Federated Intrusion Detection In NG-IoT Healthcare Systems: An Adversarial Approach

Iliias Siniosoglou†, Panagiotis Sarigiannidis‡, Vasilis Argyriou†, Thomas Lagkas§, Sotirios K. Goudos¶ and Maria Poveda∥

Abstract—In recent years and with the advancement of IoT networks, malicious intrusions aiming at disrupting the services and getting access to confidential information in medical environments is ever progressing. To that end, this paper proposes a Federated Layered Architecture to be used in Medical Cyber-Physical Systems (MCPS) Networks that entails the creation of multiple aggregation layers to induce further security to the model training process. Moreover, two Deep Adversarial Neural Networks (GANs) are presented for use with data found in the MCPS environment. The evaluation of the presented work showed that the models trained in the Federated system have an increase in their ability to detect possible intrusions in the MCPS network than the commonly trained models.

Index Terms—Anomaly Detection, Auto-encoder, Cybersecurity, Generative Adversarial Network, Deep Learning, Machine Learning, Critical Infrastructures, Medical Cyber-Physical Systems

I. INTRODUCTION

In the modern age and the era of global network interconnection, most contemporary digital systems are characterized by a plethora of Internet of Things (IoT) devices connected with each other and communicating various information in order to provide a level of quality of service to the respective environment [1], and in extent in the Next-Generation IoT (NG-IoT) networks comprised of human-centric approaches, AI enabled devices, fog networks and other. MCPS are no exception to the aforementioned schemes, as they communicate critical information in a seamless interconnected network of priority devices. Specifically, MCPS describe networks comprised of medical and healthcare related devices in charge of medical-oriented tasks or information dissemination of health/patient related context in medical environments such as hospitals, infirmaries, etc. They help monitor patient condition, oversee medical data analysis and archiving, even patient treatment through specialized equipment [2] [3], such as ICU monitors, medical databases, laboratory equipment, healthcare actuators and other.

Important as they are, though, MCPS systems face certain problems like component heterogeneity, inoperability, high assurance and other serious setbacks interjecting with their crucial constitution [4][5]. Among these problems is the security and privacy of the MCPS networks as a part of IoT/NG-IoT networks [6]. The amount of diverse information, the increased use of wireless networks for information sharing and the lack of proper security infrastructures leave MCPS systems vulnerable to suspicious activity and, due to the critical nature of the network, possibly catastrophic consequences.

To tackle possible threats and secure the network’s subsequent privacy, a variety of security systems have been developed on top of common practices. Intrusion Detection and Prevention Systems (IDPS), leveraging various techniques like traffic monitoring, Machine Learning (ML) and Deep Learning (DL), signature pattern identification, have produced numerous encouraging results towards ensuring the fortification and further integrity of their corresponding environment. A problem arises, though, because of the sensitive nature of the data, being patient records or laboratory tests and results, their volume since every device is constantly producing data over the network and heterogeneity due to the different nature of each device network. Federated Learning (FL), a new DL methodology for decentralized model training copes with these setbacks by training DL models, locally, onto the different edge IoT devices [7] without the need to communicate those data to a central system, only sending model weight updates, thus avoiding possible network overhead and keeping the data private.

In this paper we introduce a Federated Learning architecture for Intrusion Detection (ID) in MCPS networks by utilizing the Generative Adversarial Network (GAN) DL scheme. GAN networks have become powerful tools used in ID due to their adversarial training that pushes them to create strong DL models. Specifically, a multi-layer federated heap is produced, utilizing different data description from the MCPS network environment in order to successfully identify anomalies that may indicate an ongoing cyber-attack to the medical network. In this system, GAN models are trained on two distinct data categories, namely, a) Patient Medical Data and b) Network Traffic Data, using publicly available datasets, to simulate an MCPS data ecosystem. Consequently, quantitative metrics are used to measure the performance of both the GAN models.
and the FL architecture. The contributions of this paper can be summarized as follows:

- Producing a Federated Layered Architecture to be used in MCPS Networks: The FL scheme was devised with respect to the medical environment as a Critical Infrastructure (CI) and to its need for continuous operation.
- Delivering an Online Federated Adversarial Training Method for core GAN Training logic: The produced system offers a generalized method for GAN training in order to accommodate a variety of GAN designs.
- Presenting a GAN network for two Intrusion Detection problems: Two GAN models are produced and tested based on two morphologically different Intrusion Detection Data approaches.

The rest of this paper is organised as follows. Section II presents relevant work. Section III provides a basic background to the tools and methods used in this work. Section IV analyses the Federated architecture and the produced DL models, while Section V presents the evaluation of the introduced system. Finally, Section VI concludes this work.

II. RELATED WORK

In the field of MCPS Intrusion Detection some worth mentioning work has been previously performed. In [2], the authors produce a distributed intrusion detection system for MCPS networks. Specifically, the work presents a Federated Learning IDS (FLIDS), based on the architecture developed in [8], adapted to the MCPS usecase by utilizing distributed Multilayered Perceptions trained on the MIMIC public dataset containing Intensive Care Unit (ICU) data. The designed system architecture supports distributed IDSs on mobile devices controlling Body Area Networks (BAN) communication [9] with a hospital back-end server. The work targets only the i)Denial of Service (DoS), ii)Data Modification, iii)Data Injection and iv)Eavesdropping attack categories usually seen on healthcare networks. The FL system groups data based on identifying factors in order to generalize the produced global models and reduce computation and communication costs. The evaluation of the proposed system shows a significant increase in communication efficiency while keeping high accuracy score and low false positive rate (FPR) on the FL anomaly detection.

In [10], the authors explore the vulnerabilities of personal healthcare systems while applying a series of available exploitations to those devices. Subsequently, they present the HEKA IDS system that is trained to recognize medical data traffic, extracted with an n-gram based approach using the sequential patterns produced by the Personal Medical Devices (PMDs), in the network using different Machine Learning techniques. The proposed system, tested against a series of available cyber attacks, shows that HEKA achieves high accuracy scores over different configurations.

In [11] a novel method for distributed Generative Adversarial Deep Neural Networks is proposed. In particular, the authors present MD-GAN, that adapts the GAN system in a Federated fashion and test the produced system over three major DL public datasets, namely, a) MNIST, b) CIFAR10 and c) CelebA, while addressing distributed scalability issues. The results show that the produced implementation achieves high score among the datasets by adapting to the Federated procedure.

III. BACKGROUND

In this section, the necessary background for the tools and methods used in this work are presented.

A. GAN Architecture

The GAN architecture, is based on a pair of neural sub-networks, a) the Generator $G$ and b) the Discriminator $D$, that play and adversarial game with each other [12], [13]. $G$ usually takes an input random noise data and tries to produce similar-to-the-real data of the given usecase. On the other hand, $D$ is trained to distinguish between the fake data produced by $G$ and the real samples. The GAN architecture aims in training both sub-networks, that rival each other, so that the Generator can produce realistic samples that the Discriminator can’t differentiate from the real input data and vice versa. Equation 1 below shows the relation of $G$ and $D$.

$$
\min_{G} \max_{D} V(G,D) = \min_{G} \max_{D} \mathbb{E}_{x \sim p_{data}} [\log(D(x))] + \mathbb{E}_{z \sim p_z} [\log(1 - D(G(z)))]
$$

where $G$ accumulates noise $z$ from space $Z$ mapping it to the space $X$ from which $D$ inputs $x$. $p_{data}(x)$ and $p_z(z)$ denote the probabilistic distribution of Spaces $X$ and $Z$ respectively. For the Anomaly Detection procedure an intermediate model is exported after the training from the Discriminator module. This model is the part of the Discriminator from the input up to a latent layer before the output sequence of the network. This intermediate model is used to reduce the input dimension of the input samples into a specified latent space. To calculate the anomaly score of each sample, the Adversarial Loss ($L_{adv}$) of the two samples is computed. The $L_{adv}$ Loss is the normal distance between the generated/fake and the real sample. Since the Generator has learned to produce normal samples, the greater the Adversarial loss the bigger the probability of the input sample being abnormal. Equation 2 describes the Adversarial Loss as

$$
L_{adv}(x) = \|d_r - d_p\|
$$

where $L_{adv}$ is the Adversarial loss score of the function, $d_r$ and $d_p$ is the prediction of the latent model on the real and generated sample respectively.

B. Federated Learning

Federated Learning is a stochastic distributed learning and privacy-preserving method that undertakes the distribution, orchestration, learning and aggregation of Deep Learning model across a big corpus of devices or edge nodes in the cloud [14] [15]. It works by stochastically disbursing a central Deep Learning model over a given corpus of devices in order to train locally on the on-device collected data. Consequently,
the models are send back to the centralized system to be aggregated in a process called Federated Averaging [7], which is defined as the aggregation of the edge-computed weights with the central model.

Specifically, the central server distributes a global model $w^0_{Global}$, along with training instructions to a Federated population $P_f \in [1, N]$ where $N \in \mathbb{N}^+$, each holding a set of local data $D_i \in N$ and local models $w_i^l$. The distributed models are subsequently trained on the local data $D_i$ and then the weights $w^l_{Global}$ are send back to the central system to be aggregated through the Federated Averaging (3), or similar, process in order to produce an updated global model $w^k_{Global}$ [16].

$$w^k_{Global} = \frac{1}{\sum_{i\in N} D_i} \sum_{i=1}^N D_i w^k_i$$

(3)

Where $w^k_{Global}$ is the global model at the $k_{th}$ training iteration and $w^k_i$ denotes the Federated population $i_{th}$ model at that iteration.

IV. DESIGN & IMPLEMENTATION

In this section the architecture design is laid out and the methodology implemented is explained. In particular, the devised Federated MCPS Environment and its interactions are described. Moreover, the MCPS Federated GAN architecture is presented in respect to the usecases used in this work.

A. Federated MCPS Network

As mentioned, the MCPS ecosystem comprises of critical equipment and data transmission pathways that play an important role in the stability of the underlying goals of the healthcare system and in extent the preservation of human life. As such, this environment has small to zero tolerance for the obstruction of the operation of those critical elements. With that in mind the proposed architecture supports abstraction of the Federated aggregation process into single or multiple aggregating subsystems that can be decoupled from the internal MCPS network while at the same time preserving the privacy of the data. Specifically, it composed of a number of sub-layers that support both vertical and horizontal expansion. The base layer of the systems, as can be seen in Figure 1, consists of the medical equipment of the different networks in a healthcare system, which produce data. The data acquired from those devices are aggregated in $K$ gateway device responsible for monitoring the said equipment, where $K$ is considered the total of the smallest medical device network in an MCPS environment. These edge IoT devices host the local IDS system correlated with its connected devices. Over the Data Aggregation Layer (DAL) exist $L \in [1, K/2]$ Federated Aggregation Layers (FAL). The design leaves room for extra Federated Aggregation Layers in the case of a Wide Area MCPS ecosystem in order to optimize the information abstraction of homogeneous DL models and thus minimize any possible data interception in any level of the architecture. Every DAL hosts $N$ number of Federated Aggregators, where $N \in [1, P_f]$, and depending on the layers’ configuration. Finally, the last DAL is connected to the Global MCPS Model server that keeps track of the MCPS Federated populations, global shared models and model versions and to which the aggregated models are communicated to.

To understand the averaging of the Federated models throughout the Federated procedure we have the following relation (4)

$$w^k_{L_{Group}} = \frac{1}{N_A} \sum_{i=1}^{N_A} w^k_i$$

(4)

where $w^k_{L_{Group}}$ denotes the averaged model weights of a specific group in a layer over the $k_{th}$ iteration which is defined by the average of the weights of the Aggregators of that certain population of size $N_A$ at that iteration. Generalizing the notation, to find the averaged model of a layer $L$ we use (5)

$$f^k_{w_l}(l, group) = \frac{1}{N_{group_1}} \sum_{i=1}^{N_{group_1}} f^k_{w_j}(l-1, group)$$

(5)

where $f^k_{w_l}(l, group)$ returns the averaged weight of the $l_{th}$ FAL, $l \in [0, L]$, of a certain Federated group. $w_j$ denotes the $j_{th}$ Aggregator’s weights of the group in a layer.

B. Federated GAN Training Methodology

At this point we examine the training between the lower Federated Aggregation Layer and the Data Aggregation Layer.

The process begins by the FAL receiving a request for a model update on a specific Federated population. The FAL stochastically decides on a subset of data gateways of the mentioned group. Once it decides on a number of devices it sends a training signal to those gateways. Since the GAN architecture consists of two sub-networks it initializes the weights of the said networks, or in the case of multiple FALs it forwards the global model it received from the global server, and sends them to the gateways. For every iteration of the Federated training and since this is an online training, it sends
the networks to the nodes and at the end of the iteration it receives them back so to aggregate them with the rest of the received models. Algorithm 1 shows the in-depth procedure of the GAN model training.

Algorithm 1: Federated GAN Training

Data: $F_{\text{group}}$ - Federated population of certain group $w_{0_{G}}$ - Global G model before the Federated Training $w_{0_{D}}$ - Global D model before the Federated Training $\text{numOfIterations}$ - Number of Federated Iterations

Result: $w^{k}_{G_{\text{client}}}, w^{k}_{D_{\text{client}}}$ - Global models after k iterations

$F_{\text{clients}} = \text{SelectSubsetOfFedClients}(F_{\text{group}})$;
$S = F_{\text{group}}.\text{size}()$;
$k = 0$;

while $k < \text{numOfIterations}$ do
  for each $F_{\text{client}} \in F_{\text{clients}}$ do
    $w^{k}_{G_{\text{client}}} = \text{trainGeneratorOnClient}(F_{\text{client}})$;
    $w^{k}_{D_{\text{client}}} = \text{trainDiscriminatorOnClient}(F_{\text{client}})$;
  end
  $w^{k}_{G} = \frac{1}{S} \sum_{i=1}^{S} w^{k}_{G_{i}}$
  $w^{k}_{D} = \frac{1}{S} \sum_{i=1}^{S} w^{k}_{D_{i}}$
  $k = k + 1$;
  for each $F_{\text{client}} \in F_{\text{clients}}$ do
    sendTrainedGenerator($F_{\text{client}}, w^{k}_{G_{\text{client}}}$);
    sendTrainedDiscriminator($F_{\text{client}}, w^{k}_{D_{\text{client}}}$);
  end

It is worthy to mention that a security problem arises from the nature of the GAN architecture itself. Since the Generator of the GAN system is trained to produce realistic samples that mimic the data it trained on, any possible interception of the G module could potentially reveal confidential data to the malicious party that intercepted it. This particular problem can be overcome by enforcing a Differential privacy scheme and further cryptographic approaches on the Federated communication model [17]–[19], though this is out of the scope, and will be looked upon on further extension of this work.

C. Federated GAN Model For Medical Data

One of the goals of this work is to produce a GAN Deep Neural Network model to be used for intrusion detection from healthcare data. To that end, a compact GAN architecture was produced in order to accommodate the communicational needs of the Federated environment and the ID demands. The network is trained on normal medical data from homogeneous sources and tested on data with added abnormalities. The network is defined as follows.

The Generator $G$ takes in a noise vector of size 10 and is trained to produce realistic samples of size $N_{D_{i}}$, close to the real data, where $D_{i}$ is the length of the feature vector. The Generator’s structure consists of eight layers, an input Dense layer, an output Tanh layer (6) and a sequence of Dense, ReLU, LeakyReLU, Batch 1DNormalization and Dropout. On the other hand, the Discriminator $D$ takes in a sample vector of size $N_{D_{i}}$. $D$ is trained discriminate the input samples as real or fake. The Discriminator’s structure consists of eight layers, an input Dense layer, an output Sigmoid layer (7) and a sequence of Dense, ReLU, LeakyReLU, Batch 1DNormalization and Dropout. The overall structure of the network and its sub-networks can be seen in figure 2.

$$\tanh(x) = 2s(2x) - 1, \tanh \rightarrow [-1, 1] \quad (6)$$

where equation (6) describes the Tanh function. $\tanh(x)$ is the output of the $\tanh$ function, $s(x)$ is a Sigmoid function (7) and $x$ is the input vector.

$$s(x) = \frac{1}{1 + e^{-x}}, s \rightarrow [0, 1] \quad (7)$$

Both sub-networks are compiled with the Binary Cross-Entropy function (8) and the Adam [20] optimizer with a learning rate parameter of 0.01.

$$H_{p}(q) = -\frac{1}{N} \sum_{i=1}^{N} y_{i} \cdot \log(p(y_{i})) + (1 - y_{i}) \cdot \log(1 - p(y_{i})) \quad (8)$$

The Binary Cross-Entropy function is defined as follows. $N$ is the number of samples given, $y$ is the label. $p(y_{i})$ is the probability of the sample being a match to the label sample when $1 - p(y_{i})$ presents the inverse of that probability. Finally, $H$ represents the result of the Binary Cross-Entropy loss in a given point.

Fig. 2: Anomaly Detection Medical Data GAN Model

Though the size of the presented GAN is aimed for a modest amount of features that was decided on based on the data
distribution of common MCPS equipment, it can be scaled using relation (9) which was empirically found to work in the presented scheme of the GAN structure for one-dimensional data input.

\[ h_{size} = \frac{2^{log_2(M)} / 2^{l_i-1}, l \rightarrow log_2(M)}{(9)} \]

where \( h_{size} \) is the size of the hidden layer, \( M \) denotes the number of features in the dataset, and \( l_i \) is the current layer.

**D. Federated GAN Model For Medical Network Traffic Data**

Except the Data produced on healthcare devices, i.e. patient monitoring data, reports and other, Intrusion Detection can be achieved on the network flow level. Utilizing the network flows describing the communication over the MCPS network, over the different protocols used by medical equipment, a significant portion of incoming intrusions or malicious third party meddling can be detected. Consequently, this work presents a novel compact GAN architecture for network flow Anomaly detection. The produced models are trained on normal network traffic data, which is by its nature homogeneous, and tested on data containing a variety of different network exploitations. The GAN sub-networks are defined as follows.

\[ G \] takes in a noise vector of size \( z = 10 \), where \( z \) is the length of the noise vector, and is trained to produce realistic samples of size \( M \) number of flow features, imitating the real data. The Generator’s structure consists of ten layers, an input Dense layer, an output Tanh layer (6) and a sequence of Dense, ReLU, LeakyReLU, Batch 1DNormalization, and Dropout. On the other hand, \( D \) takes in a sample vector of size \( N_{D_v} \). \( D \) is trained discriminate the input samples as real or fake. The Discriminator’s structure consists of eight layers, an input Dense layer, an output Sigmoid layer (7) and a sequence of Dense, ReLU, LeakyReLU, Batch 1DNormalization and Dropout. The overall structure of the network and its sub-networks can be seen in figure 3.

Both sub-networks are compiled with the Binary Cross-Entropy function (8) and the Adam optimizer with a learning rate parameter of 0.001.

**V. Evaluation**

In order to evaluate the produced architecture and Deep Learning model, the presented system was tested and measured using a series of quantitative metrics.

**A. Evaluation Environment**

To test the proposed Federated scheme an experimental testbed was devised. Due to the fact that MCPS environments, as Critical Infrastructures, are characterized by the need for continuous and unperturbed operation but also by their confidential nature, a virtual testbed was built. This installation consists of an Ubuntu 20.04 system having the role of the global Control and Command (C&C) server, a subsystem within this server serving as the lower FAL layer and 6 virtual remote Data Aggregators, three for each usecase, namely, a) Medical Data and b) Network Traffic Data, providing the respective data. To each remote worker was distributed a subset of the training data, so to avoid training over the same training batches.

The training process began with a new untrained model send to the remote workers form training, aiming to evaluate the full potential of the Federated procedure.

**B. Medical Data Anomaly Detection**

For the Medical Data Anomaly Detection usecase the CHARIS [21] public clinical dataset was used. This dataset contains multi-channel recordings of ECG, arterial blood pressure (ABP), and intracranial pressure (ICP) of patients diagnosed with traumatic brain injury (TBI). To simulate malicious meddling with the data, a portion was selected as testing data and half of those were enriched with a Gaussian distribution of noise with a factor of 2 based on the deviation of the same data. The alternated samples differ by a fraction from the real data while introducing the effects of the i) Data Modification and ii) Data Scrambling cyber attacks. The data were fed into the Federated GAN network as time-series with a window size of 10 records and normalized in the range of (0, 1) using a Min-Max Scaler.

After a number of experiments with the Federated Medical Data Anomaly Detection part of this work it was observed that the model converged over a small number of Federated iterations. The results show that the Federated procedure produces better metrics that the non-Federated model, as can be seen in Table I.

**C. Network Traffic Data Anomaly Detection**

For the Network Traffic Data Anomaly Detection usecase the UNSW-NB 15 [22], [23] public Intrusion Detection dataset was utilized. This dataset contains network flows containing normal and abnormal traffic from a test bed environment. The abnormal samples contain a series of cyber attacks to be used
for the detection test of the algorithm, namely, i) Fuzzers, ii) Analysis, iii) Backdoors, iv) DoS, Exploits, v) Generic, vi) Reconnaissance, vii) Shellcode and viii) Worms. For the anomaly detection part the attack classes were converted to binary ground-truth labels depicting Normal and Abnormal traffic. The data were fed into the Federated GAN networks single records and normalized in the range of (0,1) using a Min-Max Scaler.

After a number of experiments with the network Traffic Data Anomaly Detection part of this work it was observed that, even though the results of the network were not very high at the Anomaly Detection task, the Federated model showed an improvement against the non-Federated, as can be seen in Table I.

| Model                       | ACC    | REC    | PRE    | F1     | AUC    |
|-----------------------------|--------|--------|--------|--------|--------|
| Medical Anomaly Detection   | 0.9151 | 0.8969 | 0.9167 | 0.9151 |
| FL Medical Anomaly Detection| 0.9298 | 0.9054 | 0.9518 | 0.9283 | 0.9298 |
| Network Flow Anomaly Detection | 0.7674 | 0.8139 | 0.7446 | 0.7777 | 0.7674 |
| FL Network Flow Anomaly Detection | 0.7837 | 0.8056 | 0.7718 | 0.7883 | 0.7838 |

TABLE I: Federated and non-Federated model result comparison

VI. CONCLUSION

While the advancement in Intrusion Detection and Prevention system is evolving continuously in most modern networked systems, some cyber-physical systems, like most healthcare networks, that were supposed to be isolated are left unprotected due to the restricted work done their involved field. Due to their critical nature, security plays a major role to the fortification of the integrity and continuous operation of those systems. To that end, this paper proposes a distributed layered IDS scheme, enriched with the power of a novel Deep Learning training method, Federated Learning. The presented system leverages the Adversarial architecture of GAN Deep Neural networks, tested on two distinct usecases, namely, a) Patient Medical Data and b) Network Traffic Data, to both produce Federated modes capable of detecting possible intrusions in a MCPS network, in a distributed manner, but also to establish the use of Federated Learning in the IDS methodology. Evaluating the proposed networks showed an increase in the Federated model’s ability to detect anomalies in both usecases than the non-Federated Models.

Our future plans in this field include the implementation of a range of DL models for Federated Training and augment them with privacy preserving policies to further secure the implementation.

VII. ACKNOWLEDGMENT

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 957406 (TERMINET).

REFERENCES

[1] A. Triantafyllou, P. Sarigiannidis, and T. Lagkas, “Network protocols, schemes, and mechanisms for internet of things (iot): Features, open challenges, and trends,” Wireless Communications and Mobile Computing, vol. 2018, pp. 1–24, 09 2018.
[2] W. Schneble and G. Thamilarasu, “Attack detection using federated learning in medical cyber-physical systems,” 08 2019.
[3] S. Misra and S. Sarkar, “Priority-based time-slot allocation in wireless body area networks during medical emergency situations: An evolutionary game-theoretic perspective,” IEEE Journal of Biomedical and Health Informatics, vol. 19, no. 2, pp. 541–548, 2015.
[4] N. Dey, A. S. Ashour, S. Fuqian, S. Fong, and J. Tavares, “Medical cyber-physical systems: A survey,” Journal of Medical Systems, vol. 42, no. 7, 04 2018.
[5] A. Humayed, J. Lin, F. Li, and B. Luo, “Cyber-physical systems security—a survey,” IEEE Internet of Things Journal, vol. 4, no. 6, pp. 1802–1831, 2017.
[6] P. Rodologlu Grammatikis, P. Sarigiannidis, and I. Moscholios, “Securing the internet of things: Challenges, threats and solutions,” 11 2018.
[7] B. McManah, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, “Communication-Efficient Learning of Deep Networks from Decentralized Data,” ser. Proceedings of Machine Learning Research, 2017, pp. 1273–1282.
[8] J. Konečný, H. McMahan, D. Ramage, and P. Richtárik, “Federated optimization: Distributed machine learning for on-device intelligence,” 10 2016.
[9] S. Misra, S. Moulik, and H.-C. Chao, “A cooperative bargaining solution for priority-based data-rate tuning in a wireless body area network,” Wireless Communications, IEEE Transactions on, vol. 14, pp. 2769–2777, 05 2015.
[10] A. I. Newaz, A. K. Sikder, L. Babun, and A. S. Uluagac, “Heka: A novel intrusion detection system for attacks to personal medical devices,” in 2020 IEEE Conference on Communications and Network Security (CNS), 2020, pp. 1–9.
[11] C. Hardy, E. Le Merrer, and B. Sericola, “Md-gan: Multi-discriminator generative adversarial networks for distributed datasets,” in 2019 IEEE International Parallel and Distributed Processing Symposium (IPDPS), 2019, pp. 866–877.
[12] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. Bharath, “Generative adversarial networks: An overview,” IEEE Signal Processing Magazine, vol. 35, 10 2017.
[13] Y. Hong, U. Hwang, J. Yoo, and S. Yoon, “How generative adversarial nets and its variants work: An overview of gan,” ACM Computing Surveys, vol. 52, 11 2017.
[14] K. Bonawitz, H. Eichner, W. Grieskamp, D. Huba, A. Ingerman, V. Ivanov, C. Kiddon, J. Konečný, S. Mazzocchi, H. McMahan, T. Overveldt, D. Petrou, D. Ramage, and J. Roselander, “Towards federated learning at scale: System design,” 02 2019.
[15] S. Misra, S. Chatterjee, and M. S. Obaidat, “On theoretical modeling of sensor cloud: A paradigm shift from wireless sensor network,” IEEE Systems Journal, vol. 11, no. 2, pp. 1084–1093, 2017.
[16] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y. C. Liang, Q. Yang, D. Niyato, and C. Miao, “Federated learning in mobile edge networks: A comprehensive survey,” IEEE Communications Surveys Tutorials, vol. 22, no. 3, pp. 2031–2063, 2020.
[17] R. Geyer, T. Klein, and M. Nabi, “Differentially private federated learning: A client level perspective,” 12 2017.
[18] S. Truex, N. Baracaldo, A. Anwar, T. Steinke, H. Ludwig, R. Zhang, and Y. Zhou, “A hybrid approach to privacy-preserving federated learning,” 11 2019, pp. 1–11.
[19] M. Hao, H. Li, X. Luo, G. Xu, H. Yang, and S. Liu, “Efficient and privacy-enhanced federated learning for industrial artificial intelligence,” IEEE Transactions on Industrial Informatics, vol. 16, no. 10, pp. 6532–6542, 2020.
[20] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” International Conference on Learning Representations, 12 2014.
[21] N. Kim, A. Krasner, C. Kosinski, M. Winingier, M. Qadri, Z. Kappus, S. Danish, and W. Cruelius, “Trending autoregulatory indices during treatment for traumatic brain injury,” J Clin Monit Comput, vol. 30, pp. 1–11, 10 2015.
[22] N. Moustafa and J. Slay, “Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set),” in 2015 Military Communications and Information Systems Conference (MICICS), 2015, pp. 1–6.
[23] N. Moustafa and J. Slay, “The evaluation of network anomaly detection systems: Statistical analysis of the unsw-nb15 data set and the comparison with the kdd99 data set,” Information Security Journal: A Global Perspective, vol. 25, no. 1–3, pp. 18–31, 2016.