Federated Learning and Network Security: Foundations, Potential, and Resilience

Yann Busnel
Institut Mines-Télécom
Palaiseau, France
yann.busnel@imt.fr

Léo Lavaur
University of Luxembourg
Luxembourg
leo.lavaur@uni.lu

***Abstract*—Federated Learning (FL) is a distributed learning paradigm that enables training models across distributed clients without accessing their data. In the context of network security, FL can be used to collaboratively train Intrusion Detection System (IDS) models across multiple organizations, allowing participants to share knowledge without compromising data privacy. However, the distributed nature of FL raises new challenges, notably the heterogeneity of clients' data distributions and the identification of malicious contributions.**

This three-part tutorial introduces the audience to (i) the principles of FL, (ii) its application to network security, focusing on building Collaborative Intrusion Detection Systems (CIDSs) using FL, and (iii) the security challenges associated with deploying Federated Intrusion Detection System (FIDS), with a focus on poisoning attacks. Each part is illustrated with hands-on exercises, with step-by-step instructions provided in the companion material.

***Index Terms*—Federated Learning, Network Security, Collaborative Intrusion Detection System, Network Monitoring, Data poisoning.**

I. Introduction

Federated Learning (FL) has emerged as a promising paradigm for collaborative machine learning, enabling model training across decentralized data sources without compromising privacy. It quickly became prevalent in the distributed systems' community, with applications ranging from healthcare to fraud detection. In the context of network security, FL offers a compelling solution to the challenges of training Intrusion Detection System (IDS) models across multiple organizations, allowing participants to share knowledge without compromising data privacy. Yet, this use case comes with its own set of challenges, notably due to the heterogeneity of clients' data distributions.

A. Motivation

This tutorial delves into the fundamentals of FL, exploring its core principles and applications (Section II), with a focus on collaborative network security. The topic is particularly relevant for the distributed system community, as it echoes to a lot of the challenges faced in this field. This aims to establish a solid foundation for understanding the subsequent discussions on the applications of FL in collaborative network security and the associated security challenges.

B. Tutorial content

The tutorial is divided into three parts, each doubled with hands-on exercises to provide participants with practical insights into the complexities of collaborative network security using FL. Specifically, the tutorial covers the following.

(i) *Fundamentals of Federated Learning* (Section II), where participants are introduced to the core principles of FL and its applications.
(ii) *FL for Collaborative Network Security* (Section III), focusing on the application of FL to network security, and more specifically to the training of Collaborative Intrusion Detection System (CIDS) models.
(iii) *Security Challenges in FL* (Section IV), addressing the challenges of deploying and running Federated Intrusion Detection Systems (FIDSs), with a focus on poisoning attacks and mitigation strategies.

Hands-on: Through examples and hands-on exercises provided in the companion repository[1], the audience should gain insights into practical implementation of FL using Flower [1], an open-source Python framework. Hands-on activities involve constructing a basic CIDS model using Flower and experimenting with real-world network traffic datasets, providing participants with practical insights into tackling the complexities of collaborative network security using FL. Through interactive exercises, attendees can also simulate and analyze poisoning attacks on CIDS models, along with devising and testing mitigation strategies to safeguard against such threats.

Practical implementation: The tutorial is supported by PDF slides that notably include references to the literature for further reading, and Jupyter notebooks that contain step-by-step instructions and code snippets to fill. The Jupyter notebooks are also loadable in Google Colab, so that participants can run them directly in their browser without any installation, albeit with potential limitations on the available resources.

C. Paper content

The rest of this paper summarizes the tutorial content, starting with the fundamentals on FL in Section II, its application to network security in Section III, and the challenges associated with its security in Section IV.

[1]Available at: https://github.com/leolavaur/icdcs_2025

II. Fundamentals of Federated Learning

FL emerges at the intersection of collaborative computing and machine learning paradigms, offering a revolutionary approach to training models across a distributed network of devices without centralized data aggregation. Rooted in the concept of crowdsourcing, where large groups contribute or produce goods and services, FL extends this notion to machine learning, enabling diverse participants, from smartphones to IoT devices, to collaboratively improve model performance while preserving data privacy.

In its foundation, FL addresses the limitations of centralized machine learning by distributing the learning process across multiple nodes, each possessing local data and processing capabilities. This distributed approach not only enhances scalability to handle large datasets, but also mitigates privacy concerns associated with sharing sensitive data. Thus, the motivation for FL includes performance improvement with more data, meaningful combination of models and local training at node scale (and not only prediction at the edge). By allowing nodes to collaborate on model updates without sharing raw data, FL ensures privacy compliance with regulations such as HIPAA and GDPR, crucial in sensitive domains like healthcare and advertising.

A. FL in a Nutshell

At its core, unlike traditional centralized learning approaches, FL does not require training data to be uploaded to a central server. Instead, the latter generates an initial model that will be trained locally by the participants, as illustrated in Figure 1 for an IDS use case. Depending on the setting, the server selects the set or subset of clients that will be participating in the next round, either randomly or by following a specific strategy. The server then transmits the model weights to the selected clients, which will train the model on their local data. The clients then upload their new model weights to the server, which aggregates them to create the next global model. As this iterative process continues, the central server improves the global model's accuracy without ever accessing the data stored on the clients.

FL brings several advantages over traditional centralized machine learning approaches, but also introduces unique challenges. Unreliable clients and heterogeneous computing capabilities can cause higher latency, or even plain connection dropouts, particularly in cross-device settings. Its centralized process can be a single point of failure, hinder scalability, and pose sovereignty and privacy concerns. Moreover, due to its distributed nature, FL is more vulnerable to adversaries, whether they target model performance or data privacy. These challenges require robust solutions such as end-to-end encryption and secure aggregation to safeguard data integrity and confidentiality.

B. Different Approaches of FL

Two primary types of federation exist, tailored to different contexts and requirements: Cross-Device Federated Learning (CD-FL) and Cross-Silo Federated Learning (CS-FL).

Cross-device FL: In *cross-device* scenarios, the participating devices are typically diverse and widely distributed, encompassing a vast number of entities such as smartphones, IoT devices, and personal computers. These devices may range from thousands to billions, each possessing a relatively small dataset. Due to the heterogeneity of devices and their varying computational capabilities, cross-device FL often faces challenges like limited availability, reliability issues, and communication overhead. However, it offers scalability and adaptability, making it well-suited for scenarios where a large and diverse set of devices collaborates to train a single model for a third party.

Cross-silo FL: Conversely, *cross-silo* FL operates within organizational boundaries or distinct data silos, where each silo represents a separate entity. These silos could correspond to different departments within a company, independent organizations, or even geographical regions. Unlike CD-FL, which involves heterogeneous devices, CS-FL typically involves organizations with more uniform capabilities and larger datasets for training. Participants in cross-silo FL are generally more reliable and consistently available, as they are institutional entities with dedicated infrastructure and resources. However, these entities often exhibit significant differences in objectives, data distributions, and sometimes even model architectures. Cross-silo FL also provides greater control over data governance and security, as data sharing occurs within predefined organizational boundaries, providing easier compliance with regulatory requirements and privacy policies.

Beyond these, the conventional server-orchestrated approach is being challenged in the literature by alternative architectures aimed at eliminating this single point of failure. These include hierarchical aggregation mechanisms and fully decentralized settings, each offering unique benefits and constraints, though these are beyond the scope of this tutorial.

III. Federated Learning for Collaborative Network Security

Artificial Intelligence (AI) has become a cornerstone in modern network security, offering advanced capabilities in threat detection, alert correlation, and triage. Among its applications, AI-enhanced IDSs stand out in identifying security incidents within the massive amount of complex data generated by network infrastructures. Deep Learning (DL), in particular, has demonstrated its potential in improving IDS performance by uncovering intricate patterns and generalizing them to previously unseen attacks. However, the effectiveness of DL hinges on access to extensive (and often labeled) datasets, a requirement that is rarely met in practice.

FL emerges as a compelling solution to these challenges. By enabling organizations to collaboratively train IDSs models without sharing data, FL preserves data privacy while enabling collective intelligence. Traditionally, FL has been applied in cross-device scenarios, where a single entity aggregates knowledge from numerous devices. However, the context of CIDSs often involves multiple organizations, each with its own datasets and infrastructures, more akin to a cross-silo setting.

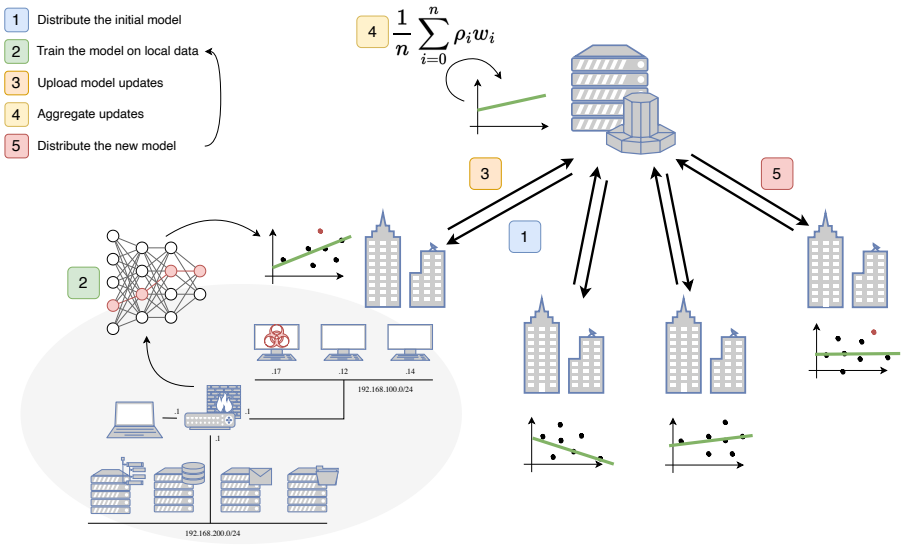

Figure 1. *FL process overview, applied to intrusion detection.* The server sends the initial model to the participating nodes (1), which train it locally on their collected traffic (2). The nodes then send their updates (3) to the server, which aggregates them to create a global model (4). The global model is then sent back to the nodes (5), and the process continues iteratively.

We refer to CIDS leveraging FL as FIDS, as they are federated across organizations.

Despite its promise, FIDSs face significant hurdles. The heterogeneity of data distributions across organizations poses a critical challenge, as variations in monitored traffic, deployed security solutions, and even model architectures complicate the aggregation process. The latter is particularly crucial, as it prevents the direct aggregation of models, a common practice in traditional FL approaches like `FedAvg` [2]. While adopting a shared model architecture can streamline FL strategies, it does not fully resolve the issuers introduced by data heterogeneity. Addressing these challenges remains a major area of research, as demonstrated by the existing literature [3].

Previous works [4] provided a comprehensive overview of the state-of-the-art of FIDSs, identifying clear research directions. These challenges span over three main axes:

(i) *Transferability, adaptability, and scalability.* While cross-silo is the most common setting for FIDS, on-device data collection for Endpoint Detection and Response (EDR) or localized IDS could also benefit from FL. Handling data heterogeneity and ensuring the transferability of models across different settings is already crucial in CS-FL, but it becomes even more difficult in these emerging cross-device settings. FL systems must also be able to adapt to the evolving nature of network traffic and threats, ensuring that models remain effective over time.

(ii) *Security, trust, and resilience.* FL inherently faces a wider attack surface than its centralized counterparts. As such, systematically understanding the various threats and their impact remains an open challenge, although some already work towards this direction [5], [6]. Mitigating these threats is crucial for ensuring the integrity and reliability of FIDSs, with an increasing need for adaptable reputation-aware systems [7].

(iii) *Local algorithm and aggregation performance.* The performance of FL systems is heavily influenced by the choice of local algorithms and aggregation techniques. As such, it cannot be decoupled from the latest Machine Learning (ML) and DL advancements for intrusion detection, where the semantics of the data and the model architecture could play a role in designing more appropriate aggregation algorithms. Finding the right balance between local and global model performance is key to fit the different use cases and settings.

IV. Security Challenges in Federated Learning

A. Threats against Federated Learning

Hinted in the previous section, the distributed and collaborative nature of FL introduces new security challenges, ranging from poisoning attacks to privacy breaches. The former target the global model's performance, while the latter aim at inferring sensitive information about the participants' training data. Because FIDS final goal is to ensure the security of the protected system, poisoning attacks are particularly relevant in this context. In fact, having adversaries manipulate the global model poses a significant threat, as they could theoretically use it to evade detection or cause alert fatigue.

Rodríguez-Barroso *et al.* [8] describe attacks against FL with four criteria:

(i) *Attack Moment*: attacks can be executed during the training or inference phase, depending on the objective.

(ii) *Attackers' Objective*: poisoning attacks, in particular, can target different subset of the data depending on the attacker's goal: a specific class for backdoors (*i.e.*, targeted

attacks), or the entire dataset for data pollution (*i.e.*, untargeted attacks).

(iii) *Poisoned Components*: the attacker can target the model weights, the training data, or the communication channel.

(iv) *Frequency*: attacks can be executed in a single round or over multiple rounds, depending on their impact and the attacker's strategy; *e.g.*, increasing the percentage of data over time to slowly divert the global model of its original local optimum.

B. Mitigation strategies

The community has proposed various strategies to mitigate adversarial attacks on FL systems and FIDS. These strategies primarily focus on developing robust aggregation algorithms and filtering mechanisms to counteract poisoning attacks. They can be categorized as follows:

Server-side evaluation: The server assesses the received contributions using a purpose-built representative dataset. However, this approach is often impractical in Non Identically or Independently Distributed (NIID) settings, as constructing a representative dataset would require access to the clients' data distribution.

Server-side model comparison: The server compares the received contributions either with a reference model or among themselves. In the former case, as with server-side evaluation, the lack of a single source of truth in NIID settings complicates this approach. By comparing models with each other, the server can identify discrepancies and detect potentially malicious contributions. This strategy is employed by methods like `FoolsGold` and `FLAME`, albeit with different objectives and methodologies.

Client-side evaluation: Clients evaluate the contributions of other participants using their local datasets and generate metrics. These metrics are then used to weight or filter models during aggregation. Notably, `radar` leverages these metrics as feedback for a reputation system and as input for a clustering algorithm. This approach eliminates the need for a single source of truth while ensuring high performance, even in heterogeneous settings.

References

[1] D. J. Beutel, T. Topal, A. Mathur, X. Qiu, T. Parcollet, and N. D. Lane, "Flower: A friendly federated learning research framework," 2020. arXiv: 2007.14390.

[2] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *20th intl conf. AIStat*, 2017.

[3] H. Zhu, J. Xu, S. Liu, and Y. Jin, "Federated learning on non-IID data: A survey," *Neurocomputing*, 2021.

[4] L. Lavaur, M.-O. Pahl, Y. Busnel, and F. Autrel, "The Evolution of Federated Learning-based Intrusion Detection and Mitigation: A Survey," *IEEE Transactions on Network and Service Management*, 2022.

[5] M. A. Merzouk, F. Cuppens, N. Boulahia-Cuppens, and R. Yaich, "Parameterizing poisoning attacks in federated learning-based intrusion detection," ser. ARES '23, 2023.

[6] L. Lavaur, Y. Busnel, and F. Autrel, "Investigating the impact of label-flipping attacks against federated learning for collaborative intrusion detection," *Computers & Security*, p. 104 462, 2025, ISSN: 0167-4048.

[7] L. Lavaur, P.-M. Lechevalier, Y. Busnel, R. Ludinard, G. Texier, and M.-O. Pahl, "Radar: Model quality assessment for reputation-aware collaborative federated learning," in *Proceedings of the 43rd International Symposium on Reliable Distributed Systems (SRDS)*, Sep. 2024.

[8] N. Rodríguez-Barroso, D. Jiménez-López, M. V. Luzón, F. Herrera, and E. Martínez-Cámara, "Survey on federated learning threats: Concepts, taxonomy on attacks and defences, experimental study and challenges," *Information Fusion*, 2023.

Yann Busnel is Senior Vice-President for Research at the Institut Mines-Télécom (IMT), France. With several years of experience within the IMT group, particularly at IMT Atlantique and IMT Nord Europe, he is responsible for developing and implementing the institution's scientific strategy. His work involves fostering inter-affiliate collaborations, overseeing doctoral programs, managing the PhD diploma of IMT, and coordinating the activities of its 11 scientific communities. He has over 15 years of experience in research and higher education. Between June 2023 and September 2024, he served as Director of Research and Innovation at IMT Nord Europe. Before that, he has held various academic positions and responsibilities in France and abroad: Full Professor at IMT Atlantique, Associate Professor at ENSAI in Rennes, Assistant Professor at the University of Nantes, and Researcher at Sapienza University of Rome.

His research topics are mainly related to Models for large-scale distributed systems and networks, with application (1) cybersecurity and dependability, to (2) the analysis of medical data, in the context of pharmacovigilance or genomic sequence analysis, and (3) the self-organized coordination of drone fleets. He has published over 100 articles in peer-reviewed journals and conferences. He has also coordinated several national and international collaborative research projects and is currently a member of the steering committee of the French national research group on networks and distributed systems (GDR RSD).

Léo Lavaur is a Researcher at the Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg. He received his Ph.D. in cybersecurity from IMT Atlantique, France, in 2024, where he focused on applying Federated Learning to Collaborative Intrusion Detection. During his Ph.D., he collaborated closely with the Chair on Cybersecurity in Critical Networked Infrastructures of IMT Atlantique, who partially funded his research. He holds an Engineering degree in Information Security from ENSIBS, Vannes, France, during which he also worked part-time at Orange Cyberdefense as a security analyst.

He explores various aspects of distributed system security, with a particular emphasis on collaborative knowledge sharing through machine learning. His research topics range from federated and decentralized learning to network security and intrusion detection, with a particular interest on reproducibility and the evaluation of distributed systems. He has published several papers on the application of Federated Learning to network security, focusing on the challenges of data heterogeneity and adversarial behaviors. His current research focuses on modeling causal dependencies in distributed microservice and GenAI architectures.