Secure Peer-to-Peer Federated Learning for Efficient Cyberattacks Detection in 5G and Beyond Networks

Fahdah Alalyan, Badre Bousalem, Wael Jaafar, Rami Langar

***Abstract*—The Open radio access network (O-RAN) supports the multi-class wireless services required in beyond 5th-generation (B5G) mobile networks. However, it also increases the threat surface, thus requiring enhanced cyberattack detection mechanisms. To do so, advanced Artificial Intelligence (AI) algorithms combined with RAN intelligent controllers (RICs) can be leveraged to detect cyberattacks, such as distributed denial-of-service (DDoS) attacks. Nevertheless, data privacy becomes a significant concern when using AI-based operations. To bypass this issue, secured Federated Learning (FL) can be leveraged. Specifically, training cyberattack detection models locally and securely communicating the models' data for aggregation would guarantee protection against eavesdropping. In addition, the usage of Peer-to-Peer (P2P) FL would allow to avoid the centralized FL's single point of failure. However, securing P2P FL with encryption/decryption or using the Secure Average Computation (SAC) would incur high communication costs that scale poorly with the number of FL clients. Hence, we propose in this paper a novel P2P FL strategy that guarantees secure FL, while significantly reducing the communication cost. Specifically, we incorporate client selection and transfer learning within the RIC-based P2P FL system to detect cyberattacks. Through experiments, we demonstrate our method's performances across different scenarios with both balanced and unbalanced dataset distributions. Finally, its superiority in terms of accuracy, robustness, and cost, compared to existing benchmarks, is illustrated.**

***Index Terms*—5G, Cybersecurity, Cyberattack, FL.**

I. Introduction

Wireless communication technology has become a critical enabler of emerging technologies such as vehicle-to-everything (V2X) networks, smart infrastructure, autonomous vehicles, and the Internet-of-Things (IoT) [1]. Moreover, various emerging applications such as virtual reality (VR) and Artificial Intelligence (AI) are being rapidly deployed, resulting in massive volumes of data traffic [2]. Consequently, wireless communications have undergone several transformations in the past decades. With transitions in cellular networks towards the fifth-generation (5G) and beyond (B5G) [3], 5G networks are able to support heterogeneous devices, offering them computational resources and seamless connectivity for intelligent and autonomous operations [1]. In addition, 5G facilitates the immersive growth of data transmission by providing higher data rates and lower latency [4].

This work was supported in part by the ANR 5G-INSIGHT project (Grant no. ANR-20-CE25-0015), and in part by the Innovation for Defence Excellence and Security (IDEaS) program of the Department of National Defence Canada.
Fahdah Alalyan, Wael Jaafar, and Rami Langar are with the École de Technologie Supérieure (ÉTS), Canada (e-mail: fahdah.alalyan.1@ens.etsmtl.ca; wael.jaafar@etsmtl.ca; rami.langar@etsmtl.ca).
Badre Bousalem and Rami Langar are with the University Gustave Eiffel, France (e-mail: badre.bousalem@univ-eiffel.fr; rami.langar@univ-eiffel.fr).

However, the advent of 5G brings its share of challenges. Indeed, the complexity of 5G systems expands the threat surface and makes it hard to define system boundaries [5]. Furthermore, the early stages and rapid deployment of 5G have led to a need for greater awareness of threats. For instance, softwarization, virtualization, and cloudification are critical for the network's performance but do introduce several security breach opportunities. In the same logic, open radio access networks (O-RANs) enhance 5G multi-vendor interoperability; however, their risks are significantly high due to the O-RANs' inherent open and modular architecture [6]. Hence, enhanced security measures, such as cyberattack detection, are indispensable for O-RAN in 5G. While a considerable amount of research on cyberattack and anomaly detection using machine learning (ML) in RAN has been done, only a few studies have focused explicitly on O-RAN [7].

AI, in particular ML, provides robust, innovative, and dynamic solutions for privacy, security, and threat detection in B5G systems. Yet, one significant challenge lies in achieving secure and private knowledge share between the ML-based detection agents [4]. Federated Learning (FL) is a compelling alternative to guarantee data privacy in such a context. This distributed ML technique is primarily aimed at ensuring privacy-preserved collaborative training through sharing model updates instead of explicitly sharing or accessing raw training data [8]. As such, FL proves to be more suitable than conventional ML for data privacy. Despite its merits, centralized FL is prone to defects such as the single point of failure and imbalanced data distribution. The introduction of Peer-to-Peer (P2P) FL mitigates, to some extent, these issues [7].

In the context of cyberattack detection, several FL-based mechanisms have been proposed. In particular, P2P FL is suggested for complex O-RAN environments, owing to the hierarchical architecture of the RAN intelligent controllers (RICs) and data-driven inputs via open interfaces [9]. Amid this development, a refined method has emerged, called P2P FL with Secure Average Computation (SAC) [7], and utilizes averaging and n-out-of-n secret partitioning to counter semi-honest participants. Despite its interesting results in terms of accuracy, this approach may incur high communication costs in large-scale systems. Authors of [7] proposed another variation of SAC-based P2P FL, where K-means forms clusters based on the clients' locations. In this setup, SAC operates within a given cluster instead of among all peers. Even though this approach is inherent in the structure of peer networks and can minimize the communication cost of SAC with clustering, this

operation did not rely on characteristics intrinsic to the peers' local datasets or other common criteria, such as data similarities or peer performance. Alternatively, authors in [10] proposed Performance-Based Neighbor Selection (PENS), where a client shares its model and training loss with others targeting to form a cluster with clients with similar data distributions. Nevertheless, by sharing their models and training loss, clients could potentially disclose sensitive information from their local datasets.

To circumvent the aforementioned issues, we propose in this paper a novel cyberattack detection system that incorporates client selection and transfer learning within a RIC and SAC-based P2P FL. The proposed method is well-suited for O-RAN to guarantee security, privacy, and low communication costs. The contributions of our paper can be summarized as follows:

1) We propose a novel RIC and SAC-based P2P FL for cyberattack detection where clients' peering undergoes a prior selection mechanism, in contrast to the conventional P2P FL where all clients are involved in training.
2) To benefit from P2P FL with client selection, we propose transfer learning between selected and discarded clients. While selected clients are directly involved in SAC, the remaining ones access the resultant global model via transfer learning. This approach not only curbs communication costs but also prioritizes top performers.
3) We provide a thorough performance evaluation of our approach in terms of accuracy and communication effort, and under different dataset distribution conditions.

The remaining of the paper is structured as follows. Section II describes the system model. Section III details the proposed cyberattack detection mechanism, followed by a presentation of our experimental results in Section IV. Finally, Section V concludes the paper.

II. System Model

We consider a 5G O-RAN architecture, where *RAN Intelligent Controllers* (RIC) are deployed to ensure the resource management tasks. In essence, RICs are logical controllers in O-RAN that comply with the 3rd Generation Partnership Project (3GPP) and Software-Defined Radio Access Network (SD-RAN) standards. The RICs include Near-real-time RIC (Near-RT RIC) and Non-real-time RIC (Non-RT RIC), which are based on Software-Defined Networks (SDN). They conduct radio resource management tasks, which may be realized with the help of AI/ML techniques [7], [8], [11].

Given that RICs can be operating in large-scale systems, besides being ML-based, it is interesting to consider cooperative mechanisms to benefit from each other's experience. To do so, *federated learning* mechanisms can be deployed among RICs. The primary purpose of FL is to maintain data privacy by enabling collaborative training of ML models on local datasets and sharing only model parameters. In particular, decentralized FL is an attractive approach since it eliminates the need for an aggregation server, hence alleviating the risks of a single point of failure and of global model alteration, corruption, slow convergence, or data misclassification. Furthermore, decentralized FL, such as SAC-based P2P FL, provides protection against semi-honest participants by securing the communication of model updates [7], [12], [13].

Secure average computation, as illustrated in Fig. 1, has initially been introduced in [13]. In essence, the roles of SAC input/output in distributed FL and of the aggregation server in centralized FL are the same, i.e., they average the participants' model updates into a global model. However, their transmissions within the FL framework differ. In particular, the SAC relies on two mechanisms, namely lightweight n-out-of-n secret partitioning and secure multi-party average calculation that leverages the partitioning method. In the secret partitioning method, each $Agent_j$ ($j \in \{1, \ldots, N\}$) generates N positive random numbers $\{rn_{j1}, \ldots, rn_{jN}\}$, which are then used to compute the percentage distributions, denoted as $prn_{j1}, \ldots, prn_{jN}$, such that:

$$prn_{ji} = \frac{rn_{ji}}{\sum_{k=1}^{N} rn_{jk}}, \quad (i,j) \in \{1, \ldots, N\}^2, \tag{1}$$

where N indicates the number of $Agents$ in the P2P FL environment. Subsequently, the percentage distributions are used to generate N partial weights $\{w_{j1}, \ldots, w_{jN}\}$ given by

$$w_{ji} = w_j \times prn_i, \quad (i,j) \in \{1, \ldots, N\}^2, \tag{2}$$

where w_j indicates the model update of $Agent_j$. For instance, $Agent_1$ has $w_1 = 30$ as the model update. This update is partitioned securely into $w_{11} = 4$, $w_{12} = 13$, and $w_{13} = 13$ as shown in Fig. 1. The resulting $\{wj_1, \ldots, w_{j_N}\}$ will be used in the second method, i.e., the multi-party average calculation. Specifically, each $Agent_j$ keeps its partial weights w_{jj} and shares the other parts w_{ji} with the respective $Agent_i$, $\forall i \in \{1, \ldots, N\}$ and $i \neq j$. To illustrate this mechanism in Fig. 1, $Agent_1$ keeps $w_{11} = 4$ and shares $w_{12} = 13$ with $Agent_2$ and $w_{13} = 13$ with $Agent_3$. Next, each $Agent_j$ computes its subtotal ps_j as

$$ps_j = \sum_{i=1}^{N} w_{ji}, \quad j \in \{1, \ldots, N\}. \tag{3}$$

Then, it computes the aggregated SAC weights S and the averaged weight Avg as follows:

$$S = \sum_{j=1}^{N} ps_j \text{ and } Avg = \frac{S}{N}. \tag{4}$$

III. Proposed SAC-based P2P FL-ASTL for Cyberattacks Detection

A. Description

We propose a communication-efficient SAC-based P2P FL framework that integrates agent selection and transfer learning, called SAC-based P2P FL-ASTL. Initially, client (or agent) selection for peering is realized based on the client's performance in each round. Then, selected clients participate in SAC to generate a global model. For efficient SAC peering, we select at each round only the agents that have a good

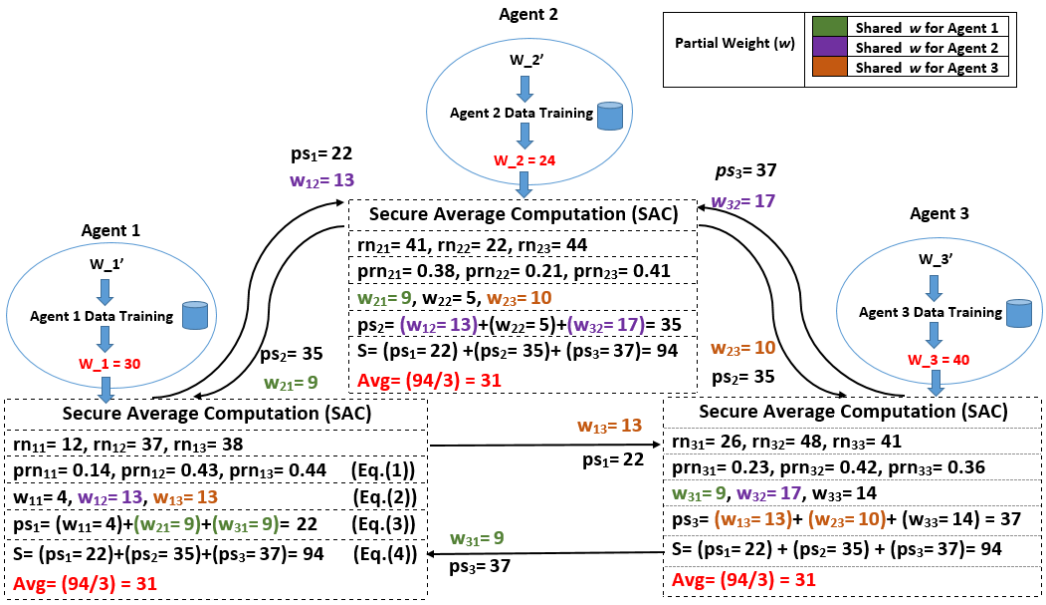

Fig. 1: Secure Average Computation (SAC) with three agents.

performance, in terms of accuracy, calculated based on their local validation dataset. Consequently, the global model is derived while ensuring the protection of both the model updates and performance through SAC. To strengthen the protection of datasets within the FL framework, each agent utilizes a distinct local validation dataset to generate the performance metrics, thus differently from the conventional FL.

In the literature, a trade-off between communication costs, computation costs, and privacy can be identified. For instance, encryption improves privacy but increases the computation cost due to encryption and decryption operations. In our approach, we aim to balance between communication costs and privacy. Specifically, we propose a novel methodology that reduces communication events within the large-scale SAC-based P2P FL, without compromising the privacy of both model updates and local datasets. The operation of our method operates based on two main steps, namely, "Initialization" and "Learning Process", which are iteratively executed for all FL rounds.

B. Operation

Let our system consists of N clients (e.g. agents) denoted as $\mathcal{A}_\mathcal{N} = \{A_1, \ldots, A_N\}$, where each agent A_i has a local training dataset D_i and a local validation dataset V_i from $\mathcal{D} = \{D_1, \ldots, D_N\}$ and $\mathcal{V} = \{V_1, \ldots, V_N\}$, respectively. Our method operates as follows:

1) **Initialization:** It involves multiple steps as summarized in lines 1 to 6 in Algorithm 1. First, in each round, agent A_i starts training from D_i to update its model weights w_i. Then, it uses its validation dataset V_i to generate its local performance metrics, namely F1-score, denoted $F1_i$ and accuracy Acc_i.
2) **Learning process:** This corresponds to the operations of lines 6 and 14 of Algo. 1.Specifically, it comprises three steps, namely agent selection, global model design, and transfer learning, as summarized in Algorithm 2. In the first phase (Algo. 2, lines [1-11]), each engages in SAC to securely exchange validation dataset metrics, i.e., F1-score and accuracy. Then, the average F1-score and accuracy are shared among all agents. To be selected for peering, each agent A_i verifies that it satisfies the conditions $F1_i \geq Avg_{F1}$ and $Acc_i \geq Avg_{acc}$. Subsequently, we have

$$\mathcal{S} = \{A_i \mid F1_i \geq Avg_{F1} \text{ and } Acc_i \geq Avg_{acc}\}. \quad (5)$$

Algorithm 1 Proposed SAC-based P2P FL-ASTL

Input: Number of agents N, training datasets $\mathcal{D}$, validation datasets $\mathcal{V}$, number of FL rounds T.

Initialization:

1: **for** $i = 1$ to N **do**
2: Get w_i after local training of D_i.
3: Test the updated model w_i on validation dataset V_i.
4: Get $F1_i$ and Acc_i
5: **end for**
6: Update $\mathcal{W} = \{w_1, \ldots, w_N\}$ using *Algorithm 2*

SAC-based P2P FL for subsequent FL rounds:

7: **while** $t \leq T$ **do**
8: **for** $i = 1$ to N **do**
9: Train local dataset D_i using W_{A_k} for ε episodes.
10: Get the updated model w_i'.
11: Test the updated model on validation dataset V_i.
12: Get $F1_i$ and Acc_i.
13: **end for**
14: Update $\mathcal{W} = \{w_1, \ldots, w_N\}$ using *Algorithm 2*
15: **end while**

Algorithm 2 Learning Process

Input: Number of agents $N, performancemetrics : [F1_i, Acc_i]_{i=1,\dots,N}$

Agent selection:

1: **function** SELECT($N, [F1_i, Acc_i]_{i=1,\dots,N}$)
2: Get $Avg_{F1} \leftarrow$ **SAC**$(N, [F1_i]_{i=1,\dots,N})$
3: Get $Avg_{acc} \leftarrow$ **SAC**$(N, [Acc_i]_{i=1,\dots,N})$
4: Initialize an empty set $\mathcal{S}$ for selected agents
5: **for** $i = 1$ to N **do**
6: Assign A_i to set $\mathcal{S}$ based on eq.(5)
7: **end for**
8: **return** $\mathcal{S}$
9: **end function**
10: Get $\mathcal{A}_\mathcal{K} = \{A_1, \dots, A_K\} \leftarrow \mathcal{S}$ % Set of selected agents
11: Get $\mathcal{A}_\mathcal{R} = \{A_i, \dots, A_R\} = \mathcal{A}_\mathcal{N} \backslash \mathcal{A}_\mathcal{K}$ % Set of disregarded agents

Global model design:

12: Get $W_{\mathcal{A}_\mathcal{K}} \leftarrow$ **SAC**$(\mathcal{A}_\mathcal{K}, K)$

Transfer learning:

13: Update $w_i, \forall i \in \mathcal{A}_\mathcal{R}$ using eq.(6)

Let $\mathcal{A}_\mathcal{K} = \{A_1, \dots, A_K\}$ and $\mathcal{A}_\mathcal{R} = \{A_i, \dots, A_R\}$ be the sets of selected and disregarded agents, respectively. Then, in the second phase (Algo. 2, line 12), selected agents participate in SAC to generate the global model W_{A_k}, following steps in (1)-(4). During the final phase, transfer learning for the global model W_{A_k} occurs, where the model is transferred to the disregarded agents of set $\mathcal{A}_\mathcal{R}$ such that

$$w_i = W_{\mathcal{A}_\mathcal{K}}, \forall A_i \in \mathcal{A}_\mathcal{R}, \tag{6}$$

where w_i refers to the learning model of agent A_i.

3) **SAC-based P2P FL-ASTL for subsequent rounds:** For the following FL rounds, the same initialization and learning processes are successively executed until the last FL round is reached. This is emphasized in lines 7-15 of Algo. 1.

C. Analysis of the Communication Effort:

In one hand, within the conventional SAC-based P2P FL, each agent shall transmit the computed partitions and subtotals of W model weight values to the other $(N-1)$ agents in each FL round [13]. Thus, the total communication effort per round can be computed as $2WN(N-1)$ and the total communication effort is given by

$$c = 2WN(N-1)T. \tag{7}$$

On the other hand, due to the designed learning process that involves agent selection, our proposed method requires less effort for training. Specifically, in a given round t, K_t agents engage in SAC. Consequently, the communication effort for training in round t is $c_{t,1} = 2WK_t(K_t - 1)$, $\forall t = 1, \dots, T$. Moreover, due to the SAC-based exchange of averaged F1 Avg_{F1} and Accuracy Avg_{acc}, a communication effort of $c_{t,2} = 2QN(N-1)$, where Q is the cost for averaging. Finally, due to transfer learning, a broadcast effort of $c_{t,3} = W$ is needed. Subsequently, the total communication effort of a single round for the proposed solution is $c_{t,1} + c_{t,2} + c_{t,3}$, and the total effort can be written as:

$$\begin{aligned} c' &= \sum_{t=1}^{T} (c_{t,1} + c_{t,2} + c_{t,3}) \\ &= \sum_{t=1}^{T} (2WK_t(K_t - 1) + 2QN(N-1) + W). \end{aligned} \tag{8}$$

Finally, for the conventional centralized FL [14], the communication effort can be evaluated as:

$$c'' = W(N+1)T, \tag{9}$$

since it requires N transmissions from the agents to the central aggregator to upload their model parameters and only a broadcast transmission from the aggregator to the agents to update their global models.

IV. Experimental Results

In this section, we evaluate the performances of our proposed FL method, denoted by "SAC-based P2P FL-ASTL", to detect cyberattacks in the network traffic in terms of accuracy and communication effort. We will compare our approach to the conventional "Centralized FL" [14], and decentralized "SAC-based P2P FL" [7].

A. Dataset Setup

For cyberattack detection, we have chosen the UNSW-NB15 dataset, which is a modern Network Intrusion Detection Systems (NIDS) dataset [15]. The UNSW-NB15 dataset has nine classes of attacks. Each attack category contains a set of records, and each record has 49 extracted features.

Pre-Processing Steps: We have considered a subset of the UNSW-NB15 dataset (2 data files out of the four available) and pre-processed it to be suitable for the purpose of cyberattack detection in O-RAN. Specifically, we merged the data of the two files and removed six non-relevant features, namely $\{srcip, dstip, attack\_cat, ct\_flw\_http\_mthd, is\_ftp\_login, ct\_ftp\_cmd\}$. The first three features have been eliminated for effective detection since, in real-world scenarios, network flows lack attack categories, and IP addresses may be dynamic or manipulated through IP spoofing. The other three features have been discarded due to their significant number of null values. As a result, each record has now 43 features. Furthermore, pre-processing steps such as splitting, feature/categorical encoding, and normalizing have been applied. The dataset has been partitioned into training, validation, and testing datasets. Each agent in the system has local training and validation datasets, and a final testing dataset to evaluate the global model.

Dataset Distribution: Following the pre-processing steps, the remaining dataset includes 150,000 records, to be distributed among $N = 100$ agents for training and validation, with an additional separate test dataset comprising 10,000

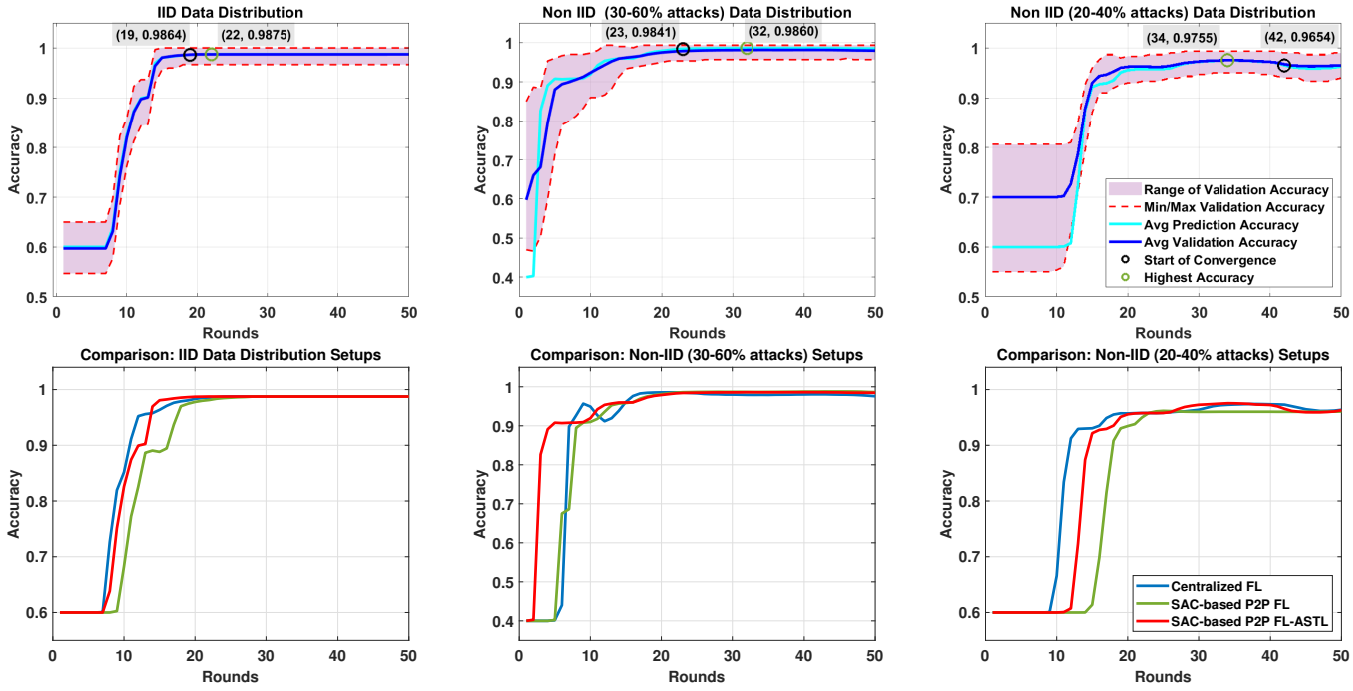

Fig. 2: Accuracy of: SAC-based P2P FL-ASTL (top row); Several FL methods (down row), with different data distributions.

records for global model testing. Within the records, 60% represent the attack classes. Each agent receives 1,500 records split $(80\%, 20\%)$ between training and local validation. The partition of the attack classes at each agent depends on the type of distribution, i.e., independently and identically distributed (IID), or non-IID. For IID data distribution, agents have an equal attack class partition of 60%. In contrast, for non-IID data distribution, the partition of the attack class was varied randomly within specific ranges of attack class partitions, in particular, we designed the range $[20\%, 40\%]$ for the intense non-IID setting and $[30\%, 60\%]$ for the moderate non-IID setting.

B. FL Model Architecture and Hyperparameters Selection

We adopt a deep learning (DL) architecture comprising four layers for our system. Specifically, the input layer is tailored to handle 42 features, followed by two dense hidden layers with 30 and 10 neurons, respectively. The architecture concludes with an output layer consisting of two neurons and is accompanied by a softmax layer for probabilistic classification between "attack" and "no attack". The ReLU activation function is applied to the hidden layers, supplemented by L1 regularization. Throughout our experiments, we set the learning rate to 10^{-4} and the batch size to 100. Also, we run FL for $T = 50$ rounds, where, in each round, an FL agent trains locally for $\varepsilon = 10$ episodes.

C. Results

In Table I, we evaluate and compare the communication effort results for the proposed "SAC-based P2P FL-ASTL", and the two benchmarks "Centralized FL" and "SAC-based P2P FL". We assume here an IID distribution of datasets, and that the FL model at any agent has $W = 1622$ weight values. According to Table I, the Centralized FL has the lowest communication effort, approximately 31.25 megabytes (MB), compared to the other methods. This is expected since it relies on a very low number of transmissions to/from the aggregation server. However, we notice that SAC-based P2P FL presents the highest communication effort, around 6127 MB, due to the necessary data exchanges among all $N = 100$ agents within the FL system. Finally, the proposed method demonstrates a communication effort reduction of 74% compared to SAC-based P2P FL, due to its support of agent selection and transfer learning. Even though Centralized FL achieves the lowest communication effort, it is prone to security breaches and failures due to its centralized architecture and lack of security mechanisms.

TABLE I: Communication Effort Results

	# model weight values	Avg. commun. effort per round (MB)	Total commun. effort (MB)
Centralized FL	$c'' = 163822$	0.625	31.25
SAC-based P2P FL	$c = 32115600$	122.55	6127.5
SAC-based P2P FL-ASTL	avg. $c' = 8068500$	31.92	1595.6

Fig. 2 depicts the global accuracy of our proposed approach (top row), then compared to the benchmarks (down row), for both IID and non-IID distributions. Looking at the top row of

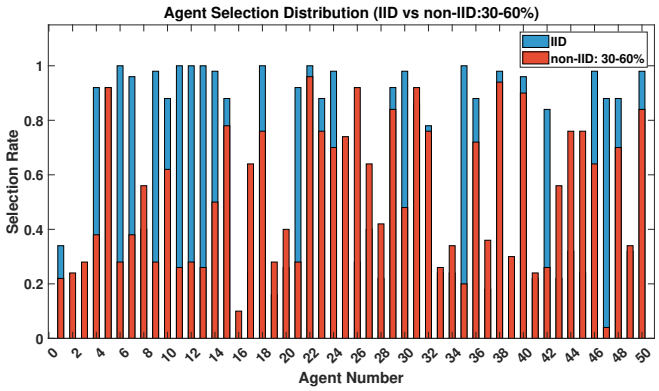

Fig. 3: Selection rate distribution for 50 agents, sampled from $N = 100$ agents (different IIDness settings).

Fig. 2, SAC-based P2P FL-ASTL converges after 19, 23, and 42 rounds and at accuracy values above 98.6%, 98.4%, and 96.5%, for the IID, moderate non-IID, and intense non-IID cases, respectively. Clearly, our approach is efficient in detecting cyberattacks. However, as non-IIDness increases, convergence slows down and accuracy negligibly degrades (between 0.3% and 2%), which demonstrates the robustness of our method in non-IID settings. This result is also in line with the increasing variance of the accuracy performance at convergence (area between red dashed lines).

When compared to the benchmarks in the down row of Fig. 2, we notice that all methods converge almost to the same accuracy values, meaning that they demonstrate similar robustness levels with respect to dataset non-IIDness. Nevertheless, our approach converges faster in the IID setting at round 19, while centralized FL and SAC-based P2P FL converge starting from rounds 23 and 25, respectively.

To understand the behavior of our proposed method, we plot in Fig. 3 the selection rates of 50 agents among the 100 available ones that perform FL, in both settings IID (blue) and non-IID (red). We notice that the selection rate varies significantly among agents, reflecting the system's adaptability in choosing agents based on their best F1 and accuracy performances. When an agent has a high selection rate, nearing 1, it indicates a higher likelihood of being selected over others in most rounds. The selection rate varies more in the non-IID setting compared to the IID one. This strategic selection in non-IID conditions highlights the model's robustness by maintaining performance and reducing the impact of unbalanced data distributions by adaptively modifying its dependence on different agents.

V. Conclusion

To bypass the single point of failure and security risks of Centralized FL, we propose here a novel SAC-based P2P FL-ASTL method adapted for use within the RICs of O-RAN network to detect cyberattacks. Unlike the conventional SAC-based P2P FL, we aim to reduce the communication effort through the integration of two mechanisms, namely agent selection and transfer learning. Agent selection has been developed in a secure manner where only agents presenting high performances, in terms of F1-score and accuracy, are allowed into P2P FL, while transfer learning ensures that all involved FL agents benefit from the SAC-based P2P FL. This approach enhances security in parameter sharing and reduces SAC's computational burden, especially with numerous trainers, paving the way for a more secure and streamlined O-RAN in 5G and beyond. Through experiments that used the modern UNSW-NB15 cyberattack dataset, we evaluated the performances of our approach in terms of communication effort and accuracy. The proposed SAC-based P2P FL-ASTL method succeeded in cutting the communication effort of the conventional SAC-based P2P FL by 74% while maintaining equivalent or higher accuracy performances than benchmarks. Finally, our approach has been proven robust against moderate and intense dataset non-IIDness, with a negligible degradation in accuracy (below 2%).

References

[1] K. Ramezanpour and J. Jagannath, "Intelligent zero trust architecture for 5G/6G networks: Principles, challenges, and the role of machine learning in the context of O-RAN," *Computer Networks*, vol. 217, p. 109358, Nov. 2022.

[2] M. Z. Chowdhury, M. Shahjalal, S. Ahmed, and Y. M. Jang, "6G wireless communication systems: Applications, requirements, technologies, challenges, and research directions," *IEEE Op. J. Commun. Soc.*, vol. 1, pp. 957–975, Jul. 2020.

[3] U. Ghafoor, M. Ali, H. Z. Khan, A. M. Siddiqui, and M. Naeem, "NOMA and future 5G & B5G wireless networks: A paradigm," *J. Network & Computer Appl.*, vol. 204, p. 103413, Aug. 2022.

[4] A. Afaq, N. Haider, M. Z. Baig, K. S. Khan, M. Imran, and I. Razzak, "Machine learning for 5G security: Architecture, recent advances, and challenges," *Ad Hoc Networks*, vol. 123, p. 102667, Dec. 2021.

[5] V. Sritapan, D. Massey, and B. Talbot, "5G security evaluation process investigation," *White paper*, May 2022. [Online]. Available: https://www.cisa.gov/sites/default/files/publications/5G_Security_Evaluation_Process_Investigation_508c.pdf

[6] M. Liyanage, A. Braeken, S. Shahabuddin, and P. Ranaweera, "Open RAN security: Challenges and opportunities," *Journal of Network and Computer Applications*, vol. 214, p. 103621, May 2023.

[7] D. Attanayaka, P. Porambage, M. Liyanage, and M. Ylianttila, "Peer-to-peer federated learning based anomaly detection for open radio access networks," in *Proc. IEEE Int. Conf. Commun.*, 2023, pp. 5464–5470.

[8] A. Abouaomar, A. Taik, A. Filali, and S. Cherkaoui, "Federated deep reinforcement learning for open RAN slicing in 6G networks," *IEEE Commun. Mag.*, vol. 61, no. 2, pp. 126–132, Feb. 2022.

[9] O. Alliance, "O-RAN-WG1-O-RAN architecture description v01.00.00," *Technical Specification*, 2020.

[10] N. Onoszko, G. Karlsson, O. Mogren, and E. L. Zec, "Decentralized federated learning of deep neural networks on non-IID data," *arXiv preprint arXiv:2107.08517*, Jul. 2021.

[11] S. Soltani, M. Shojafar, A. Brighente, M. Conti, and R. Tafazolli, "Poisoning bearer context migration in O-RAN 5G network," *IEEE Wireless Commun. Lett.*, vol. 12, no. 3, pp. 401–405, Mar. 2022.

[12] J. Le, D. Zhang, X. Lei, L. Jiao, K. Zeng, and X. Liao, "Privacy-preserving federated learning with malicious clients and honest-but-curious servers," *IEEE Trans. Info. Forensics & Sec.*, Jul. 2023.

[13] T. Wink and Z. Nochta, "An approach for peer-to-peer federated learning," in *Proc. Ann. IEEE/IFIP Int. Conf. Depend. Syst. Networks Wrkshps. (DSN-W)*. IEEE, 2021, pp. 150–157.

[14] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of FedAvg on Non-IID data," in *Int. Conf. Learn. Represent.*, 2020.

[15] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Proc. Military Commun. Info. Syst. Conf. (MilCIS)*. IEEE, 2015, pp. 1–6.