IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 21, 2021, accepted October 28, 2021, date of publication November 1, 2021, date of current version November 9, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3124634

Asynchronous Peer-to-Peer Federated Capability-Based Targeted Ransomware Detection Model for Industrial IoT

MUNA AL-HAWAWREH, ELENA SITNIKOVA, (Member, IEEE), AND NEDA ABOUTORAB, (Senior Member, IEEE)
School of Engineering and Information Technology, University of New South Wales (UNSW), Campbell, ACT 2612, Australia
Corresponding author: Muna Al-Hawawreh (m.al-hawawreh@student.adfa.edu.au)

ABSTRACT Industrial Internet of Thing (IIoT) systems are considered attractive ransomware targets because they operate critical services that affect human lives and have substantial operational costs. The major concern is with brownfield IIoT systems since they have legacy edge systems that are not fully prepared to integrate with IoT technologies. Various existing security solutions can detect and mitigate such attacks but are often ineffective due to the heterogeneous and distributed nature of the IIoT systems and their interoperability demands. Consequently, developing new detection solutions is essential. Therefore, this paper proposes a novel targeted ransomware detection model tailored for IIoT edge systems. It uses Asynchronous Peer-to-Peer Federated Learning (AP2PFL) and Deep Learning (DL) techniques as a targeted ransomware detection algorithm. The proposed model consists of two modules: 1) Data Purifying Module (DPM) aims to refine and reconstruct a valuable and robust representation of data based on Contractive Denoising Auto-Encoder (CDAE), and 2) Diagnostic and Decision Module (DDM) is used to identify targeted ransomware and its stages based on Deep Neural Network (DNN) and Batch Normalization (BN). The main strengths of this proposed model include: 1) each edge gateway's modules work cooperatively with its neighbors in an asynchronous manner and without a third party, 2) it deals with both homogeneous and heterogeneous data, and 3) it is robust against evasion attacks. An exhaustive set of experiments on three datasets prove the high effectiveness of the proposed model in detecting targeted ransomware (known and unknown attacks) in brownfield IIoT and the superiority over the state-of-the-art models.

INDEX TERMS Edge system, IIoT, federated learning, detection, targeted ransomware.

I. INTRODUCTION

With the emergence of the Internet of Things (IoT), digitization has increasingly become more prevalent in the industrial space. With a major focus on Machine-to-Machine (M2M) communications and Artificial Intelligence (AI)-based data analytics while realising Quality of Service (QoS), the Industrial IoT (IIoT) enables devices and machines of different vendors and generations to communicate and be highly efficient, productive and reliable [1]–[3]. Many IIoT implementations have followed the brownfield approach in which legacy systems co-exist with new IoT technologies [4]. To facilitate these systems' interoperability and convergence using IoT technologies, new devices such as edge gateways are deployed as bridges between legacy Operational Technology (OT) and new IoT ones. However, this advance and the tight integration between OT and IoT also comes with cyber-risks [1] as new entry points have opened the way for more sophisticated attacks, such as ransomware ones, that target these critical devices and systems [4].

A recent trend observed in ransomware attacks is using targeted and double-extortion ones [5] which behave similarly to Advanced Persistent Threats (APTs) as they follow multiple stages and cause as much damage as possible through several harmful actions to increase the size of the ransom payment [4]; for example, attackers can exfiltrate and encrypt critical data, deny access to these systems and damage their physical processes before demanding a ransom fee [5]. One high-profile recent ransomware incident affected Colonial Pipeline. It was the most significant cyber-attack on an American power system, whereby attackers gained

The associate editor coordinating the review of this manuscript and approving it for publication was Vyasa Sai.

control of more than 100 Gigabytes of information which led to the fuel distribution network being shut down for a week [6]. Other ransomware events associated with industrial systems are Ryuk, REvil, Ekans, LockerGoga and Snake attacks which have proven their capabilities to spread from Information Technology (IT) to OT networks [7]. Such incidents demonstrate that IIoT systems are very likely to be the most ongoing targets for ransomware actors [7], [8].

Industrial and cybersecurity agencies and vendors pay particular attention to ransomware attacks, with new intelligence related to their attackers' tactics, techniques and procedures reported and awareness provided from time to time to prevent them [8]. However, a prevention technique is not always the appropriate solution as attackers can continuously develop new strategies and find new ways of bypassing the perimeters of defences to achieve their goals. This has increased the interest of researchers in addressing such advanced and multi-stage attacks using Artificial Intelligence (AI)-based detection models to provide an effective and robust security posture [2], [9]. Although there are many AI-based ransomware detection models [8], [10]–[13] most cannot be directly applied to a brownfield IIoT system, which has a distributed, and heterogeneous nature and interoperability demand, for the following main reasons: 1) As they were designed for Windows and Linux Operating Systems (OSs), they do not work for IIoT edge gateway devices operating on proprietary hardware and software and connectivity protocols, 2) They focus on handling crypto-ransomware but not the current trends of ransomware attacks and their activities, 3) They are isolated and can easily be bypassed by evasion attacks or new attack tactics and techniques as they are not capable of learning the ongoing ones faced by their peers, 4) Most existing IIoT intrusion detection models depend on a cloud server which faces security and privacy issues while moving data from edge devices, and 5) Brownfield IIoT systems have many distributed edge gateways designed to provide less communication with cloud servers to reduce bandwidth and network latency, and encounter high levels of cloud disconnection in reality.

Consequently, the development of a new AI-based detection model tailored for the edge gateways of brownfield IIoT systems is essential. Motivated by this, we propose a new model for detecting targeted ransomware attacks against the edge gateways of brownfield IIoT systems. It is based on Federated Learning (FL) and Deep Learning (DL) techniques. The former is a new learning paradigm that splits data collection and model training via multi-party computation and model aggregation. Considerable work recently conducted in the field of FL [2], [14]–[16] shows a trend of shifting from pooling or isolated detection models to client-server cooperative ones using FL. However, as most FL-based detection models follow client-server and synchronous communication approaches, they are not suitable for the edge gateways of brownfield IIoT systems. This is because these gateways are designed to operate time-sensitive processes and provide less communication with cloud servers to reduce bandwidth and network latency [17]–[20]. Furthermore, the existing models are less robust against heterogeneous data (i.e.,non-Identically Independent Distribution (non-IID) as they highly depend on homogeneous training data (i.e., IID). Therefore, we propose a new FL model that depends on asynchronous and Peer-to-Peer (P2P) communications among connected edge gateways to build a comprehensive targeted ransomware detection model in a privacy-preserving way.

The main contributions of this paper are as follows.

1) We propose the first-of-its-kind targeted ransomware detection model tailored for IIoT edge gateways. It employes Asynchronous Peer-to-Peer Federated Learning (AP2PFL) and Deep Learning (DL) techniques as a targeted ransomware detection algorithm and includes IID and non-IID learning models.
2) We propose and design new Deep Learning (DL)-based model for revealing targeted ransomware in IIoT edge gateway. It consists of a Data Purifying Module (DPM) and Diagnosis and Decision Module (DDM).
3) We design and develop a hybrid Auto-Encoder (AE) algorithms to power a DPM. We present a Contractive Denoising Auto-Encoder (CDAE) for refining and reconstructing a valuable and relevant representation of the input data. DPM helps to build a robust decision process and improve performances against evasion attacks.
4) We also present DDM to reveal targeted ransomware attacks at edge gateways. It is based on Deep Neural Network DNN) with a Batch Normalization technique.
5) We conduct an exhaustive set of experiments for validating the proposed model on the X-IIoTID, ISOT, and NSL-KDD datasets.
6) Finally, we evaluate the robustness of the proposed model using white- and black-box evasion attack techniques, whereby targeted ransomware attacks change their behavior to appear as legitimate ones.

The remainder of this paper is structured as follows. Section II explains the existing ransomware detection and Federated Learning (FL) models. In Section III, the system, threat models and data representation are described. This is followed by the proposed model in Section IV and the performance evaluation in Section V. Lastly, Section VI concludes the paper.

II. RELATED WORK

This section briefly highlights state-of-the-art studies that focused on ransomware detection models based on network traffic and FL intrusion detection models for an IIoT/IoT system. A comparison of them is presented in Table 1.

A. RANSOMWARE DETECTION MODELS

The interest in developing ransomware detection models has been increasing in recent years. Many studies focus on network traffic to detect ransomware attacks; for example, Almashhadani *et al.* [10] proposed a multi-classifier model that depended on the features of the HTTP and DNS

TABLE 1. Comparison of different intrusion detection models.

Model	Ransomware	Asynchronous	P2P	IID	Non-IID	FL	Evasion	Accuracy% (binary-class)	Used for
Almashhadani et al.[10]	✓	✗	✗	✗	✗	✗	✗	> 98.50	IT
Almashhdani et al.[11]	✓	✗	✗	✗	✗	✗	✗	97.82	IT
Piskozub et al.[21]	✓	✗	✗	✗	✗	✗	✗	89.00	IT
Alhawi et al.[12]	✓	✗	✗	✗	✗	✗	✗	97.10	IT
Morato et al.[22]	✓	✗	✗	✗	✗	✗	✗	100	IT
Modi et al.[23]	✓	✗	✗	✗	✗	✗	✗	98.00	IT
Kozik et al. [13]	✓	✗	✗	✗	✗	✗	✗	> 91.00	IT
Modi et al.[9]	✗	✗	✗	✓	✗	✗	✗	-	IT
Rey et al.[24]	✗	✗	✗	✓	✗	✓	✗	> 99.00	IoT
Nguyen et al.[25]	✗	✗	✗	✓	✓	✓	✗	> 95.00	IoT
Hei et al.[26]	✗	✗	✗	✓	✗	✓	✗	90.80	IoT
Liu et al.[2]	✗	✗	✗	✓	✗	✓	✗	97.25	IIoT
Mowla et al.[14]	✗	✗	✗	✓	✗	✓	✗	> 82.00	FANET
Li et al.[27]	✗	✗	✗	✓	✗	✓	✗	99.20	IIoT
Taheri et al.[28]	✗	✗	✗	✓	✗	✓	✗	> 89.00	IIoT
Schneble and Thamilarasu [29]	✗	✗	✗	✓	✗	✓	✗	99.00	CPS
Proposed model	✓	✓	✓	✓	✓	✓	✓	> 97.03	IIoT

protocols to reveal crypto-ransomware attacks. Their experiments showed that a Random Tree (RT) achieved better accuracy (98.72%) than its peers (e.g., Random Forest (RF), and Support Vector Machine (SVM)) at the packet level while Naive Base (NB) (99.83%) was the best at the flow level. In related work, Almashhdani *et al.* [11] used features from a DNS request packet (i.e., domain-name characters) and the randomness measure algorithm to reveal a malicious domain using multi-detectors. Their proposed model achieved an accuracy of 97.82%. These models assumed that domain names with random characters are malicious and indicate a ransomware attack. It is known that domains generated by a dynamic generation algorithm have the highest levels of randomness. Although these detection models could significantly detect attacks, they failed to discover targeted ransomware using a legal domain server or another C&C technique. Piskozub *et al.* [21] proposed MalAlert, a detection model based on the RF algorithm. It adopted the number of transmitted bytes as the critical statistical network feature for detecting crypto-ransomware. Their main contribution was an approach for collecting and aggregating ransomware network traffic flows into several flow-sets and then extracting features for each set. Also, to preserve users' privacy, these features were based on only the number of bytes transmitted and were IP address- and port-agnostic. However, these features are insufficient to detect ransomware attacks with legitimate network traffic or without any network activity. Also, their proposed model's generated high false alarms and had a low ransomware detection rate.

Alhawi *et al.* [12] introduced NetConverse, a machine learning-based model that depended greatly on conversations between crypto-ransomware and the features of a C&C server network. A Decision Tree (DT) approach achieved better accuracy (97.10%) than peer techniques, such as the SVM and RF. The key limitation of this work is that the manual features selection process and the human intervention make their model unsuitable for the real environment. Morato *et al.* [22] developed an algorithm for inspecting the traffic of an SMB protocol to extract statistical features related to sharing files and then used a predefined threshold to detect crypto-ransomware. The model achieved high results (roughly 100%) due to the fact that it was specific for SMB protocol's activity and tested using few samples of ransomware attacks. However, detecting ransomware using a predefined threshold is a significant challenge. This is because of the dynamic behavior and the evolving techniques of ransomware attacks.

Moreover, Modi *et al.* [23] focused on HTTPS traffic and machine learning (e.g., RF) to detect crypto-ransomware, demonstrating the feasibility of using encrypted network traffic to detect ransomware. However, the key issue is the limited number of network traffic flows that were tested. Akbanov *et al.* [9] concentrated on inspecting packets and matching them with malicious IPs and ports. One shortcoming of their proposed models is that the new pattern of ransomware attacks that rely on different protocols for lateral movement and C&C might not be identified precisely. A time windows embedding solution whereby network traffic flows were grouped based on a specific time window to extract features was proposed by Kozik *et al.* [13]. Their proposed model relied on the transformer's encoder followed by a fully connected feed-forward neural network for classification. Although their proposed model achieved better accuracy than the classical machine learning algorithms, the customizing process for the transformer's parameters is a significant challenge that should be handled before deploying the model in real-world environments.

Although these existing models use network traffic to detect ransomware, they rely on specific features extracted from HTTPS and DNS packets. This makes them unsuitable for a brownfield IIoT system with heterogeneous devices, connectivity and messaging protocols and interoperability demand. Also, as these models are isolated/centralized and cannot learn about the ongoing attacks faced by their peers, new ransomware versions and evasion attacks can easily bypass them. Most existing ransomware-based detection models use classical machine learning techniques with few generalization capabilities and heavily depend on manual feature engineering. They often generate high false alarms, fail to detect new attack patterns and deal with high-volume and -speed IIoT network traffic. They are dependent on moving data to the central server, exposing them to privacy and security problems. Unlike them, the model proposed in this paper addresses all these issues. It handles the targeted ransomware attacks and their full stages. It also utilizes federated and deep learning techniques to deal efficiently with IIoT network and system activities and protect the distributed edge gateways against known and unknown targeted ransomware attacks in a privacy-preserving manner.

B. FEDERATED LEARNING (FL)-BASED INTRUSION DETECTION MODELS

FL has emerged as a promising technique for collaboratively learning a shared model while preserving data privacy. In particular, many researchers have recently used it to develop intrusion detection models; for example, Rey *et al.* [24] proposed a client-server FL model using a DNN. The proposed model obtained a high accuracy (99.00%), mainly because of the used dataset, that is, N-BaIoT, which is known as a very easy and less complex dataset. However, the performance significantly dropped under the adversarial attacks, showing the need for more robust countermeasures. Also, Nguyen *et al.* [25] used a Gated Recurrent Unit (GRU) for detecting Mirai malware in an IoT network. Although this model obtained a good performance in detecting IoT malware (i.e., accuracy = 95%), its performance was significantly reduced under adversarial examples due to the aggregation function's lack of resiliency and robustness.

Similarly, Li *et al.* [27] designed a client-server FL detection model based on a Convolutional Neural Network (CNN) and GRU to produce new data, which it passed as new features to a DNN. An attention CNN-Long Short-Term Memory (LSTM) model within an FL framework was presented by Liu *et al.* [2] for detecting anomalies in IIoT edge devices. These models are complex and require much longer calculation and training times. Schneble and Thamilarasu [29] presented a Multi-Layer Perceptron (MLP) as the critical decision engine in an FL model deployed in mobile clients. The key limitation of this model is the need to perform an extensive management process and the organization of patients' groups by the cloud server before starting the FL technique. This is because each intrusion detection model was designed for patients with similar behaviors (i.e., IID data), and it needed accurate patients' clustering.

An FL detection model for revealing a jamming attack in a Flying Ad-hoc Network (FANET) was proposed by Mowla *et al.* [14]. They used the Dempster-Shafer theory to choose the best Unmanned Aerial Vehicle (UAV) client groups for calculating the global model update. Although using Dempster-Shafer theory achieved promising performance, the selection process caused a delay in the models' updating. Also, the selection process' performance was only tested using a small number of clients (=6), which is insufficient to demonstrate its effectiveness. Hei *et al.* [26] presented an FL detection model that focused on sharing features such as information related to each of its alerts. Their proposed model achieved promising results; however, it had challenges with blockchain storage and further improvement on the performance is also required. To create a robust detection model, Taheri *et al.* [28] used a Generative Adversarial Network (GAN) to detect Android malware in IIoT systems by generating adversarial examples that poisoned the FL training process. An anomaly-based threshold was used in their cloud server to reject combining these examples. Their proposed federated model with Byzantine Median (BM) and Byzantine Krum (BK) adversarial attacks defence mechanisms obtained an accuracy of 89.51% and 93.24% for the Gnome malware dataset. However, selecting the appropriate threshold is challenging, and it could be ineffective due to the heterogeneous nature of IIoT devices and network traffic.

Although these existing models offer promising FL solutions (as described in Table 1, they depend strongly on the client-server architecture in the particular cloud server, which has some drawbacks. A cloud server may pose a single point of failure, and these models use a synchronous protocol whereby participating users/devices must send their parameters simultaneously. In reality, synchronizing these devices is a complicated task that may also affect industrial operations, which are the most sensitive to time. Also, edge gateways in brownfield IIoT systems are designed to provide less communication with cloud servers to reduce bandwidth and network latency. However, as they encounter high levels of cloud disconnection, existing models are unsuitable for them. Furthermore, these models always assume that the collected data is homogeneous (i.e., IID) and less robust against evasion attacks. The proposed model considers all these aspects by providing a robust asynchronous peer-to-peer federated deep learning model for the edge gateways of brownfield IIoT systems.

III. SYSTEM, THREAT MODELS AND DATA REPRESENTATION

A. PROPOSED SYSTEM'S ARCHITECTURE

The proposed system architecture is illustrated in Figure 1. A brownfield IIoT system, in which many edge gateways are distributed in the one edge tier, is considered. These gateways connect with legacy field devices and SCADA on the OT sides and a cloud broker, mobile and enterprise devices

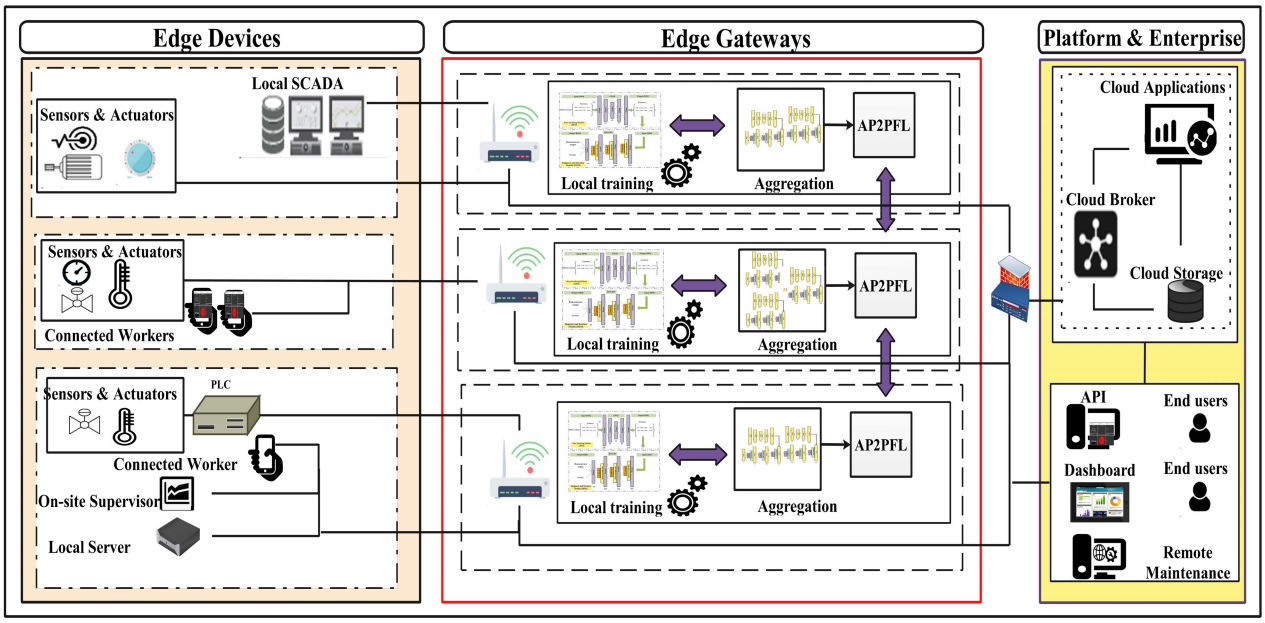


FIGURE 1. Proposed system's architecture.

on the others. They also act as master industrial devices in the system, each of which can communicate with its neighbors (i.e., other connected edge gateways) in a Peer-to-Peer (P2P) manner. The main goal of the detection model (i.e., DL-based model) in each edge gateway is to discover any targeted ransomware attacks, identify their activities and send an alert to the security team to respond appropriately. The main objective of the proposed model is to make all the detection models (i.e., DL-based models) in the distributed edge gateways identical and perform approximately the same irrespective of their personalized data using AP2PFL technique.

This proposed system consists of a group of IIoT edge gateways, each of which has the following two roles.

- **Training its local detection model-** each IIoT edge gateway monitors its connected devices in all its interfaces, collects data related to network and system activities and builds a local detection model (i.e., DL-based model) based on its own collected data. Then, it trains this model and updates its variables to identify targeted ransomware attacks in a brownfield IIoT system.
- **Aggregating and stacking its neighbor detection models-** each IIoT edge gateway is responsible for building a comprehensive detection model by aggregating and stacking the variables of its locally learned models' at its neighboring devices using AP2PFL. Multiple rounds of communications between each edge gateway and its neighboring devices can obtain the final best detection model (i.e., DL-based model) which is approximately identical for all connected IIoT edge gateways.

It is worth noting that edge gateway has only one detection model whereby its variables are updated locally using incoming local observations. Also, the same model's variables are updated using its neighboring devices at a random time.

B. ASSUMPTIONS AND THREAT MODEL

We focus on the full life-cycle of targeted ransomware attacks against brownfield IIoT systems. It includes, for example, reconnaissance (e.g., scanning vulnerabilities, discovering CoAP resources, and WebSocket fuzzing), weaponization (e.g., brute force and malicious insider), and exploitation (e.g., reverse shell and Man-in-the-Middle (MitM)). In addition to the lateral movement (e.g., Modbus register reading, MQTT cloud broker subscription, and TCP relay attacks), C&C, data exfiltration, tampering (e.g., false notifications and false data injection), Ransom Denial of Service (RDoS) and Crypto-Ransomware attacks. Since the proposed model has no third party (e.g., cloud server) included in its training and transformation processes, its framework has fewer security and privacy issues than a client-server FL one as its P2P communication eliminates the threat of its data being leaked, or privacy violated. Furthermore, as each edge gateway knows its neighbors in advance, it is protected against receiving a malicious model's variables. We assume that these connected edge gateways are honest and strictly follow the designed protocol in the updating model as well as exchange their parameters using encryption. Therefore, we focus on only evasion attacks as APT attackers are always keen to avoid being detected by converting ransomware observations to legitimate ones [30]. The following attacks are common evasion techniques that could be used by attackers to evade detection by machine and deep learning-based models.

- **Fast Gradient Sign Method (FGSM)-** in it, an attacker generates targeted adversarial examples that cause

targeted ransomware observations or instances to be classified as normal using the gradient of the loss function concerning the inputs. The gradient step is computed in the direction of the negative gradient with respect to the target class [31]. In this paper, we perform FGSM attack in a white-box approach in which the attacker has complete access to the DL-based model.

- **Brute Force (BF)-** in it, an attacker generates targeted adversarial examples that cause targeted ransomware observations or instances to be classified as normal using Gaussian noise instead of optimization or gradients. We execute the BF attacks in a black-box approach whereby the attacker does not have complete access to the model [32].

C. DATA REPRESENTATION

The collected data is partitioned based on class and considered in two distributions or learning models. The first data distribution is IID data, which is evenly distributed among the connected edge gateways. The second one is non-IID data, whereby each edge gateway has different data (e.g., different classes). The IID data in this work also has two cases. In the first, the binary-class's data is distributed evenly among edge gateways. Each edge gateway $((i),\ i = 1, 2, \ldots n)$ has its own dataset $(D_i = (a_i,\ c_i))$, where (a_i) denotes the normal (i.e.,legitimate) and ransomware observations for edge gateway (i) and $(c_i) = \{0, 1\}$ is the class of observation. In the second, the multiple-classes' data is distributed evenly among edge gateways, whereby it is assumed that each edge gateway $((i),\ i = 1, 2, \ldots n)$ has its own dataset $(D_i = (a_i,\ c_i))$ with many classes represent normal and targeted ransomware activities or stages $(c_i) = \{0, 1, 2, 3, 4, 5, C\ (C = \#classes)\}$. In the non-IID data, we assume that each edge gateway (i) faces different stages of targeted ransomware (i.e., non-IID). Therefore, each has different number of classes and observations (a_i) related to different classes (c_i), where $(c_i) \subset \{0, 1, 2, 3, 4 \ldots \mathrm{C}\}$. However, as we design an intrusion detection model, the normal observations are distributed among IIoT edge gateway devices (without overlapping).

IV. PROPOSED MODEL

In this section, the proposed model is elaborated on by firstly defining its workflow and then introducing the DL-based model designed for detecting targeted ransomware in each edge gateway.

A. WORKFLOW OF PROPOSED MODEL

The basic idea behind this proposed model is networking multiple IIoT edge gateways to collectively build identical targeted ransomware detection models (i.e., DL-based models) in all of them based on AP2PFL, as illustrated in Figure 1. A Primal-Dual Method of Multiplier-Stochastic Gradient Descent (PDDM SGD) is employed as an optimization and learning algorithm [33] which updates the models' variables in asynchronous P2P communications, as described in the following paragraphs and Algorithm 1.

- **Model initialization-** in this phase, each edge gateway selects an array of initial random weights for local DL-based model and values for some other parameters relevant to the AP2PFL training such as (i) edge gateway neighbors $(N_i) = \{j\}$, a dual variable $(Z_{(i|j)})$ to facilitate the asynchronous communications and encourage the detection models' variables to be identical among edge gateways, a matrix $(A_{(i|j)})$ contains entries of 1 (i $>$ j) or -1 (j $>$ i) to enforce consistency and equality over the edge between the node or edge gateway (i) and its neighbor (j), a model gradient $\nabla F_i(w_i^t)$ at iteration (t) which is calculated by back-propagation, (mu) which constructs the learning rate $(1/mu)$, as well as penalty coefficient momentum (Γ) and discounting factor (ρ) for providing stable convergence. Also, a loss function (L), the data assigned to each edge gateway $(D_i = (a_i,\ c_i))$, number of epochs (m), number of iterations (t), and batch size (B) are determined. A predefined edge gateway-activation strategy for asynchronous P2P communication is used, where each edge gateway pair $(i,\ j)$ randomly communicates once per approximately every (k) number of updates for each edge gateway.
- **Local model training by edge gateways-** after receiving the initial model parameters, each edge gateway trains a DL-based model using their own private data $(D_i = (a_i,\ c_i))$. Details of the training procedure are provided in the following subsection and Algorithm 2.
- **Updating variables for each edge gateway-** each edge gateway (i) updates its model variables $((w_i^t + 1)$ and $y_{i|j}^{t+1})$ based on the provided parameters and batch size. Suppose that an edge gateway (i) has initial weight values (w_i^k), neighbor(s) (j), and number of neighbors (N_i), it updates its weights based on Eq.1 and the dual variable value based on Eq.2.

$$w_i^{t+1} = (mu\ w_i^m - \nabla F_i(w_i^t) + (\sum_{j \in N(i)} (\tau_{i|j} A_{i|j}^T \widetilde{z}_{i|j}^t + \rho w_j^t)) \oslash (mu + \sum_{j \in N(i)} diag(\tau_{i|j}) + \rho |N(i)|) \tag{1}$$

$$\widetilde{y}_{i|j}^{t+1} = (\widetilde{z}_{i|j}^t + 2A_{i|j} w_i^{t+1}) \tag{2}$$

- **Exchanging and updating variables-** each pair of edge gateways exchanges (using a pull protocol) and updates its variables per around (k) updates for each edge gateway. These variables include weights (w_j^{t+1}) and dual variable $(\widetilde{z}_{i|j}^{t+1} = \widetilde{y}_{j|i}^{t+1})$.

B. PROPOSED DEEP LEARNING (DL)-BASED MODEL

In this section, a newly designed DL-based model for revealing targeted ransomware is described.

Algorithm 1 Asynchronous Peer-to-Peer Federated Learning Procedure

Input: T, i, D_i, $A_{(i|j)}$, N_i, j
Output: Comprehensive DL detection model

Initialization
$z_{i|j} = 0$, mu, Γ,ρ, $A^T_{i|j}$, L, m, B
for $t \leq T$ **do**
  for each i in n **do**
    Compute the t-the iteration model weights w_i^{t+1} *as per algorithm 2* and based on Eq.1 with input parameters N_i, $z_{i|j} = 0$, mu, Γ,ρ, $A^T_{i|j}$, L, m
  end for
  for each i in n **do**
    $update\_count$ += 1
    if ($update\_count \geq$ k) **then**
      $update\_count = 0$
      Select randomly $j \in N(i)$
      Tarnsmit $(w_j^{t+1}, \widetilde{y}_{j|i}^{t+1})$ *from* $j \rightarrow$ *to* i
    end if
  end for
end for

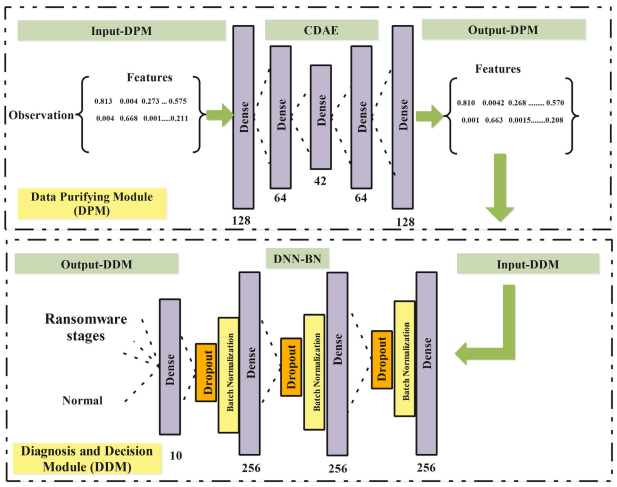


FIGURE 2. The structure of proposed DL-based model.

1) STRUCTURE AND COMPONENTS OF THE MODEL

The designed detection model (illustrated in Figure 2) is composed mainly of Data Purifying Module (DPM) using Contractive Denoising Auto-Encoder (CDAE), and Diagnosis and Decision Module (DDM) constructed by a Deep Neural Network and Batch Normalization (BN), details of which are provided in the following paragraphs.

- **Data Purifying Module (DPM)-** to obtain good capabilities for identifying ransomware attacks, even evasion ones, this module aims to refine and reconstruct a valuable, relevant and robust representation of the input data before passing it to the next module. A hybrid Auto-Encoder algorithm is designed and used as a base for the DPM. We combine Contractive Auto-Encoder (CAE) and Denoising Auto-Encoder (DAE), that is, CDAE, to develop this module. This hybrid approach is advantageous as it improves accuracy and robustness of model. The DAE helps to create a robust reconstruction of the input data stochastically by deliberately corrupting versions of the training data while the CAE to develop an analytically robust feature representation by making some neurons in its network less active [34]. Thus, in CDAE network, an input data observation (a) is corrupted $(\widetilde{a})$ using Gaussian distribution noise with a certain destruction rate during training to hide some of the information from the original data. The CDAE can learn an approximation of identity function $(\widetilde{a} \rightarrow (a))$ and train the hidden layers to extract the robust representation and reconstruct the full input data (a) from the partial information of the original data $(\widetilde{a})$.
We assume that a simple CDAE network architecture consists of one input layer (encoder), one bottleneck layer and one output layer (decoder). Each corrupted input data observation $(\widetilde{a})$ has a feature vector with a dimension (d) that is passed to the bottleneck layer which maps it to an h-dimensional hidden representation (h), where $(h < d)$. Then, the output from the bottleneck layer/hidden layer (h) is used as input to the decoder layer to reconstruct the original data (a) from the corrupted and noisy data $(\widetilde{a})$. The CDAE attempts to make the bottleneck layer's output (h) in a localized space contracted or robust. This is a useful property as it indicates that the mapping is not very sensitive and supports generalizations beyond the training data. The transformation function for the bottleneck layer is calculated using an activation function $(relu)$ to speed up the training process and then (h) is passed to and proceeded by the output layer (g).
The CDAE aims to reduce the distance between (a) and (g), $\underset{\theta}{argmin} L(a, g)$ and obtain a robust representation by finding the optimal variables. $L1$ is the loss function that measures the distance between the original input data (a) and output (g) (reconstructed data) for a batch of data or observations (B) and is calculated using Eq.(3). It consists of the reconstruction error (the first part) for DAE, i.e., the Mean Square Error (MSE) in this paper, and the penalty term which is measured by Frobenius norm of the Jacobean $(||J_h(a)||_F^2)$ of the learned information in the hidden layers to give the effect of contraction or robustness. In Eq.(3), the (λ) is the weight of the penalty term. The Frobenius norm of the Jacobean formula can be calculated using Eq.(4) and is the sum of the squares of all the partial derivatives of the learned information or features through the hidden layer h with respect to the input.

$$L1 = 1/B \sum_{b}^{B} ((a - g)^2 + \lambda ||J_h(a)||_F^2) \quad (3)$$

$$||J_{h\sigma}(\tilde{a})||_F^2 = \sum (\partial h_\sigma(\tilde{a})/\partial \tilde{a}) \quad (4)$$

By increasing the Frobenius norm of the Jacobian, the CDAE can prevent large changes in the hidden layers. In its differential form, the Jacobian matrix reacts to a set's sensitivity caused by small changes in the original space. Therefore, the penalty term ensures that

representation of the learned features is locally invariant and avoids any specific preferential direction [35]. Once it is combined with the reconstruction error, the invariance of the directions can be achieved. In this way, any variation in the data can be captured in the learned representation and other directions contracted. Consequently, the CDAE in the DPM can learn a meaningful, robust and relevant representation of the input data and remove any noise and irrelevant information.

- **Diagnosis and Decision Module (DDM)-** this is a DL-based module that identifies a targeted ransomware attack at an edge gateway and its stages using the output from the previous module (i.e., DPM). A DNN, which is a network with an input layer, two or more hidden ones and an output one, is used as a basis for the DDM. It uses the output from the CDAE as input to train the network by propagating it to the hidden layer(s), and the output from the non-linear transformation of the input data is passed to the output layer to identify the appropriate class. The DNN receives an observation (g) from the DPM and passes it to the hidden layers, the output from which is calculated using the *Relu* (activation function). This is then passed to the output layer (i.e., the soft-max layer in multi-classes) to determine the probability that a particular observation *(g)* belongs to either the normal or specific targeted ransomware stage *(c is a C-dimensional vector)* as in

$$p(\hat{c} = c|g) = e^g / \sum_C e^g \quad (C = 1, 2, 3, \ldots c) \qquad (5)$$

The cross-entropy is used to measure the loss error in the DNN and can be calculated over a batch of observations by Equation.6.

$$L2(\hat{c}, c) = -\sum_B \sum_C c^g . \log(p(\hat{c} = c|g) \qquad (6)$$

where the DNN updates its variables for each batch of observations based on the assumption that the previous layer's output values are within a given distribution. In a non-IID data, updating variables increases the possibility of encountering the problem of vanishing gradients and slows the training process. To handle this issue, BN layers are used [33] to standardize the input for the layers of each batch by obtaining a zero-mean and unit variance distribution of them. Therefore, the BN can handle the sensitivity problem, increase regularization capabilities and provide a stable training process. Furthermore, the dropout layers are used to prevent overfitting.

2) TRAINING OF LOCAL MODEL

Each edge gateway *(i)* trains the proposed DL-based model locally on its own data $(D_i = (a_i, c_i))$ (Algorithm 2). Specifically, each time a new data (i.e., batch) arrives, the edge gateway updates its local model's variables. In the $k - th$ round of updates, the edge gateway *(i)* communicates and exchanges its variables with its neighbors and updates its model weights w_i^{k+1} and dual variables $y_{i|j}^{k+1}$ based on the updated models' given variables.

TABLE 2. Parameters of proposed model.

Federated Learning	($Z_{(i\|j)}$ =0), (mu) =1000, (Γ =3), (ρ =0.5), (t= 657074), epoch (50), (B=500), (k = 6)
DPM-CDAE	Number of hidden layer neurons (128,64,42,64, 128), activation Function (ReLU), loss functions (Mean Square error and Frobenius norm of the Jacobian), noise = 0.05
DDM-DNN-BN	Number of hidden layer neurons (256, 256, 256), activation Function (ReLU), loss function (Cross-Entropy), dropout(0.25), number of BN layers' neurons (256,256,256)

Algorithm 2 Training of Local Model

Input: $D_i, N_i, j, z_{i|j} = 0, mu, \Gamma, \rho, A_{i|j}^T, L1, L2, m$
Output: $w_i^{k+1}, y_{i|j}^{k+1}$

Initialization
Split D_i into batches with equal size B B;
Set the initial model parameters W_i and $yi|j$.
for each b in B **do**
  $\widetilde{a}, c \leftarrow get\_input(b)$
  $g \leftarrow$ *forward* $\widetilde{a}$ *to CDAE*
  $\widetilde{c} \leftarrow$ *forward g to DNN-BN*
  Compute L1 for CDAE
  Compute L2 for DNN-BN
  Compute gradient $\nabla F_i(w_i^t)$ *for L1 and L2*
  Update the model's parameters based on eq.1 and eq.2
end for
return $w_i^{k+1}, y_{i|j}^{k+1}$

V. PERFORMANCE EVALUATION

A. EXPERIMENT SETTINGS

1) ENVIRONMENTAL SETUP AND PARAMETERS

The proposed model was simulated and implemented using the Pytorch framework on an Ubuntu platform (NVIDIA Tesla K80 Accelerator, 4 GPU), with all its edge gateways run in one node. The parameters used for FL, the DPM based on the CDAE and the DDM based on the DNN-BN are described in Table 2.

2) DATA RESOURCES AND DESCRIPTION

We use three datasets to evaluate our proposed model, namely, X-IIoTID [36], ISoT [37] and NSL-KDD dataset [38]. X-IIoTID is connectivity- and device-agnostic features collected and generated from multiple data sources, including network traffic, system and application logs, system resources and commercial IDSs. It includes the normal behaviors of brownfield IIoT systems and multi-stage targeted ransomware attacks against edge gateways, with its final version having 421,417 normal observations and 399,417 attack ones. The ISoT dataset, a commonly used and publicly available ransomware one, was also used in the experiments. It includes data related to the most popular ransomware families and Windows users' software applications. Although it does not represent the realistic behaviors of

TABLE 3. Results for IID-binary class (X-IIoTID dataset) with various numbers of connected edge gateways.

Metric	Proposed Model				AP2PFL-DNN				AP2PFL-MLP			
	2	3	4	5	2	3	4	5	2	3	4	5
Accuracy	98.23	98.22	98.16	**98.13**	98.17	98.34	98.54	**97.95**	95.98	97.50	97.57	**97.21**
Recall	98.22	98.20	98.15	**98.10**	98.14	98.31	98.51	**97.90**	95.91	97.45	97.53	**97.16**
Precision	98.25	98.25	98.16	**98.20**	98.21	98.39	98.58	**98.03**	96.14	97.61	97.65	**97.3**
F1-Score	98.24	98.23	98.16	**98.15**	98.18	98.35	98.55	**97.97**	96.03	97.53	97.59	**97.23**

TABLE 4. Results for IID-binary class (ISoT dataset) with various numbers of connected edge gateways.

Metric	Proposed Model				AP2PFL-DNN				AP2PFL-MLP			
	2	3	4	5	2	3	4	5	2	3	4	5
Accuracy	98.23	98.33	98.26	**98.33**	98.17	98.58	98.64	**98.15**	87.49	87.1	89.55	**90.09**
Recall	97.88	98.13	98.01	**97.78**	97.89	98.49	98.49	**98.18**	90.55	89.21	92.51	**91.94**
Precision	97.57	97.61	97.54	**97.93**	97.44	97.9	98.03	**97.15**	83.67	83.07	85.76	**85.24**
F1-Score	97.73	97.87	97.77	**97.86**	97.67	98.19	98.26	**97.66**	86.97	86.03	89.01	**88.46**

IIoT systems and the potential targeted ransomware stages in them, it shows a network's lateral movements and the C&C stages of popular ransomware-affected OT such as Petya and Wannacry. The final ISoT dataset included 7392 normal and 20,508 attack observations (i.e., normal and ransomware classes).

The final dataset that was used in the experiments is the NSL-KDD dataset. Although it is obsolete and does not represent the new generation of traffic and system behavior in IIoT systems, it is the most common benchmark dataset in the intrusion detection field, and researchers extensively use it to date. Therefore, it can be used to evaluate the proposed model compared with other existing models. NSL-KDD has features collected from network and host and includes various attacks representing the early stages of targeted attacks. It consists of 77,054 normal and 71460 attack observations, including probing, DoS, User to Root (U2R), and Remote to Local (R2L) attacks. These attacks can be classified under reconnaissance, DoS, weaponization, and exploitation, respectively. These datasets were normalized using a min-max scaler to restrict the data within a specific range [0, 1] while maintaining the original data's distribution. The contents or observations of each dataset were split into 80% for training and 20% for testing.

3) BASELINE STUDIES

The performance of the proposed model was compared with those of two state-of-the-art ones, the MLP and DNN algorithms used by Schneble and Thamilarasu [29] and Rey *et al.* [24], respectively, as FL models to detect malware in edge devices. They were fully reproduced (using the same parameters) and used with the AP2PFL (i.e., AP2PFL-DNN and AP2PFL-MLP). They were also run in a centralized mode (where all the data was placed in one server) to evaluate their performances for detecting targeted ransomware attacks in brownfield IIoT systems (i.e., Cent-MLP, Cent-DNN, and Cent-proposed DL-based model). To evaluate the performance of the detection models, the common metrics used were the accuracy, recall or detection rate, precision and F1-score [39], [40].

B. EXPERIMENTAL RESULTS AND DISCUSSION

1) MODELS' PERFORMANCE IN IID DATA

The performance of the proposed model was first compared with those in the baseline studies mentioned above on the XIIoT-ID, ISoT and NSL-KDD datasets in an IID data.

a: IID DATA-BINARY CLASS

As previously discussed, in IID data, normal and ransomware observations were divided evenly between connected edge gateways (i.e., the binary class). Four groups of experiments were conducted using different numbers of these gateways (i.e., n = 2, 3, 4, 5) for each dataset. As can be seen in Tables 3, 4 and 10, the proposed model achieved good performances overall and, when five gateways (i.e., n = 5) were connected, outperformed the other baseline models trained on the X-IIoTID, ISoT, and NSL-KDD datasets. It achieved the highest values of the accuracy, recall, precision and F1-score for the X-IIoTID dataset of 98.13%, 98.10%, 98.20% and 98.15%, respectively, and those of 98.33%, 97.78%, 97.93% and 97.86% for the ISOT one (as presented in Table 4). It also obtained of the accuracy, precision, recall and F-score of 97.03% for NSL-KDD dataset.

Figures 3, 4 and 5 present the values of the loss, accuracy and F1-score of all the centralized (i.e., Cent-DNN, Cent-MLP, Cent-proposed DL-based model) and federated models (i.e., AP2PFL-DNN, AP2PFL-MLP, and proposed model) with different numbers of epochs (when $n = 5$) for the X-IIoTID, ISoT, and NSL-KDD datasets. It is clear that they all tended to converge after a sufficient number of epochs which meant that there were sufficient numbers of exchanges and updates of parameters. Importantly, the proposed model and cent-proposed DL-based model were approximately equivalent in their performance and generally performed better than the others, particularly for the NSL-KDD dataset.

TABLE 5. Results for IID-binary class (NSL-KDD dataset) with various numbers of connected edge gateways.

Metric	Proposed Model				AP2PFL-DNN				AP2PFL-MLP			
	2	3	4	5	2	3	4	5	2	3	4	5
Accuracy	97.18	97.13	97.06	**97.03**	96.83	96.73	96.77	**96.34**	94.56	94.95	95.52	**94.90**
Recall	97.18	97.13	97.06	**97.03**	96.83	96.73	96.77	**96.34**	94.56	94.95	95.52	**94.91**
Precision	97.19	97.13	97.07	**97.03**	96.84	96.73	96.77	**96.34**	94.59	94.98	95.56	**94.98**
F1-Score	97.18	97.13	97.00	**97.03**	96.83	96.73	96.77	**96.34**	94.56	94.95	95.52	**94.90**

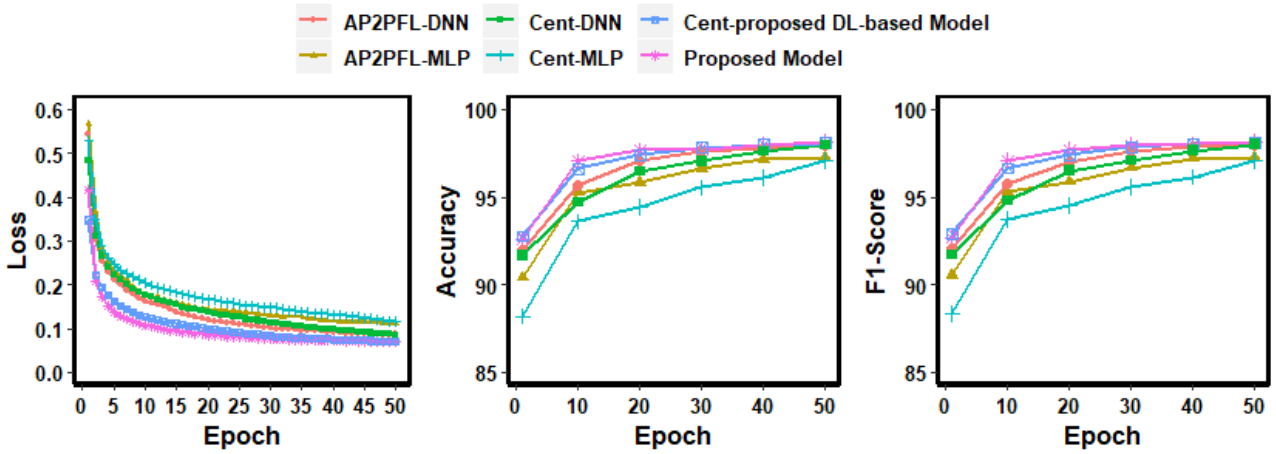


FIGURE 3. Comparison of loss, accuracy and F1-score values of detection models with various numbers of epochs for X-IIoTID dataset where n = 5.

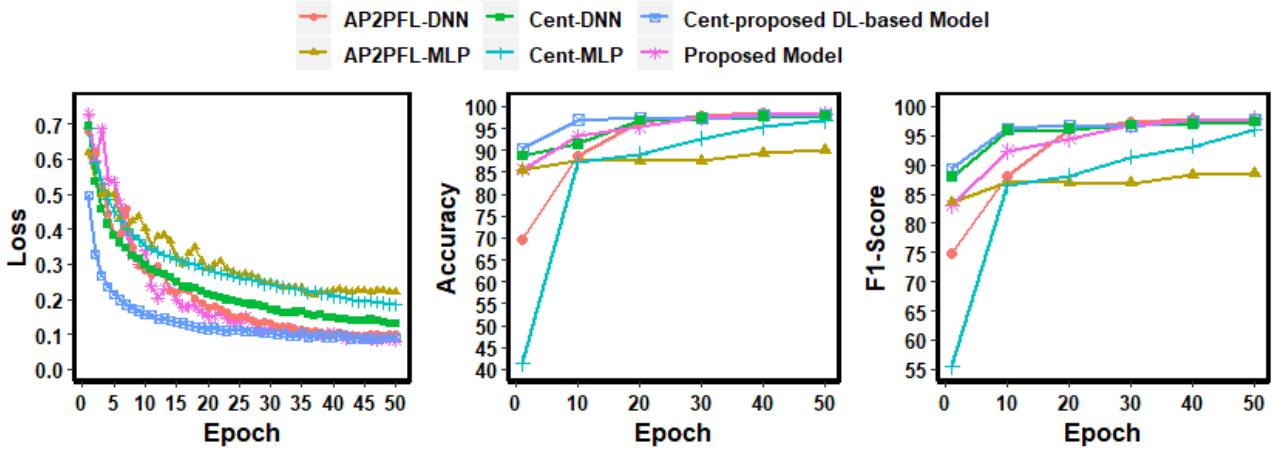


FIGURE 4. Comparison of loss, accuracy and F1-score values of detection models with various numbers of epochs for ISoT dataset where n = 5.

b: IID DATA-MULTIPLE CLASSES

In the multiple classes of IID data, observations in the normal and ransomware stages (multiple attack classes) were divided evenly between connected edge gateways. Four groups of experiments were conducted with different numbers of edge gateways (i.e., n = 2, 3, 4, 5) used for the X-IIoTID dataset (with one normal class and nine attacks) and NSL-KDD dataset (one normal and 4 attacks). The ISoT dataset was not used in these experiments as it contains two classes only (normal and attack). As can be seen in Tables 6 and 7, the proposed model performed better than the others for the four different groups (i.e., n = 2, 3, 4, 5) and for X-IIoTID and NSL-KDD datasets. It is also worth noting that it obtained accuracy, precision, recall and F1-score values of 97.21%, 97.78%, 97.21%, and 97.41%, respectively, for the X-IIoTID dataset, and those of 93.10%, 96.10%, 93.10%, and 94.30% for the NSL-KDD dataset (as presented in Table 7) with n = 5 connected edge gateways. It is clear that the three tested models obtained less performance in IID data-multiple classes than IID data-binary class for NSL-KDD datasets. This is because NSL-KDD has very minor classes that are not sufficient to train multiple models.

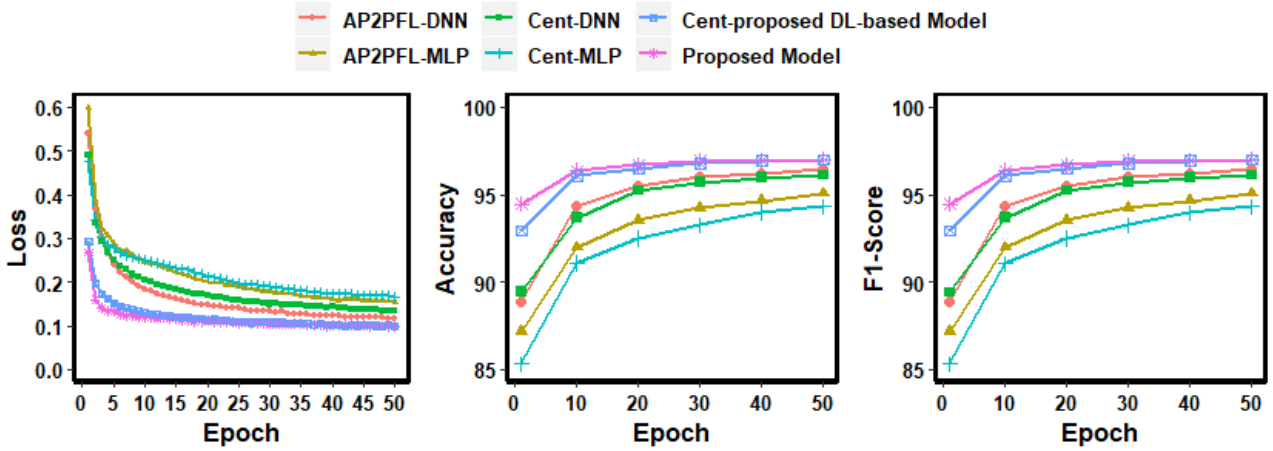


FIGURE 5. Comparison of loss, accuracy and F1-score values of detection models with various numbers of epochs for NSL-KDD dataset where n = 5.

TABLE 6. Results for detection rates of attacks in IID data-multiple classes (X-IIoTID dataset).

Metric	Proposed Model				AP2PFL-DNN				AP2PFL-MLP			
	2	3	4	5	2	3	4	5	2	3	4	5
Accuracy	96.38	96.79	96.87	**97.21**	94.65	92.82	95.61	**96.00**	91.95	92.06	91.89	**91.13**
Recall	96.38	96.79	96.87	**97.21**	94.65	92.82	95.61	**96.00**	91.95	92.06	91.89	**91.13**
Precision	97.27	97.61	97.70	**97.78**	96.27	95.39	96.99	**97.13**	94.57	94.65	94.78	**93.99**
F1-Score	96.38	96.79	97.19	**97.41**	95.23	93.73	96.14	**96.42**	92.83	92.93	92.87	**92.061**

TABLE 7. Results for detection rates of attacks in IID data-multiple classes (NSL-KDD dataset).

Metric	Proposed Model				AP2PFL-DNN				AP2PFL-MLP			
	2	3	4	5	2	3	4	5	2	3	4	5
Accuracy	93.24	92.70	92.24	**93.10**	91.46	90.87	91.90	**91.29**	89.16	87.59	87.43	**87.94**
Recall	93.24	92.70	92.24	**93.10**	91.46	90.87	91.90	**91.29**	89.16	87.59	87.43	**87.94**
Precision	96.03	96.01	95.85	**96.10**	95.60	95.63	95.99	**95.60**	93.45	93.37	93.22	**93.41**
F1-Score	94.30	93.99	93.57	**94.30**	93.06	92.69	93.58	**92.97**	90.63	89.45	89.33	**89.93**

However, the proposed model was able to maintain its good performance.

Figures 6 and 7 illustrate the loss, accuracy and F1-score values of all the centralized and federated models with various numbers of epochs (when n = 5) for the X-IIoTID and NSL-KDD datasets (multiple classes). As shown, the proposed model and cent-proposed DL-based model performed best. Mainly, Figure 7 shows a clear difference in loss, accuracy and F1-score between the proposed model and others for the NSL-KDD dataset.

Tables 8 and 9 show the performance of models in identifying targeted ransomware stages in X-IIoTID and NSL-KDD datasets. Overall, the proposed model had the best performance. It achieved values of 94.15%, 99.94%, 95.69%, 99.10%, 98.91%, 99.98%, 99.81%, 99.95% and 100% for detecting reconnaissance, weaponization, exploitation, lateral movement, C& C, exfiltration, tampering, RDoS and crypto-ransomware, respectively, in the X-IIoTID dataset. The cent-DNN had the best performance for detecting C&C and tampering. It achieved values of 99.64% and 99.91 respectively. As presented in Table 9, the cent-proposed DL-based model achieved the best performance for detecting weaponization and exploitation in the NSL-KDD dataset, whereby it obtained values of 98.69% and 94.17% respectively. The proposed model achieved a value of 89.57% for detecting reconnaissance, and the AP2PFL-DNN obtained the highest detection rate (i.e., 83.72%) for Denial of Service (DoS) in the NSL-KDD dataset.

Discussion results of IID data: The capabilities of the proposed model to handle targeted ransomware attacks were tested using the X-IIoTID, ISoT and NSL-KDD datasets. The proposed model proved its good capability for dealing with homogeneous data (i.e., IID-binary class and IID-multiple classes) distributed evenly among connected edge gateways, each of which had the same number of classes. In IID data-binary class and using different numbers of connected edge gateways (i.e., 2, 3, 4, 5), the proposed model performed well and was better in terms of the accuracy, recall, precision and F1-score values than the other baseline models (i.e., AP2PFL-DNN and AP2PFL-MLP) for

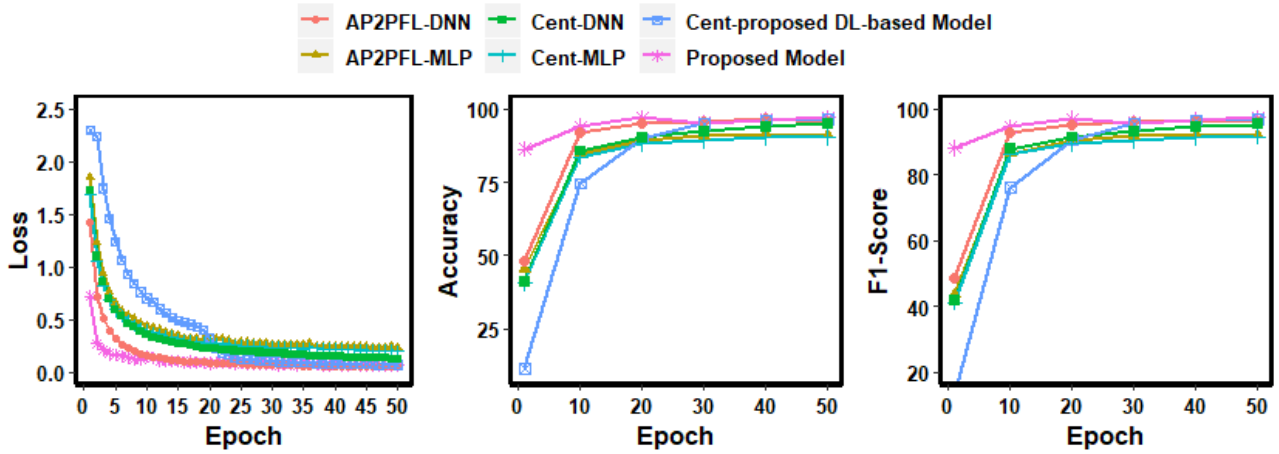


FIGURE 6. Comparison of loss, accuracy and F1-score values of detection models with various numbers of epochs for X-IIoTID dataset (multiple classes) where n = 5.

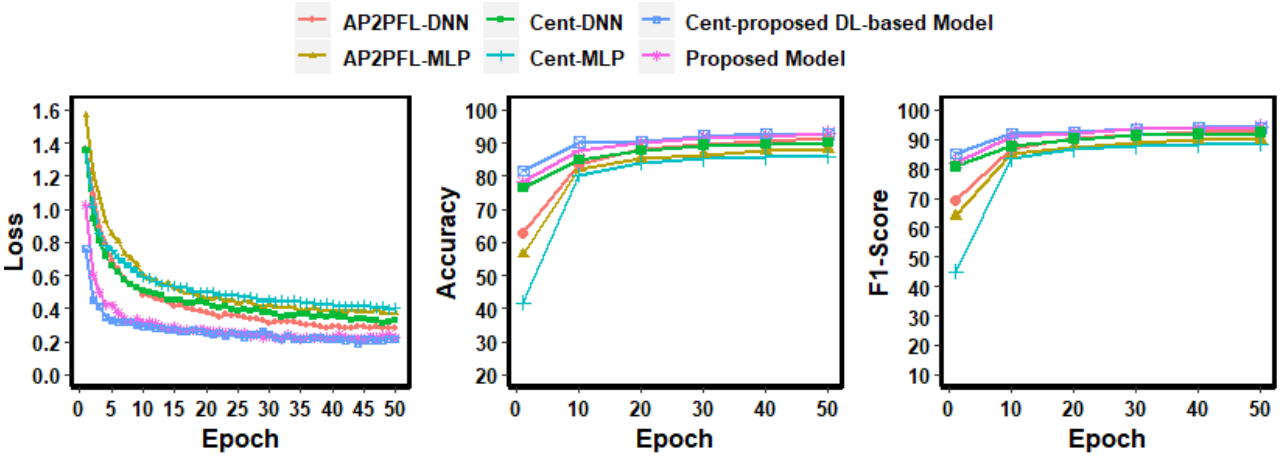


FIGURE 7. Comparison of loss, accuracy and F1-score values of detection models with various numbers of epochs for NSL-KDD dataset (multiple classes) where n = 5.

TABLE 8. Detection rate's results for targeted ransomware stages in IID data-multiple classes (X-IIoTID dataset).

Attack	Cent-MLP	Cent-DNN	Cent-Proposed DL-based Model	AP2PFL-MLP	AP2PFL-DNN	proposed model
Reconnaissance	88.68	92.20	92.70	88.75	93.74	**94.15**
Weaponization	99.79	99.67	99.65	99.73	99.76	**99.94**
Exploitation	91.81	95.69	93.53	87.07	95.26	**95.69**
Lateral Movement	95.77	98.56	98.89	94.11	98.80	**99.10**
C&C	99.45	**99.64**	99.45	98.36	99.45	98.91
Exfiltration	99.95	99.95	99.93	99.95	99.98	**99.98**
Tampering	99.53	**99.91**	99.81	99.34	99.81	99.81
RDoS	99.78	99.86	99.89	99.78	99.86	**99.95**
Crypto-ransomware	100	100	100	100	100	**100**

five connected edge gateways and three tested datasets. The same conclusion can be obtained for IID data-multiple classes. The proposed model achieved the best performance compared with the baselines models for X-IIoTID and NSL-KDD datasets. Furthermore, these results clearly indicate that the performance of the proposed model (i.e., AP2PFL with DL) was as good as that of the cent-proposed DL-based model. This was remarkable because the centralized model was easier to optimize than the federated one, but both models obtained approximately the same performance.

TABLE 9. Detection rate's results for targeted ransomware stages in IID data-multiple classes (NSL-KDD dataset).

Attack	Cent-MLP	Cent-DNN	Cent-Proposed DL-based Model	AP2PFL-MLP	AP2PFL-DNN	proposed model
Reconnaissance	78.18	84.48	88.00	81.33	86.52	**89.57**
Weaponization	74.42	76.74	**81.40**	81.40	83.72	79.07
Exploitation	90.52	91.48	**94.17**	89.65	91.47	91.98
Denial of Service (DoS)	95.17	97.29	98.69	95.17	**97.83**	98.23

TABLE 10. Results for non-IID data (X-IIoTID dataset) with various numbers of connected edge gateways.

Metric	Proposed Model				AP2PFL-DNN				AP2PFL-MLP			
	2	3	4	5	2	3	4	5	2	3	4	5
Accuracy	95.14	95.74	96.14	**96.42**	94.70	95.20	95.21	**95.82**	92.96	92.39	93.36	**92.85**
Recall	95.14	95.74	96.14	**96.42**	94.70	95.20	95.21	**95.82**	92.96	92.39	93.36	**92.85**
Precision	96.43	96.67	97.00	**97.20**	97.02	95.51	95.38	**96.05**	93.41	93.48	94.55	**94.08**
F1-Score	95.61	96.06	96.25	**96.72**	95.43	95.27	95.01	**95.87**	93.10	92.67	93.73	**93.24**

TABLE 11. Results for non-IID data (NSL-KDD dataset) with various numbers of connected edge gateways.

Metric	Proposed Model				AP2PFL-DNN				AP2PFL-MLP			
	2	3	4	5	2	3	4	5	2	3	4	5
Accuracy	91.19	92.23	93.39	**94.39**	88.79	91.70	89.72	**88.72**	86.91	87.74	88.58	**89.13**
Recall	91.19	92.23	93.39	**94.39**	88.79	91.70	89.72	**88.72**	86.91	87.74	88.58	**89.13**
Precision	93.96	95.35	95.60	**95.02**	92.85	92.08	92.72	**93.34**	91.49	90.25	92.36	**89.84**
F1-Score	92.17	93.39	94.30	**94.59**	89.93	91.78	91.00	**90.77**	88.02	88.35	89.60	**89.13**

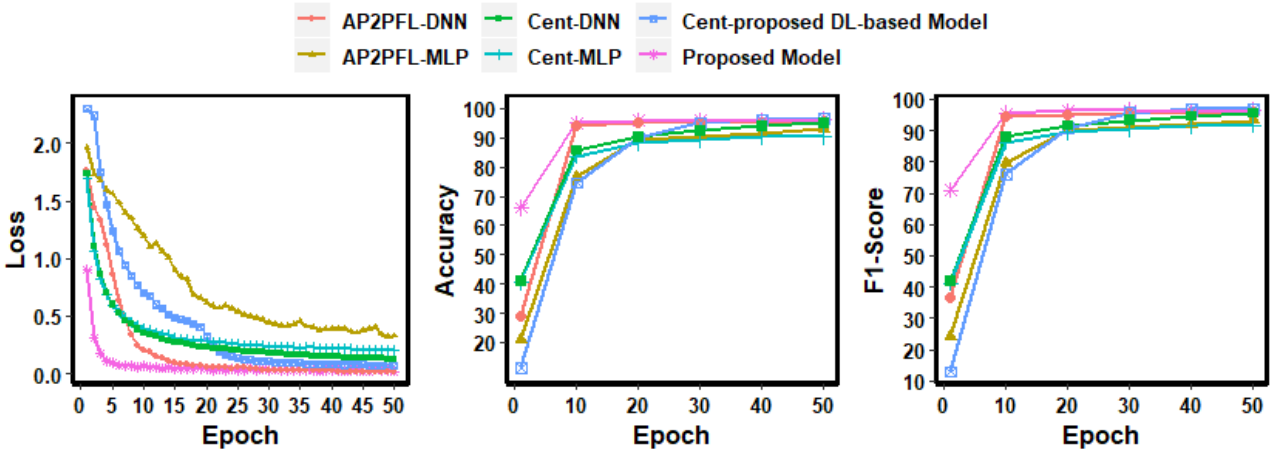


FIGURE 8. Comparison of values of loss, accuracy and F1-score of detection models with various epochs for non-IID data (X-IIoTID dataset).

2) MODELS' PERFORMANCES IN NON-IID DATA

Tables 10 and 11 show the results of the four metrics for the tested models with different numbers of connected edge gateways (i.e., n = 2, 3, 4, 5) in the non-IID data and for X-IIoTID and NSL-KDD datasets. As can be seen, the proposed model achieved better results than the others in terms of the accuracy, recall, precision and F1-score values. It performed best with five connected edge gateways (i.e., n = 5), obtaining accuracy, precision, recall and F1-score values of 96.42%, 97.20%, 96.42% and 96.72%, respectively for the X-IIoTID dataset. It also achieved accuracy, precision, recall and F1-score values of 94.39%, 95.02%, 94.39%, and 94.59%, respectively, for the NSL-KDD dataset. Also, the loss, accuracy and F1-score values for all the centralized and federated models with different numbers of epochs and five connected edge gateways for X-IIoTID and NSL-KDD datasets are shown in Figures 8 and 9. It can be noted that the proposed model performed satisfactorily, better than the others for both datasets. As the number of epochs increased from 1 to 50, its performance generally improved and was better when the number of epochs was sufficiently large.

The detection rates for the different stages of a targeted ransomware attack are shown in Tables 12 and 13. As can be seen, the centralized models achieved slightly better performance than the federated ones for the X-IIoTID dataset.

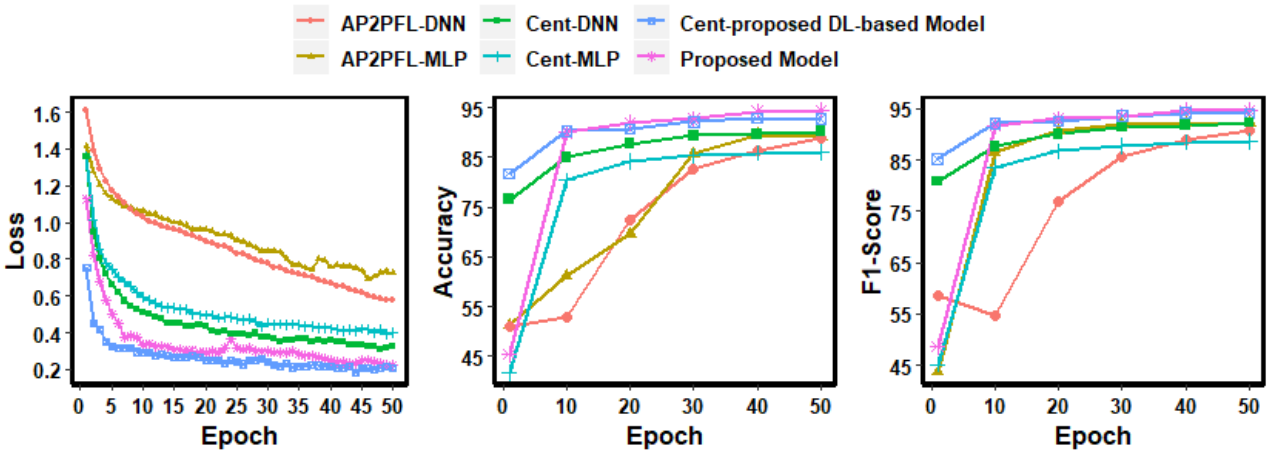


FIGURE 9. Comparison of values of loss, accuracy and F1-score of detection models with various epochs for non-IID data (NSL-KDD dataset).

TABLE 12. Detection rate's results for targeted ransomware stages based on Non-IID (X-IIoTID).

Attack	Cent-MLP	Cent-DNN	Cent-Proposed DL-based Model	AP2PFL-MLP	AP2PFL-DNN	proposed model
Reconnaissance	88.68	92.20	92.70	86.74	**93.96**	93.00
Weaponisation	**99.79**	99.67	99.65	99.30	98.89	99.76
Exploitation	91.81	**95.69**	93.53	70.26	62.93	64.66
Lateral Movement	95.77	98.56	**98.89**	92.58	86.18	91.48
C&C	99.45	**99.64**	99.45	94.73	71.09	99.27
Exfiltration	99.95	99.95	99.93	99.98	98.28	**99.98**
Tampering	99.53	**99.91**	99.81	96.62	97.46	99.72
RDoS	99.78	99.86	99.89	99.72	98.47	**99.94**
Crypto-ransomware	100	100	100	94.44	46.67	**100**

TABLE 13. Detection rate's results for targeted ransomware stages based on Non-IID (NSL-KDD dataset).

Attack	Cent-MLP	Cent-DNN	Cent-Proposed DL-based Model	AP2PFL-MLP	AP2PFL-DNN	proposed model
Reconnaissance	78.18	84.48	88.00	87.80	85.72	**96.02**
Weaponisation	74.42	76.74	81.40	25.66	**97.67**	55.81
Exploitation	90.52	91.48	**94.17**	41.86	54.01	86.73
Denial of Service (DoS)	95.17	97.29	**98.69**	96.86	93.32	87.58

The cent-proposed DL-based model achieved the best value of 98.89% for detecting lateral movement, and the Cent-MLP obtained 99.79% for detecting weaponization. Also, the Cent-DNN was better for detecting exploitation, C&C, and tampering, obtaining 95.69%, 99.64%, and 99.91%, respectively. However, the proposed model achieved the best values of 99.98%, 99.94%, and 100% for detecting exfiltration, RDoS, and crypto-ransomware, and AP2PFL-DNN was better for identifying reconnaissance. Overall, the proposed model performed satisfactorily compared with the others and would improve by increasing the training data size for minor classes. For the NSL-KDD dataset, the centralized models performed best. This is because the NSL-KDD dataset has minor classes distributed among connected edge gateways and are not sufficient for training. However, as can be seen, the cent-proposed DL-based model was better than other models, achieving 94.17%, and 98.69% for detecting exploitation and DoS.

Discussion results of Non-IID data: The proposed model demonstrated a significant performance for detecting targeted ransomware attacks against IIoT edge gateways using the heterogeneous data (i.e., non-IID data). The X-IIoTID and NSL-KDD datasets were used to evaluate the model's performances for identifying the various activities and stages of targeted ransomware attacks. However, the proposed model using non-IID data achieved better accuracy, recall and F1-score than IID data for the NSL-KDD dataset. This was mainly because each edge gateway was trained using the full data of minor classes, and the class's data was not divided between edge gateways. This allowed these edge

TABLE 14. Comparison between different models under evasion attacks (X-IIoTID dataset).

Model	Non-evasion		FGSM		BF	
	IID	Non-IID	IID	Non-IID	IID	Non-IID
AP2PFL-MLP	95.77	94.80	93.23	91.11	93.83	92.58
AP2PFL-DNN	97.80	95.76	93.68	85.54	96.07	89.54
Proposed model	**98.02**	**96.92**	**97.00**	**94.84**	**97.61**	**96.44**

TABLE 15. Comparison between different models under evasion attacks (NSL-KDD dataset).

Model	Non-evasion		FGSM		BF	
	IID	Non-IID	IID	Non-IID	IID	Non-IID
AP2PFL-MLP	83.99	85.54	81.33	81.26	80.55	76.40
AP2PFL-DNN	88.58	85.05	82.40	78.82	84.86	81.13
Proposed model	**91.05**	**94.08**	**87.26**	**92.26**	**89.23**	**93.05**

gateway to be well trained. The proposed model performed much better overall than the others, noting the role of the DPM in improving its generalization capabilities and refining and reconstructing a robust representation of the input data. This was also because of the well-structured and developed DDM using a DNN and BN. The proposed model performed approximately the same as cent-proposed DL-based model in terms of accuracy, loss, and F1-score. This significant achievement proved that the proposed model would be eminently usable in real brownfield IIoT systems in which distributed edge gateways with heterogeneous data (i.e., non-IID) work collaboratively and efficiently in an asynchronous P2P manner.

3) MODELS' ROBUSTNESS AGAINST EVASION ATTACKS

Table 14 and 15 show the detection rate for targeted ransomware evasion attacks (i.e., unknown targeted ransomware ones) with epsilon = 5% in the IID (multiple classes) and non-IID data for the X-IIoTID and NSL-KDD datasets. All the targeted ransomware observations were modified to be similar to normal or legitimate ones using FGSM and BF attack techniques (i.e., white- and black-box techniques, respectively). As can be seen, all the models performed well for detecting targeted ransomware attacks in the X-IIoTID dataset when there were non-evasion ones. However, their performances significantly decreased under FGSM attacks, particularly those in the non-IID data. Nevertheless, overall, the proposed model was more robust than the others against evasion attacks in both the IID and non-IID data for the X-IIoTID dataset. In the IID data, it achieved values of 98.02 %, 97.00% and 97.61% for non-evasion and FGSM and BF attacks, respectively, for the AP2PFL-DNN, 97.80%, 93.68% and 96.07%, respectively, and for the AP2PFL-MLP 95.77%, 93.23% and 93.83%, respectively. As can be noted, the attack detection rate of the proposed model was approximately 1.02% and 0.41% less than the non-evasion one, 4.12% and 1.73% for the AP2PFL-DNN one and 2.54% and 1.94% for the AP2PFL-MLP one. In the non-IID data, the proposed model achieved values of 96.92%, 94.84% and 96.44%, results that were 2.08% and 0.48% less than those of the non-evasion one. The AP2PFL-DNN and AP2PFL-MLP models' performances decreased by approximately 10.22% and 6.22%, and 3.69% and 2.22%, respectively. These results proved that the proposed model was more robust than the others against targeted ransomware evasion attacks (i.e., FGSM and BF attacks). The same conclusion could be reached for the proposed model for the NSL-KDD dataset (as shown in Table 15) as its performances decreased by approximately 3.79% and 1.82% in the IID data and 1.82% and 1.03% in the non-IID one under FGSM and BF attacks, respectively.

The robustness of the proposed model against white-box FGSM and black-box BF evasion attacks was another strong point in its favour which strongly reinforced the proposed model's superiority and suitability for real-world brownfield IIoT deployments. Most importantly, it is clear that the proposed model performed better than the others and achieved a balance between targeted ransomware detection and robustness against evasion attacks. Therefore, it is worth noting that it would be a suitable and efficient solution for protecting the edge gateways of brownfield IIoT systems against targeted ransomware attacks because of its superior performance on homogeneous and heterogeneous datasets and its robustness against evasion attacks.

4) TIME COMPLEXITY OF PROPOSED MODEL

The proposed model's processing times, which were collected during the experiments to analyze the amount each edge gateway required in each epoch, are shown in Figures 10 and 11. These times include those for training, transfers and exchanges (communications with neighbors) and updates. Figure 10 shows that the average processing times (seconds) per epoch and edge gateway in the IID and non-IID data for IIX-IIoTID dataset varied, increasing with increasing numbers of connected edge gateways (i.e., n = 2, 3, 4, 5) and ranging between 30 and 36. It is worth noting that each edge gateway performed approximately 220 communication rounds in each epoch which meant it exchanged its variables with its neighbors, on average, approximately 220 times in each epoch. As the total number of epochs in the experiments was 50, each edge gateway had, on average, approximately 11,000 communication rounds during the entire training process. Figure 11 shows average processing times (seconds) per epoch and edge gateway in the IID and non-IID data for the NSL-KDD dataset. The proposed model took less processing time in this dataset, ranging between 7-9 seconds. Each edge gateway exchanged

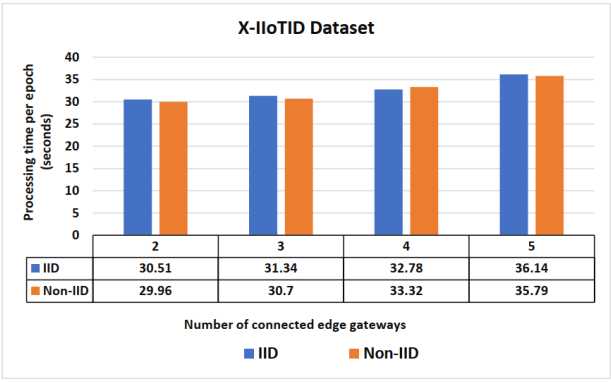


FIGURE 10. Average processing time per epoch/edge gateway with a varying number of connected edge gateways for X-IIoTID dataset.

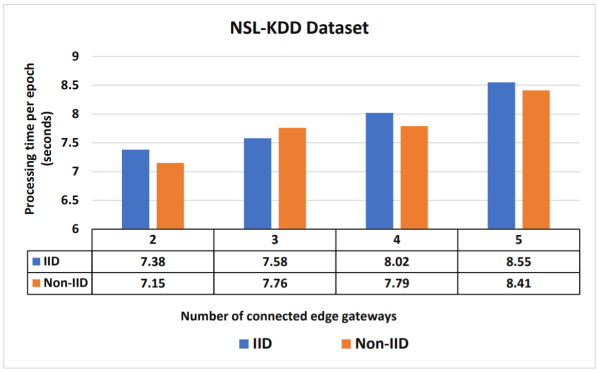


FIGURE 11. Average processing time per epoch/edge gateway with a varying number of connected edge gateways for NSL-KDD dataset.

its variables with its neighbors approximately 49 times in each epoch and 2450 communication rounds during the entire training process. NSL-KDD has less processing time and communication rounds since it has fewer observations, classes, and features than the X-IIoTID dataset. Given that the proposed model consisted of two modules built based on DL algorithms and dealt with IID and non-IID data and the communication among neighbours was asynchronous, its average processing time was reasonable and totally acceptable for both datasets and edge gateways in the real environment.

VI. COMPARATIVE STUDY WITH OTHER FL MODELS

To illustrate the effectiveness of our proposed model, the proposed model's performance is compared with those of three recently developed FL-based intrusion detection models tested on the NSL-KDD dataset, namely, the Multi Criteria Client Selection in FL (FedMCCS) [41], Hierarchical FL(HFL) [16], and Probabilistic Hybrid Ensemble Classifier (PHEC) [42]. Table 16 demonstrates the result achieved by the proposed model for both IID (binary- and multiple-classes) and non-IID data compared with other models. It is clear that the proposed model obtained the best accuracy for homogeneous (i.e., IID) and heterogeneous

TABLE 16. Comparison with other FL-based intrusion detection models.

Model	Accuracy (%)
FedMCCS [41]	81.00
Fed-PHEC [42]	88.42
HFL [16]	77.50
Proposed Model (IID-binary)	97.03
Proposed Model (IID-multiple classes)	93.10
Proposed Model (Non-IID)	94.39

(i.e., Non-IID) data with 97.03%, 93.10% and 94.39%, respectively. The three models achieved a good performance in terms of accuracy based on the client-server FL approach. FedMCCS adopted a client-server FL approach with DNN. Clients or edge gateways participate in the global model's update based on their resources (CPU, memory, and energy) and ability to successfully train and send the needed updates. Therefore, FedMCCS achieved 81% and maximized the number of participated clients/edge gateways while reducing the number of communications with the cloud server. Nevertheless, this model still needs to optimize the client selection approach to improve the efficiency of the intrusion detection model. HFL model used DNN based on client-server FedAvg but with edge layer between IoT devices (clients) and cloud. It relied on trained two global models at the two edge gateways, whereby each edge gateway has its own clients. The edge gateways acted as clients for the cloud server to build another higher-level global model. This Hierarchical approach has many problems related to delay caused by each layer and the energy constraints of clients (i.e., IoT devices). It also obtained less accuracy than other models (i.e., 77.50%).

The Fed-PHEC model used many MLP networks that share their parameters with a central server where they are aggregated to construct a global model. For each data sample, the global model (or aggregated mode) yields a set of probabilities, where each probability represents the probability of a specific local training model. This model is the best out of the three models, whereby it achieved a value of 88.42% for accuracy. However, further analysis and experiments are needed to improve the performance.

However, our proposed model performs better than the above models as it relies on AP2PFL and DL techniques. It used CDAE to refine and reconstruct the input data, improving the generalization and robustness of models against noise data. Therefore, it can learn a good and relevant representation for input data in an unsupervised manner before passing it to the DDM. Also, it used DNN with BN to facilitate the detection process, whereby the constructed network architecture can identify the attack and normal behavior patterns and classify them efficiently to the appropriate class. The proposed model is different from the previous intrusion detection models as it depends on AP2PFL, whereby each client connects only with its neighbors in a P2P manner. This eliminates the need to have a critical selection method to choose the best participants in updating process and the

need to have a global model. It also guarantees that all connected clients participate in the updating process in direct and indirect ways. Each client has only one model, which is updated based on its neighbors' models. It also relied on an asynchronous algorithm, whereby each client does not have to wait for all its neighbors to complete aggregating models. Given that the devices of IIoT edge gateways are designed to operate time-sensitive processes, provide fewer communications with cloud servers and reduce bandwidth and network latency, the proposed model is more suitable than existing client-server ones for edge gateways in brownfield IIoT systems. It proved a superior performance on homogeneous and heterogeneous datasets and maintained its robustness against evasion attacks. Therefore, these traits ensure that our proposed model is appropriate for deployment in a real IIoT and protecting edge systems against targeted ransomware.

Nevertheless, the proposed detection model has many limitations. For example, the DL-based model faces significant challenges in choosing the best network structures and architectures to guarantee stable and good detection accuracy. This is not a simple task as it requires numerous empirical experiments. This limitation can be exposed using optimization techniques such as Particle swarm optimization [43]. Another limitation is that DL-based models deal with only numerical data, and this issue is solved using pre-processing in the proposed model. Furthermore, although the proposed model uses asynchronous and P2P communication, is fully decentralized and does not have a central server, it still faces security and privacy challenges, such as poisoning training attacks. This limitation could be mitigated to some extent by the developed DPM, which, as stated previously, refines and reconstructs the representation of the input data, which improves the model's robustness. Also, the proposed detection model uses a novel FL algorithm, that is, a PDDM-SGD [33]. Although the model designed based on this algorithm displayed good performances for detecting targeted ransomware and dealing with both homogeneous (i.e., IID) and heterogeneous (i.e., non-IID) data, in its current form, this algorithm has limitations in terms of its stability and performance, and more research is required to improve it.

VII. CONCLUSION AND FUTURE WORK

This paper proposed the first-of-its-kind targeted ransomware detection model tailored for IIoT edge gateways. It used an Asynchronous Peer-to-Peer Federated Learning (AP2PFL) and Deep Learning (DL) as a targeted ransomware detection algorithm and included IID and non-IID learning models. The detection model (i.e., DL-based model) consists of two modules: DPM and DDM modules. The DPM refines and reconstructs a valuable and robust representation for the input data before passing it to the DDM module to identify targeted ransomware attacks. These modules in each edge gateway work collaboratively with its neighbors and share their knowledge about targeted ransomware stages using AP2PFL. A comprehensive set of experiments proved that the proposed model outperforms the baseline models in terms of performance metrics. It has high capabilities to deal with homogeneous (i.e., IID) and heterogeneous (i.e., Non-IID) data and protect brownfield IIoT systems' edge gateways. The performance of the proposed model under evasion attacks (i.e., unknown targeted ransomware) was also evaluated and tested, demonstrating significant robustness against these evasion attacks.

In future works, we will test the proposed model's performance in a real IIoT environment. We also will explore the privacy and security of federated and deep learning techniques. In particular, we plan to test the proposed model against poison training attacks and consider other robust methods to improve its security and privacy, such as the null-class technique and anomaly detectors. Furthermore, we plan to implement different P2P network architecture and improve the performance of the proposed model for the non-IID data.

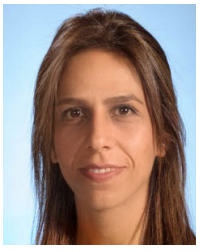
REFERENCES

[1] T. Qiu, J. Chi, X. Zhou, Z. Ning, M. Atiquzzaman, and D. O. Wu, "Edge computing in industrial Internet of Things: Architecture, advances and challenges," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 4, pp. 2462–2488, 4th Quart., 2020.
[2] Y. Liu, S. Garg, J. Nie, Y. Zhang, Z. Xiong, J. Kang, and M. S. Hossain, "Deep anomaly detection for time-series data in industrial IoT: A communication-efficient on-device federated learning approach," *IEEE Internet Things J.*, vol. 8, no. 8, pp. 6348–6358, Apr. 2021.
[3] M. S. AL-Hawawreh and A. I. Zreikat, "Performance analysis of a WIMAX network in different propagation models," *Int. J. Comput. Sci. Inf. Secur.*, vol. 15, no. 1, pp. 603–609, 2017.
[4] M. Al-Hawawreh, F. D. Hartog, and E. Sitnikova, "Targeted ransomware: A new cyber threat to edge system of brownfield industrial Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 7137–7151, Aug. 2019.
[5] K. Okereafor and O. Adelaiye, "Randomized cyber attack simulation model: A cybersecurity mitigation proposal for post COVID-19 digital era," *Int. J. Recent Eng. Res. Develop.*, vol. 5, no. 7, pp. 61–72, 2020.
[6] R. Dudley. (May 2021). *The Colonial Pipeline Ransomware Hackers Had a Secret Weapon: Self-Promoting Cybersecurity Firms*. [Online]. Available: https://www.technologyreview.com/2021/05/24/1025195/colonial-pipeline-r%ansomware-bitdefender/
[7] P. O'Connor, "2020 ecurity review: A year that shook it," *ITNOW*, vol. 62, no. 4, pp. 40–41, Dec. 2020.
[8] M. Humayun, N. Jhanjhi, A. Alsayat, and V. Ponnusamy, "Internet of Things and ransomware: Evolution, mitigation and prevention," *Egyptian Informat. J.*, vol. 22, no. 1, pp. 105–117, Mar. 2021.
[9] M. Akbanov, V. G. Vassilakis, and M. D. Logothetis, "Ransomware detection and mitigation using software-defined networking: The case of WannaCry," *Comput. Electr. Eng.*, vol. 76, pp. 111–121, Jun. 2019.
[10] A. O. Almashhadani, M. Kaiiali, S. Sezer, and P. O'Kane, "A multi-classifier network-based crypto ransomware detection system: A case study of Locky ransomware," *IEEE Access*, vol. 7, pp. 47053–47067, 2019.
[11] A. O. Almashhadani, M. Kaiiali, D. Carlin, and S. Sezer, "MaldomDetector: A system for detecting algorithmically generated domain names with machine learning," *Comput. Secur.*, vol. 93, Jun. 2020, Art. no. 101787.
[12] O. M. Alhawi, J. Baldwin, and A. Dehghantanha, "Leveraging machine learning techniques for windows ransomware network traffic detection," in *Cyber Threat Intelligence*. Cham, Switzerland: Springer, 2018, pp. 93–106.
[13] R. Kozik, M. Pawlicki, and M. Choraś, "A new method of hybrid time window embedding with transformer-based traffic data classification in IoT-networked environment," *Pattern Anal. Appl.*, vol. 24, pp. 1441–1449, May 2021.
[14] N. Mowla, N. H. Tran, I. Doh, and K. Chae, "Federated learning-based cognitive detection of jamming attack in flying ad-hoc network," *IEEE Access*, vol. 8, pp. 4338–4350, 2020.
[15] Q. Xia, W. Ye, Z. Tao, J. Wu, and Q. Li, "A survey of federated learning for edge computing: Research problems and solutions," *High-Confidence Comput.*, vol. 1, no. 1, Jun. 2021, Art. no. 100008.

[16] H. Saadat, A. Aboumadi, A. Mohamed, A. Erbad, and M. Guizani, "Hierarchical federated learning for collaborative IDS in IoT applications," in *Proc. 10th Medit. Conf. Embedded Comput. (MECO)*, Jun. 2021, pp. 1–6.
[17] M. Al-Hawawreh and E. Sitnikova, "Developing a security testbed for industrial Internet of Things," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5558–5573, Apr. 2021.
[18] A. Naser, M. F. Zolkipli, S. Anwar, and M. S. Al-Hawawreh, "Present status and challenges in cloud monitoring framework: A survey," in *Proc. Eur. Intell. Secur. Informat. Conf. (EISIC)*, Aug. 2016, p. 201.
[19] M. Hammad, M. Bsoul, M. Hammad, and M. Al-Hawawreh, "An efficient approach for representing and sending data in wireless sensor networks," *J. Commun.*, vol. 14, no. 2, pp. 104–109, 2019.
[20] Q. Althebyan, Y. Jararweh, Q. Yaseen, and R. Mohawesh, "A knowledge-base insider threat mitigation model in the cloud: A proactive approach," *Int. J. Adv. Intell. Paradigms*, vol. 15, no. 4, pp. 417–436, 2020.
[21] M. Piskozub, R. Spolaor, and I. Martinovic, "MalAlert: Detecting malware in large-scale network traffic using statistical features," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 46, no. 3, pp. 151–154, Jan. 2019.
[22] D. Morato, E. Berrueta, E. Magaña, and M. Izal, "Ransomware early detection by the analysis of file sharing traffic," *J. Netw. Comput. Appl.*, vol. 124, pp. 14–32, Dec. 2018.
[23] J. Modi, I. Traore, A. Ghaleb, K. Ganame, and S. Ahmed, "Detecting ransomware in encrypted web traffic," in *Proc. Int. Symp. Found. Pract. Secur.* London, U.K.: Springer, 2019, pp. 345–353.
[24] V. Rey, P. M. S. Sánchez, A. H. Celdrán, G. Bovet, and M. Jaggi, "Federated learning for malware detection in IoT devices," 2021, *arXiv:2104.09994*. [Online]. Available: http://arxiv.org/abs/2104.09994
[25] T. D. Nguyen, S. Marchal, M. Miettinen, H. Fereidooni, N. Asokan, and A.-R. Sadeghi, "DÏoT: A federated self-learning anomaly detection system for IoT," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2019, pp. 756–767.
[26] X. Hei, X. Yin, Y. Wang, J. Ren, and L. Zhu, "A trusted feature aggregator federated learning for distributed malicious attack detection," *Comput. Secur.*, vol. 99, Dec. 2020, Art. no. 102033.
[27] B. Li, Y. Wu, J. Song, R. Lu, T. Li, and L. Zhao, "DeepFed: Federated deep learning for intrusion detection in industrial Cyber–Physical systems," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5615–5624, Aug. 2021.
[28] R. Taheri, M. Shojafar, M. Alazab, and R. Tafazolli, "Fed-IIoT: A robust federated malware detection architecture in industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 17, no. 12, pp. 8442–8452, Dec. 2021.
[29] W. Schneble and G. Thamilarasu, "Attack detection using federated learning in medical cyber-physical systems," in *Proc. 28th Int. Conf. Comput. Commun. Netw., (ICCCN)*, Jul. 2019, pp. 1–8.
[30] M. Pawlicki, M. Choraś, and R. Kozik, "Defending network intrusion detection systems against adversarial evasion attacks," *Future Gener. Comput. Syst.*, vol. 110, pp. 148–154, Sep. 2020.
[31] B. Flowers, R. M. Buehrer, and W. C. Headley, "Evaluating adversarial evasion attacks in the context of wireless communications," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 1102–1113, Aug. 2019.
[32] A. Kuppa, S. Grzonkowski, M. R. Asghar, and N.-A. Le-Khac, "Black box attacks on deep anomaly detectors," in *Proc. 14th Int. Conf. Availability, Rel. Secur.*, Aug. 2019, pp. 1–10.
[33] K. Niwa, N. Harada, G. Zhang, and W. B. Kleijn, "Edge-consensus learning: Deep learning on P2P networks with nonhomogeneous data," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 668–678.
[34] S. Rifai, G. Mesnil, P. Vincent, X. Muller, Y. Bengio, Y. Dauphin, and X. Glorot, "Higher order contractive auto-encoder," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Berlin, Germany: Springer, 2011, pp. 645–660.
[35] P. Xiong, H. Wang, M. Liu, F. Lin, Z. Hou, and X. Liu, "A stacked contractive denoising auto-encoder for ECG signal denoising," *Physiol. Meas.*, vol. 37, no. 12, pp. 2214, Nov. 2016.
[36] M. Al-Hawawreh, E. Sitnikova, and N. Aboutorab, "X-IIoTID: A connectivity-and device-agnostic intrusion dataset for industrial Internet of Things," *IEEE Internet Things J.*, early access, Aug. 3, 2021, doi: 10.1109/JIOT.2021.3102056.
[37] *BotNet and Ransomware Detection Datasets University of Victoria*. Accessed: Nov. 2020. [Online]. Available: https://www.uvic.ca/engineering/ece/isot/datasets/botnet-ransomware/ind%ex.php
[38] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symp. Comput. Intell. Secur. Defense Appl.*, Jul. 2009, pp. 1–6.
[39] R. Mohawesh, S. Tran, R. Ollington, and S. Xu, "Analysis of concept drift in fake reviews detection," *Expert Syst. Appl.*, vol. 169, May 2021, Art. no. 114318.
[40] M. Al-Hawawreh and E. Sitnikova, "Leveraging deep learning models for ransomware detection in the industrial Internet of Things environment," in *Proc. Mil. Commun. Inf. Syst. Conf. (MilCIS)*, Nov. 2019, pp. 1–6.
[41] S. Abdulrahman, H. Tout, A. Mourad, and C. Talhi, "FedMCCS: Multicriteria client selection model for optimal IoT federated learning," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4723–4735, Mar. 2021.
[42] S. Chatterjee and M. K. Hanawal, "Federated learning for intrusion detection in IoT security: A hybrid ensemble approach," 2021, *arXiv:2106.15349*.
[43] A. Rawashdeh, M. Alkasassbeh, and M. Al-Hawawreh, "An anomaly-based approach for DDoS attack detection in cloud environment," *Int. J. Comput. Appl. Technol.*, vol. 57, no. 4, pp. 312–324, 2018.

MUNA AL-HAWAWREH received the B.E. and M.E. degrees in computer science from Mutah University, Jordan. She is currently pursuing the Ph.D. degree with the University of New South Wales (UNSW), Canberra, Australia. She works as a Research Assistant at UNSW Canberra Cyber. In her Ph.D. degree, she developed the world's first ransomware framework targeting IIoT edge gateway in the critical infrastructure. Her research interests include cloud computing, industrial control systems, the Internet of Things, cybersecurity, and deep learning. She is a program committee member and a reviewer for several cybersecurity conferences. She was awarded the First Prize for high impact publications in the School of Engineering and Information Technology (SEIT), UNSW, in 2019, and the Dr. K. W. Wang Best Paper Award (2018–2020). She is a Reviewer of high-impact factor journals, such as the IEEE Internet of Things Journal, IEEE Transactions on Industrial Informatics, and IEEE Transactions on Dependable and Secure Computing.

ELENA SITNIKOVA (Member, IEEE) received the B.E. degree (Hons.) in electrical engineering and Ph.D. degree in communication control systems. She is currently an Award-Winning Academic and a Researcher at the University of New South Wales (UNSW), Canberra, and the Australian Defence Force Academy (ADFA). Her current research interests include the critical infrastructure protection area, carrying out research projects in intrusion detection (IDS) for control systems cyber security, cyber-physical systems, and the Industrial IoT (IIoT). She is one of the first Australians to be certified in CSSLP—Certified Secure Software Lifecycle Professional. She is holding a Senior Fellowship of the Higher Education Academy (SFHEA) and the Australian Office for Learning and Teaching (OLT) Team Citation Award for Outstanding Contributions to Student Learning.



NEDA ABOUTORAB (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from The University of Sydney, Sydney, Australia, in 2012. She is currently a Senior Lecturer at the School of Engineering and Information Technology, University of New South Wales, Canberra, Australia. From 2012 to 2015 and before joining the University of New South Wales, she was a Postdoctoral Research Fellow at the Research School of Engineering, The Australian National University. Her research interests include network coding, wireless communications, data caching and storage systems, the Internet of Things, and signal processing.