, R., Yang, Z., Xu, C., & Zhang, Z. (2022). Explainable AI over the Internet of Things (IoT): Overview, state-of-the-art and future directions. *IEEE Open Journal of the Communications Society*.
6. Zhang, Z., Al Hamadi, H., Damiani, E., Yeun, C. Y., & Taher, F. (2022). Explainable artificial intelligence applications in cyber security: State-of-the-art in research. *IEEE Access*.
7. Tyagi, A. K., Dananjayan, S., Agarwal, D., & Thariq Ahmed, H. F. (2023). Blockchain—Internet of Things Applications: Opportunities and Challenges for Industry 4.0 and Society 5.0. *Sensors*, *23*(2), 947.
8. Mohanta, B. K., Jena, D., Ramasubbareddy, S., Daneshmand, M., & Gandomi, A. H. (2020). Addressing security and privacy issues of IoT using blockchain technology. *IEEE Internet of Things Journal*, *8*(2), 881-888.
9. Kumar, R., Kumar, P., Tripathi, R., Gupta, G. P., Garg, S., & Hassan, M. M. (2022). A distributed intrusion detection system to detect DDoS attacks in blockchain-enabled IoT network. *Journal of Parallel and Distributed Computing*, *164*, 55-68.
10. Ibrahim, R. F., Abu Al-Haija, Q., & Ahmad, A. (2022). DDoS attack prevention for Internet of Thing devices using Ethereum blockchain technology. *Sensors*, *22*(18), 6806.
11. Goyal, S. B., Rajawat, A. S., Solanki, R. K., Zaaba, M. A. M., & Long, Z. A. (2023, April). Integrating AI With Cyber Security for Smart Industry 4.0 Application. In *2023 International Conference on Inventive Computation Technologies (ICICT)* (pp. 1223-1232). IEEE.
12. Shafiq, M., Gu, Z., Cheikhrouhou, O., Alhakami, W., & Hamam, H. (2022). The rise of "Internet of Things": review and open research issues related to detection and prevention of IoT-based security attacks. *Wireless Communications and Mobile Computing*, *2022*, 1-12.
13. Chiba, Z., Abghour, N., Moussaid, K., Lifandali, O., & Kinta, R. (2022, September). Review of Recent Intrusion Detection Systems and Intrusion Prevention Systems in IoT Networks. In *2022 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)* (pp. 1-6). IEEE.
14. Ali, S., Abuhmed, T., El-Sappagh, S., Muhammad, K., Alonso-Moral, J. M., Confalonieri, R., ... & Herrera, F. (2023). Explainable Artificial Intelligence (XAI): What we know and what is left to attain Trustworthy Artificial Intelligence. *Information Fusion*, *99*, 101805.
15. Li, T., Song, C., & Pang, Q. (2023). Defending against social engineering attacks: A security pattern-based analysis framework. *IET Information Security*.
16. Yazdinejad, A., Dehghantanha, A., Parizi, R. M., Hammoudeh, M., Karimipour, H., & Srivastava, G. (2022). Block hunter: Federated learning for cyber threat hunting in blockchain-based iiot networks. *IEEE Transactions on Industrial Informatics*, *18*(11), 8356-8366.
17. Otoum, Y., Liu, D., & Nayak, A. (2022). DL-IDS: a deep learning–based intrusion detection framework for securing IoT. *Transactions on Emerging Telecommunications Technologies*, *33*(3), e3803.
18. Yadav, N., Pande, S., Khamparia, A., & Gupta, D. (2022). Intrusion detection system on IoT with 5G network using deep learning. *Wireless Communications and Mobile Computing*, *2022*, 1-13.
19. Nie, L., Wu, Y., Wang, X., Guo, L., Wang, G., Gao, X., & Li, S. (2021). Intrusion detection for secure social internet of things based on collaborative edge computing: a generative adversarial network-based approach. *IEEE Transactions on Computational Social Systems*, *9*(1), 134-145.
20. Nie, L., Sun, W., Wang, S., Ning, Z., Rodrigues, J. J., Wu, Y., & Li, S. (2021). Intrusion detection in green internet of things: a deep deterministic policy gradient-based algorithm. *IEEE Transactions on Green Communications and Networking*, *5*(2), 778-788.
21. Burda, P., Allodi, L., & Zannone, N. (2021, September). Dissecting social engineering attacks through the lenses of cognition. In *2021 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)* (pp. 149-160). IEEE.
22. Mughaid, A., AlZu'bi, S., Hnaif, A., Taamneh, S., Alnajjar, A., & Elsoud, E. A. (2022). An intelligent cyber security phishing detection system using deep learning techniques. *Cluster Computing*, *25*(6), 3819-3828.
23. Indrasiri, P. L., Halgamuge, M. N., & Mohammad, A. (2021). Robust ensemble machine learning model for filtering phishing URLs: Expandable random gradient stacked voting classifier (ERG-SVC). *IEEE Access*, *9*, 150142-150161.
24. Tang, L., & Mahmoud, Q. H. (2021). A deep learning-based framework for phishing website detection. *IEEE Access*, *10*, 1509-1521.
25. Ravi, V., Chaganti, R., & Alazab, M. (2022). Recurrent deep learning-based feature fusion ensemble meta-classifier approach for intelligent network intrusion detection system. *Computers and Electrical Engineering*, *102*, 108156.
26. Makkar, A., Ghosh, U., Sharma, P. K., & Javed, A. (2021). A fuzzy-based approach to enhance cyber defence security for next-generation IoT. *IEEE Internet of Things Journal*.
27. Alharbi, A., Alosaimi, W., Alyami, H., Rauf, H. T., & Damaševičius, R. (2021). Botnet attack detection using local global best bat algorithm for industrial internet of things. *Electronics*, *10*(11), 1341.
28. Abdalzaher, M. S., Elwekeil, M., Wang, T., & Zhang, S. (2021). A deep autoencoder trust model for mitigating jamming attack in IoT assisted by

cognitive radio. *IEEE Systems Journal*, *16*(3), 3635-3645.
29. Popoola, S. I., Ande, R., Adebisi, B., Gui, G., Hammoudeh, M., & Jogunola, O. (2021). Federated deep learning for zero-day botnet attack detection in IoT-edge devices. *IEEE Internet of Things Journal*, *9*(5), 3930-3944.
30. Sarhan, M., Lo, W. W., Layeghy, S., & Portmann, M. (2022). IIBFL: A hierarchical blockchain-based federated learning framework for collaborative IoT intrusion detection. *Computers and Electrical Engineering*, *103*, 108379.
31. Spathoulas, G., Giachoudis, N., Damiris, G. P., & Theodoridis, G. (2019). Collaborative blockchain-based detection of distributed denial of service attacks based on internet of things botnets. *Future Internet*, *11*(11), 226.
32. Ethala, S., & Kumarappan, A. (2022). A Hybrid Spider Monkey and Hierarchical Particle Swarm Optimization Approach for Intrusion Detection on Internet of Things. *Sensors*, *22*(21), 8566.
33. Abou El Houda, Z., Brik, B., & Khoukhi, L. (2022). "why should i trust your ids?": An explainable deep learning framework for intrusion detection systems in internet of things networks. *IEEE Open Journal of the Communications Society*, *3*, 1164-1176.
34. Shah, H., Shah, D., Jadav, N. K., Gupta, R., Tanwar, S., Alfarraj, O., ... & Marina, V. (2023). Deep learning-based malicious smart contract and intrusion detection system for IoT environment. *Mathematics*, *11*(2), 418.
35. Regan, C., Nasajpour, M., Parizi, R. M., Pouriyeh, S., Dehghantanha, A., & Choo, K. K. R. (2022). Federated IoT attack detection using decentralized edge data. *Machine Learning with Applications*, *8*, 100263.
36. Fakieh, A., & Akremi, A. (2022). An Effective Blockchain-Based Defense Model for Organizations against Vishing Attacks. *Applied Sciences*, *12*(24), 13020.
37. Jahromi, A. N., Karimipour, H., Dehghantanha, A., & Choo, K. K. R. (2021). Toward detection and attribution of cyber-attacks in IoT-enabled cyber–physical systems. *IEEE Internet of Things Journal*, *8*(17), 13712-13722.
38. Kalutharage, C. S., Liu, X., Chrysoulas, C., Pitropakis, N., & Papadopoulos, P. (2023). Explainable AI-based DDOS attack identification method for IoT networks. *Computers*, *12*(2), 32.
39. Le, T. T. H., Kim, H., Kang, H., & Kim, H. (2022). Classification and explanation for intrusion detection system based on ensemble trees and SHAP method. *Sensors*, *22*(3), 1154.
40. Javeed, D., Gao, T., & Khan, M. T. (2021). SDN-enabled hybrid DL-driven framework for the detection of emerging cyber threats in IoT. *Electronics*, *10*(8), 918.
41. Kumar, P., Gupta, G. P., & Tripathi, R. (2021). Toward design of an intelligent cyber attack detection system using hybrid feature reduced approach for iot networks. *Arabian Journal for Science and Engineering*, *46*, 3749-3778.