ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 13 (2024), e31647
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.31647

Federated Learning in Data Privacy and Security

Dokuru Trisha Reddy, Haripriya Nandigam, Sai Charan Indla and S. P. Raja

School of Computer Science and Engineering, Vellore Institute of Technology, Vellore Campus, Tiruvalam Rd, Katpadi, Vellore, Tamil Nadu 632014, India
✉ trishadokurureddy@gmail.com, kamaliharipriya@gmail.com, saicharan2872@gmail.com, avemariaraja@gmail.com

KEYWORDS

FL; federated averaging; differential privacy; FSVRG

ABSTRACT

Federated learning (FL) has been a rapidly growing topic in recent years. The biggest concern in federated learning is data privacy and cybersecurity. There are many algorithms that federated models have to work on to achieve greater efficiency, security, quality and effective learning. This paper focuses on algorithms such as, federated averaging algorithm, differential privacy, federated stochastic variance and reduced gradient (FSVRG). To achieve data privacy and security, this research paper presents the main data statistics with the help of graphs, visual images and design models. Later, data security in federated learning models is researched and case studies are presented to identify risks and possible solutions. Detecting security gaps is a challenge for many companies. This paper presents solutions for the identification of security-related issues which results in a decrease in time complexity and an increase in accuracy. This research sheds light on the topics of federated learning and data security.

1. Introduction

With substantial progress in federated learning (FL), the number of attacks is also increasing. Thus, with the aim of addressing this problem, this research mainly focuses on attacks related to FL models. To prevent attacks on FL models their complexity must be reduced and the number of security layers must be increased. Security layers in the network model first test the vulnerabilities introduced by the attack i.e., if a malicious program is installed in a computer by any spoofing websites/links, the network

layers such as the application layer, the physical layer etc., have to check and detect the attacks with the aim of finding a solution (Cheng et al., 2021). After verifying with the multi-layer models, the security checks the model interactions that the device communicates in the network communication layer. As the federated learning models are the models that train the other data handling devices/IoT devices, the federated servers have to be trained first. When the federated server is linked to a database model, for example a hospital database model linked to various network models which are further connected to a federated server. The federated server should send a data privacy layer to the other layers of network models. To solve the privacy concerns in this expensive communicative model, we are introducing another application layer into the server, it has been named the react layer. This application model first analyses the applications and then acts towards them. The proposed layer has been divided into three different sections and functionalities. Firstly, all the data that is stored in the server and the databases of the different IoT devices is connected to a federated learning server. The data is not be completely transferred to the global servers, instead, the raw data is generated for the local models and the local models are trained on the global models just like in the decentralized approach (Doku et al., 2019). In this approach, there is no direct contact with the central server, instead, transfers are made from the local models which are trained by the machine learning models. The end devices that we refer to as IoT devices store the data needed for this approach. Second, we shall update the local models and local models are sent to the federated server models which keeps the server secured and increases data privacy. In this way, the centralized model combines all the inputs given by the local servers and further combines to output as new knowledge. Since most of the data are received at a combined phase there are higher chances of it to be neutralized. Later, the server performs certain loops that raises the chances of privacy without any risk of vulnerability. At each loop, the risk management is decreased and more data is gathered. This is a suitable approach for increasing privacy. There are two different strategies and approaches in federated learning, namely, centralized and decentralized.

2. Literature Review

Cheng et al. (2021) proposed a SecureBoost system which is more efficient and capable than general federated learning methods. It can secure, expand and upgrade data better than other insecure data storing techniques. Doku et al. (2019) proposed an interlinking approach using both federated learning and blockchain technology to store data more distributedly and dynamically. Gosselin et al. (2022) used federated learning aggregation techniques, applications and topology structures to overcome attacks. The integration of blockchain technology with federated learning can overcome many other major threats. Jatain et al. (2022) discussed the different types of federated learning, including horizontal federated learning, vertical federated learning and federated transfer learning. Algorithms such as differential privacy with both local and global differential privacy methods are crucial for uplifting data privacy in FL. J. Jiang et al. (2020b) explained the federated averaging aggregation method by implementing it in smart cities. Jiang et al. (2020a) proposed FedDistill; a federated learning distributed algorithm which distills the data. By implementing the individualized approach on each device, the model helps to improve the local performance accuracy even in instances when the global model approach fails to fit to the local dataset, consequently enhancing the capacity and potential of the global model. Li et al. (2020) explained the primary goal of federated learning which promises

heterogeneous networks and decentralized data distribution. The authors also focused on the primary challenges and disadvantages, illustrated with examples and potential solutions. McMahan et al. (2017) proposed dynamic and non-IID data iterative averaging algorithm for machine learning model training. Mosaiyebzadeh et al. (2023) implemented an experimental set up of basic configuration in Collaboratory with Google by introducing panda's framework. The authors focused on solving security-based threats in federated learning such as communication issues, backdoor attacks and poisoning, which have also been discussed by Gosselin et al. (2022). Niknam et al. (2020) addressed fifth-generation networks involving federated learning for more feasible outcomes and its applications, including challenges and bottlenecks. Nilsson et al. (2018) compared the federating averaging algorithm, federated stochastic reduction gradient and CO-OP algorithms. FedAvg works more efficiently and with a higher accuracy rate. Wei et al. (2020) implemented noise before model aggregation FL (NbAFL) by using the differential privacy mechanism. The authors proposed the K-Client scheduling mechanism where K clients were selected from the group of total clients N and the aggregation process was performed. Yaacoub et al. (2023) introduced a EFCS system in federated learning to solve the credit silo problem in the most productive, efficient and secure way. Yu et al. (2022) mainly focused on the five main bottlenecks associated with training machine learning models with a federated approach, namely: cost in communication, heterogeneity of data and systems, privacy and security, other issues related to federated learning. K. Zhang et al. (2022) discussed differential privacy algorithms in regard to blockchain integration, homomorphic encryption, anomaly detection and secure multi-party computation. The desired privacy and security can be achieved by combining these mechanisms accordingly. IoT devices were studied by J. Zhang et al. (2022) in regard to security and privacy concerns in federated learning. Cutting-edge federated learning algorithms were discussed, and solutions were proposed along with their topologies, to enable their implementation in IoT devices.

2.1. Related Works on Federated Learning

Federated learning is a part of machine learning but it uses a decentralized approach. As discussed earlier, federated learning adapts this approach to provide greater data privacy. This approach helps to train the model with appropriate data and to conserve the data without actually saving or storing it. FL models help in securing private, financial, confidential or personal data. Although this approach alone does not accurately protect data, it can greatly contribute to its security. When we talk about federated learning, we mostly talk about data privacy, as it has been a topic of great concern in recent times. According to researchers, there are several algorithms that contribute to federated learning. Their experiments were intended to improve persistency and secure aggregation, wherein they found that there is no way to detect an aggregation that indulges in a malicious program or to detect who presented the aggregation. Below, we detail their experiments one by one:

- Mc Mahan et al. (2017) tested an image verifying model. The model is sent image data with a certain pixel value and the other attacker's image data with no pixels, although the images were the same and had the same colors. The authors tested their proposed model to see if the backdoor model's accuracy would match the original one. The results of the test showed that there was a slight accuracy redundancy and that there was a change compared to the original. More training time was required but the pixel patterned approach was effective. By taking experiments from the federated learning literature, the model could learn accuracy with compromising on that

particular state where the model is. There were certain features on the data input images that were classified from the backdoor working model, such as some special objects or some colors that made the differentiating factor.

- Mc Mahan et al. (2017) carried out a word prediction experiment. In the experiment, the attacker chose a sentence which ended with a word that is the main target of the sentence. There are various applications of federated learning in the fields of building databases, mobile applications and web applications. These applications have vulnerabilities due to the fact that there are malicious attacks that challenge their security. These malicious links are circulated to the models and are very hard to detect. Thus, primarily focusing on malicious link detections, deep learning and pattern learning approaches were used, including machine learning and other data learning approaches. The URL was extracted along with its length and features. According to the survey, further on, choosing the optimal algorithm and algorithm design would help. There have also been some case studies which established links between federated learning and hyper networking models, developing an image processing model in the form of an AI for more accurate classification. In these case studies, the hyper networking model was optimized to work well with the federated models. There are some methods to tackle the challenges regarding privacy. These models minimize the amount of data that reaches the model and make the server complex so that it does not extract the output information related to the training samples.

Now, we shall describe a series of case studies so as to list the applications related to federated learning and data privacy sectors:

i. Keyboard by Google (Gboard): Google uses the federated learning concept in its keyboard without compromising its privacy. Federated learning is used to improve suggestions while typing which also offers personalized suggestions. Initially, typing suggestions raised privacy concerns and to address this, Google took a step forward in building Gboard using federated learning. It was found that federated learning increased its data privacy concerns and it was implemented by training it on local servers, on several end devices/user devices. Gboard acts effectively as it does not send the users' data to the main google servers and it only encrypts the models and sends them back. In doing this, the privacy is preserved and the private data still remains on the user's device and only the local servers that aggregate models are regularly updated. This was one of the greatest innovations in the field of federated learning and sets a good example for data privacy. Although next word prediction initially seem to entail major drawbacks for data privacy, researchers have been able to prove that although it is not completely private, its accuracy is at least 70 %. Furthermore, Google has proven this with a series of case studies. The server's architecture is basically the database management that works with different layers and models. In the above figure, federated learning helps improve the models by training it on end devices. And thanks to large data sets, the training data enhances the model's ability to provide suggestions. The training tasks and inferences maintain the data as training data. The model update transfer in between the client and server is to predict the autocorrection while a user is typing/using Gboard. First, as we can see on the client's side, it maintains the user's profile and their different behavioural patterns. It takes care of the themes and layouts of the keyboard. All these data are stored in the user's data interface/server itself. The database architecture keeps improving and evolving with its new features and themes. On the server side, the cloud server helps to provide cloud services for several features. For example, features such as translation and speech to text conversion.

ii. Microsoft's Inner Eye: This Is one of the research projects of Microsoft. Federated learning is mostly used for medical image analysis. It is used in scanning cells and radiations. The use of ML can help to increase the accuracy of medical images and thus heighten the efficiency of treatment. The data is stored privately by a local server. The project mainly focused on developing AI for detection in scans such as MRI or CT. Using AI algorithms, customized plans are generated for image processing and analysis techniques. For analyzing image and other tasks this research also used deep learning along with federated and machine learning. Federated models are trained across many institutions and hospital databases, to ensure clinically relevant information. But this process also helps to improve data privacy and accuracy. All the tools are combined and are integrated with the InnerEye tool. These integrated tools are further made to combine with all healthcare tools present. Inner Eye has been mainly launched for clinical purposes and is AI-assisted.

Table 1. Federated learning applications and their uses (Niknam et al., 2020)

S. No	Industry	Use Cases
1	Healthcare	Medical image analysis, patient data research.
2	Finance	Credit scoring, fraud detection, risk assessment.
3	IoT	Smart devices, sensor data analytics.
4	Retail	Demand forecasting, personalized recommendations.
5	Manufacturing	Quality control of data and predictive maintenance.

2.2. Review of Previous Work

Table 2. Comparative review.

S.No	Study	Deployment	Methodology	Results	Strengths	Weakness
1	Cheng et al. (2021)	Secure and lossless FL	secure multi-party computation (MPC), Gradient boosting decision trees (GBDTs)	High accuracy and efficiency in federated patterns; scalable throughout different datasets and settings	High efficiency, strong security with MPC, scalability	Limited to GBDTs, potential overhead with MPC
2	Gosselin et al. (2022)	Security and privacy in FL	Overview of privacy-preserving techniques and security challenges	Comprehensive review of privacy techniques like differential privacy and homomorphic encryption; identification of ongoing challenges	Thorough overview of privacy techniques, highlights future research directions	No empirical results, primarily a review study

(*continued*)

Table 2. Comparative review. (*continued*)

S.No	Study	Deployment	Methodology	Results	Strengths	Weakness
3	Mothukuri et al. (2021)	Security and privacy in FL	Overall examination of security and privacy methods	In detail review of various security methods such as differential privacy and secure aggregation, focusing existing challenges	Extensive coverage of security techniques, identification of key challenges	No empirical results, primarily a review
4	Wei et al. (2020)	Differential privacy	Introduction of algorithms including differential privacy into FL	Illustrated that differential privacy can be efficiently utilized into FL without significant loss of accuracy	Enhanced privacy, minimal impact on accuracy	Potential computational overhead with differential privacy
5	Zhang et al. (2022)	Security and privacy risks in FL	Study of security and privacy risks, suggested methods to eradicate these risks	Identified vital security and privacy threats, proposed reduction strategies	Focused on security and privacy, practical recommendations	Limited empirical validation, primarily theoretical analysis
6	McMahan et al. (2017)	Communication efficiency in FL	Federated Averaging (FedAvg) algorithm	Significant reduction in communication-n overhead while retaining high model accuracy	Communication efficiency, maintained model accuracy, scalability	Limited to early FL scenarios, further validation required
7	Jiang et al. (2020a)	Improved FL algorithm	Implemented knowledge distillation for reducing communication-n overhead and improvising model performance ability	Better model performance and reduced communication costs through knowledge distillation	Performance improvement, communication efficiency	Limited exploration of broader application scenarios
8	Jiang et al. (2020b)	FL in smart city applications	Analysis of FL challenges and opportunities in smart city sensing	Applicability in traffic monitoring and environmental sensing; identification of data heterogeneity and computational challenges	Practical insights for smart city applications, highlights specific challenges	Limited to smart city scenarios, no new FL algorithms proposed

Table 2. Comparative review. (continued)

S.No	Study	Deployment	Methodology	Results	Strengths	Weakness
9	Mosaiyebzadeh et al. (2023)	FL-based security for IoHT devices	Applied FL to intrusion detection systems (IDS) for IoHT	Improved security and high detection accuracy with minimal data sharing	Enhanced IoHT security, high detection accuracy, efficiency	Limited to IoHT applications, broader applicability not explored
10	Niknam et al. (2020)	FL in wireless communication-s	Analysis of the potential of FL in wireless communication systems	Highlighted opportunities for FL in enhancing data privacy and reducing latency in wireless networks	Practical insights for wireless communications, identification of key opportunities	Limited empirical validation, primarily theoretical analysis
11	Yaacoub et al. (2023)	Security of FL in IoT systems	Analysis of security issues specific to FL in IoT environments	Identified key security threats and proposed potential solutions for enhancing FL security in IoT	Focused on IoT, practical recommendations for enhancing security	Limited empirical validation, primarily theoretical discussion
12	Yang et al. (2023)	Explicitness and security in FL	Integration of explainable FL with integrating blockchain for secure credit modelling	Enhanced security and explainability in credit modelling applications through blockchain and FL integration	Enhanced security, explainability, practical application in credit modelling	Specific to credit modelling, broader applicability not explored
13	Konečný et al. (2016)	Fundamental FL principles	Introduction of key FL ideas and optimization techniques	Reduced communication costs and improved model convergence in distributed settings	Foundational framework, practical applicability	Early-stage concepts, subsequent research required for refinement
14	Jatain et al. (2022)	Taxonomy, threats, vulnerabilities, and challenges in FL	Specific taxonomy and risk, threats examination	Detailed classification of FL aspects, evaluation of potential threats, identification of key challenges	Detailed taxonomy, thorough threat assessment	Theoretical focus, lacks practical implementation details
14	Nilsson et al. (2018)	Performance evaluation of FL implied algorithms	Concrete evaluation of various FL algorithms	Performance comparison across various datasets, highlighting strengths and weaknesses of different algorithms	Empirical evaluation, performance comparison	Limited scope of algorithms evaluated; further validation needed

3. Motivation and Justification

The intention of writing this paper is to showcase the existing methodologies in the field of data security and federated learning. Following the review of numerous papers, we found that there are only a few papers that have been published on this collaborative topic. Improving the data security of federated learning is the sole purpose of this paper. Federated learning papers were reviewed separately from data security papers and this research attempts to combine both aspects. The techniques and algorithms used here are as follows: 1) federated averaging and 2) differential privacy along with examples. The growth and improvement of federated learning have been mentioned in the reviewed research. Various data security topics have been spotlighted, along with a series of models. Motivated by this, an attempt has been made to combine federated learning with data security.

3.1. Contributions

1. The paper highlights key challenges concerning federated learning. It presents a thorough overview of risks and vulnerabilities with mitigation techniques, assessing possible solutions.
2. This paper explores various applications of federated learning in wireless communications and other fields, highlighting its benefits and drawbacks through case studies and performance evaluations.
3. The paper overviews the future research advancements and possibilities of federated learning. It highlights the need of continuous developments in federated learning techniques to overcome present challenges and ensure data privacy and security.

3.2. Challenges

In present life, data security plays an important role in maintaining the privacy of our data. Many organizations are growing day by day and the protection of sensitive and personal information is crucial for them. There are many aspects to be addressed in the context of data security; some of the key issues are:

a) Privacy of data: FL in data security mainly focuses on the privacy of personal and sensitive data. As this data travels through the many servers and devices of organizations, the data has a high risk of leakage and unauthorized users may hack our data and use it for illegal activities. They may even sell our valuable data to others. It is very necessary to build a strong server or mechanism which protects one's privacy and secures data from hackers or unauthorized access.
b) Security in Communication: The communication of data between two people/participants is very important. This communication of data can be attacked by hackers and the valuable communicated data can be stolen. There should be a proper authentication of communication and securing. It is needed for preventing unauthorized users from accessing and manipulating the data.
c) Verification of Model and its Accuracy: The model of the server is to protect it from unauthorized users. Models are tested for accuracy and it is verified whether they are robust against manipulation. There are many models aimed at ensuring data security in federated learning through the detection of accurate information.
d) Confidentiality of Data: The data should be confidential and should not leak to other unauthorized users which may occur in data breach attacks. It is vital to prevent the loss of confidential data from hackers and data leakages.
e) Data Attacks: There are some attacks where user data might be attacked by a virus or be poisoned with some data breaching attacks which leads to the loss of user data. We need to implement data

security models to prevent data loss. Preventing the attacks is compulsory if users' trust in the data security federated learning process is to be maintained. There are a series of key aspects involved in maintaining the data security of a user to prevent manipulation attacks from hackers.

Federated Learning (FL) offers the benefits of AI to domains with sensitive data and heterogeneity. FL acts by decentralizing data from the central server to end-devices and preserving user privacy. This paradigm emerged primarily as a result of two factors: the lack of sufficient data to reside centrally on the server-side (as opposed to traditional machine learning) due to direct access restrictions on such data; and data privacy protections using local data from edge devices, i.e., clients, rather than sending sensitive data to the server, where network asynchronous communication comes into play. Maintaining data privacy allows for the efficient application of AI benefits afforded by machine learning models across many domains. Furthermore, rather than relying on a single entity, computational power is distributed among interested parties. Overall, FL offers a decentralized and privacy-preserving machine learning technique that fosters cooperation while protecting data privacy. FL provides a potential solution for training machine learning models on sensitive and dispersed data sources by solving data privacy issues and putting in place strong security safeguards.

3.3. Descriptions

FL allows for the training of models using decentralized sources of data without the necessity of data centralization. The training process happens locally on member devices or servers in federated learning, as opposed to transmitting unprocessed information to a centralized server or the cloud. Figure 1 explains the steps involved in training the devices on FL (Jiang et al., 2020b).

Step 1: Initializing and setting up the process- The main server, which is also known as the central server, initializes the structure of federated learning. Devices are employed in the process as edge devices. The model is initially derived from the central server and is involved in the system process.

Step 2: Distributing the data- The model that is initially derived from the central sever is provided for access by the participating edge devices. The data is collected, gathered and stored, following data privacy and security mechanisms.

Step 3: Training the data on local base- The model is independently trained with its own edge device data. Various optimization techniques are performed and the model is trained locally. This process of training is performed recursively.

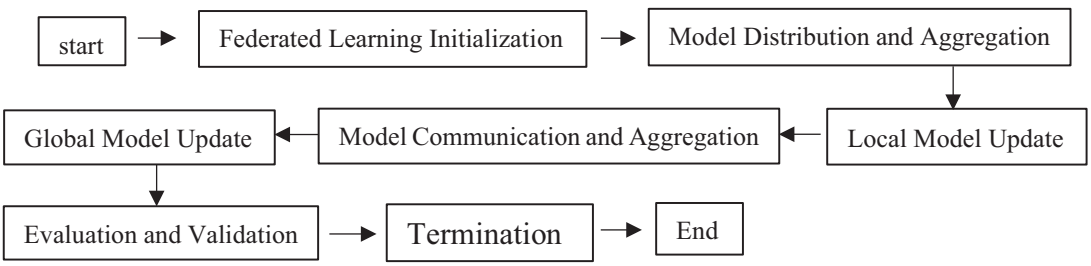

Figure 1. Stages of federated learning

Step 4: Updating and aggregating the model- Once the data has been trained on a local level, the devices do not send their original complete data to the server. Instead, they send only the required updates and the variants that happened in between. The central server is responsible for assembling the updates of the model from all the existing devices. Then the process of aggregation starts in which the model updates are aggregated with various algorithms and techniques. The final model is deployed with help of this trained and aggregated model, adjusting its constraints and understanding the data received from all the participating devices.

Step 5: Recursively repeating steps 2 to 4. The updated final model is transmitted back to the devices. Step 2 to step 4 is recursively repeated to obtain the required criteria for completing the process.

Step 6: The process concludes with a trained model- When the criteria are fulfilled by the model, the process of federated learning ends. The final model is the result of the conclusion of the process. Deployment takes place once the server takes in the final model.

The following is a more in-depth explanation of federated learning:

a) Data decentralization and localized instruction: Concerns regarding data privacy are addressed through federated learning, which stores data on personal computers or servers. This lowers the likelihood of data disclosure or hacks occurring. Member devices or servers carry out training for models using their own local data in order to protect the privacy and confidentiality of user information.
b) Model aggregation: Following the completion of neighborhood instruction, model updates are transmitted to a centralized server, also known as an aggregator. There, the updates are integrated with other local models to produce a global model as shown in Figure 2 (Niknam et al., 2020).
c) Techniques for protection of personal privacy: Differential privacy is a technique that may be included in federated learning. This approach involves the addition of regulated chaos for model updates, which protects users' privacy while still allowing for the extraction of valuable insights.
d) Distributed average: Federated averaging is a typical method used in federated training that helps maintain security by combining model updates in a way that safeguards individual data contributions. Federated learning is an approach to machine learning in which several computers work together to solve problems.
e) Edge computing: Federated learning takes advantage of edge computing infrastructure, which enables the training of models directly on edge devices. This helps to reduce latency and minimize dependency on cloud-based resources. Edge computing is also known as fog computing.
f) Broadband efficiency: Federated learning minimizes the necessity for transferring large volumes of raw data across the network, hence minimizing the requirements for bandwidth and the expenses connected with it.
g) Efficiency in computational work: Federated learning may exploit parallel processing capabilities, which allows it to be scalable and efficient. This is accomplished by spreading model training among participant devices or servers.
h) Resilience in the face of centralized data breaches: Reduced attractiveness of possible targets to attackers. Federated learning reduces the danger of just one point of malfunction or a centralized.

- Different Aspects of FL

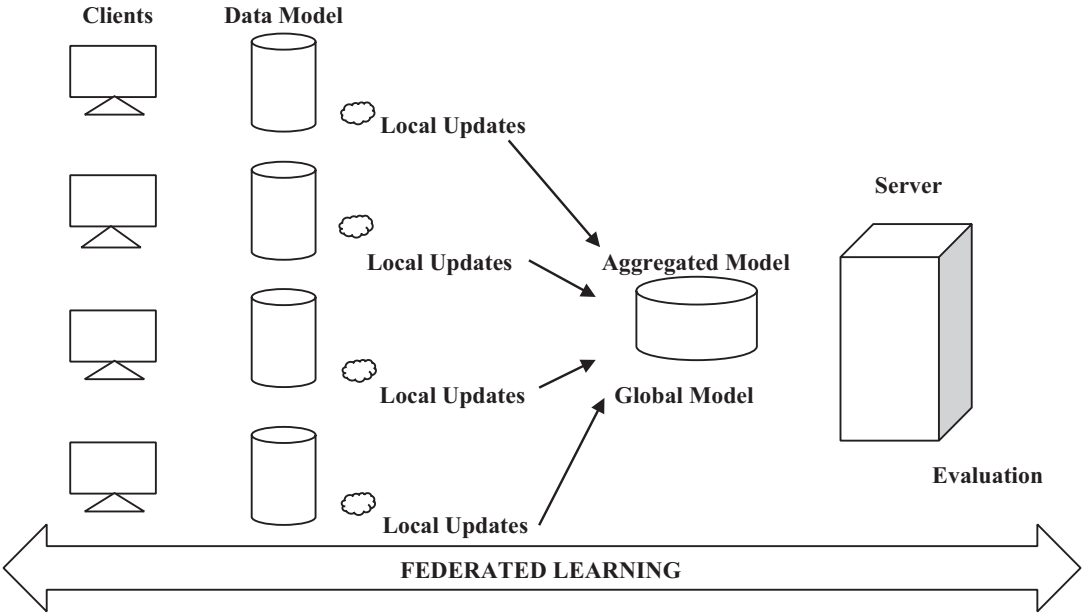

Figure 2. Client Server architecture for federated learning (Niknam et al., 2020)

Table 3. FL factors, advantages and their challenges (Jatain et al., 2022; K. Zhang et al., 2022)

S. No	Aspect	Advantages	Challenges
1	Data Privacy	Preserves individual data privacy.	Complex coordination for model updates.
2	Scalability	Scales efficiently for distributed data.	Network latency and the overhead of the communication.
3	Data Diversity	Incorporates diverse data for better models.	Heterogeneous data quality and distribution.
4	Data security	Local data remains on devices, reducing risks.	Potential for malicious attacks on local models.
5	Efficiency in data	Reduces data transfer, saving bandwidth	Convergence

4. Federated Learning Methods for Data Security

- Homomorphic Encryption: This method enables calculations on encrypted data without having to first decode it. In FL, participants may securely encrypt and submit their local model changes to the aggregator, which can then process the encrypted models. This minimizes the chance of data leaking by ensuring that the sensitive data is encrypted throughout the procedure.
- Secure Multi-Party Computation or FL Averaging: This method enables numerous participants to compute a function together while maintaining the confidentiality of their inputs. In FL,

model aggregation may be done without disclosing the specific model updates by using secure multi-party computing. Without having access to the original data, the aggregator can safely calculate the average or other aggregation functions.

- Differential Privacy: To safeguard individual privacy, differential privacy adds noise to the data as a privacy-preserving approach. In FL, differential privacy can be used for the local model changes prior to transmission, preventing the aggregated model from revealing private information about specific players. Secure aggregation techniques are designed to safeguard the aggregation process in general. These protocols use cryptographic methods to protect participant identities while ensuring that the aggregator receives accurate model updates from participants. To safeguard the aggregated model's confidentiality and integrity, a number of secure aggregation procedures, including secure sum and secure averaging, can be utilized.

4.1. Protocols for Secure Communication in FL

- Establishing Secure Communication Channels: Participants in FL must create secure communication channels via which to send model changes. To guarantee that the connection is encrypted, authenticated, and resistant to eavesdropping or man-in-the-middle attacks, secure channel setup uses encryption and authentication protocols, such as Transport Layer Security (TLS). Participants submit their model changes to the aggregator during FL in a secure manner. The model changes can be encrypted and sent securely over the network using secure transmission protocols such as Secure Socket Layer (SSL) or Secure File Transfer Protocol (SFTP), preventing unauthorized access or interception.
- Secure Model Aggregation: The aggregator executes the aggregation procedure after receiving model changes from participants. Cryptographic methods and integrity checks can be used to confirm the legitimacy and integrity of the received updates, reducing the possibility of tampering or malicious manipulation, and ensuring safe model aggregation. Mechanisms for FL authentication guarantee that only authorized parties participate in the federated learning process, in these mechanisms participants in FL must be validated. The participants' identities and reliability can be verified using methods such as public-key infrastructure (PKI), digital certificates, or secure tokens.
- Trusted Aggregator Authentication: The aggregator in FL plays a crucial role in ensuring the security of the data. It is possible to use authentication measures to confirm the legitimacy and identity of the aggregator, guaranteeing that only reliable aggregators participate in the federated learning process. For this, methods such as mutual authentication, digital signatures, or secure protocols can be applied.
- Robust Model Poisoning Detection: To alter the resultant model, model poisoning attacks entail inserting malicious data in the federated learning process. Model poisoning attacks may be detected and mitigated using a variety of anomaly detection techniques, strong aggregation algorithms, or outlier detection approaches, preserving the integrity and dependability of the federated model. Model inversion attacks aim to obtain sensitive data from the federated model. To defend against them, techniques such as model regularization, input perturbation, or model obfuscation can be used to fight against such assaults to make it more difficult for adversaries to extract sensitive information from the model (Wei et al., 2020). Attacks on membership inference are preventable. These attacks seek to ascertain if a certain data record was utilised in the training process.

- Preservation of Labelling of Data: Data labelling which protects privacy can, in reality, create a privacy risk in federated learning. This is because it exposes private information to other parties. To guarantee the secrecy of the labelled data, privacy-preserving data labelling methods can be used, such as secure crowdsourcing or cryptographic labelling protocols. Participants can provide labelled data using these strategies without disclosing the real data or jeopardising individual privacy.
- Hybrid techniques: Hybrid techniques increase data security by fusing federated learning with other privacy-enhancing technologies. Federated learning, for instance, can be used in conjunction with trusted execution environments (TEEs) or secure enclaves to offer hardware-based security for the computation and storage of model updates. By doing this, it is ensured that sensitive data is shielded from unauthorised access and that model changes are done safely. To defend against membership inference attacks and retain the privacy of individual participants, preventive measures can be used, including privacy preserving techniques such as differential privacy, information restriction approaches, or adaptive sampling.

4.2. Differential Privacy Algorithm

Differential privacy introduces noise to the computation results to provide privacy guarantees. The formula for differentially private mechanisms is (McMahan et al., 2017):

$$P\,(Q(D)\,\epsilon\,S) \leq e\text{^}\varepsilon * P(Q(D')\,\epsilon\,S) + \delta, \tag{1}$$

Where:

P(Q(D) ϵ S) represents the probability of the output of the query Q on the dataset D falling within the set S. ε is the privacy budget controlling the amount of noise added to the computation. δ is the privacy parameter providing an upper bound on the probability of any event not related to the dataset affecting the output.

4.3. Federated Averaging Algorithm

McMahan federated averaging algorithm is implemented by considering a federated learning network where stochastic gradient descent (SGD) is used to update the modifications during optimization (McMahan et al., 2017). Recently, various deep learning and machine learning algorithms have been widely optimized through SGD gradient. While implementing at each communication round 't' using the SGD, we chose a propagation set of client's 'X' to optimize the process and calculate the gradient of loss with respect to the local model. So, X= 1 which is the global step size, and the algorithm is referred to as federated SGD. Each client 'i' has a learning rate constant eta 'η'. The local model data average is optimized as $a_i = \nabla F_i(\omega_t)$. Where 'a' is known to be the difference of average of local model and global model and 's' is the step - size. The local model of client 'i' is updated to $\omega_{t+1} \leftarrow \omega_t - \eta \sum_{I=1}^{I} \frac{S_i}{S} a_i$ from the result $\eta \sum_{I=1}^{I} \frac{S_i}{S} a_i = \nabla f(\omega_t)$. Similarly, for each client that is on the next round it is updated as, $\omega_{t+1}^{i} \leftarrow \omega_t - \eta a_t$. Which further can be written as, $\omega_{t+1} \leftarrow \eta \sum_{I=1}^{I} \frac{S_i}{S} \omega_{t+1}^{i}$.

The local updating process starts once the client increments gradually and communicates with the server. The federated averaging algorithm derives after the client communicates with each participant client 'i' through iterating and optimizing $\omega^i\ \omega^i - \eta\ \nabla F_i(\omega^i)$ number of times. The learning rate is

denoted as ∇L, which is basically $\frac{\partial L}{\partial \omega}$. In solving the algorithm, we consider X=1, Y=1 the number of passes it takes to train the local data set, same as federated SGD and R which is noted to be infinity as it is marked as single set. The updates locally are derived as $u_i = Y\frac{n_i}{R}$ (McMahan et al., 2017).

Algorithm 1: Federated Averaging – Averaging technique to aggregate the model updates from client to server.

Server Side:
1 Initialize ω_0,
2 for t = 1, 2, 3, …..
3 for i ϵ **set of maximum clients of X. I,**
4 $\omega_{t+1}^{i} \leftarrow$ **Update (i,** ω_t)
5 $\omega_{t+1} \leftarrow \eta \sum_{I=1}^{I} \frac{n_i}{n} \omega_{t+1}^{i}$

Client side:
6 for each local update from 1 to Y do
7 for batch r ϵ **R do**
8 $\omega \leftarrow \omega - \eta \frac{\partial L}{\partial \omega}(\omega; r)$
9 Communicate ω **to the server**

4.4. Federated Stochastic Variance Reduced Gradient (FSVRG)

The FSVRG algorithm introduces one centrally complete gradient computation and then concentrates on updating various decentralized updates. The step-size 'h' is used as the main objective for a typical FSVRG.

Algorithm 2: FSVRG

Assume client 'i' the local step-size of the client 'i' is 'h_i**'.** $h_i = \frac{h}{n_i}$**, where** n_i **is the local size of the data with data partition '**P_i**'. M, N are the matrices of the client. The one complete gradient computation at central level is** $\nabla F(\omega_t)$**, where '**ω_t**' is the current updating model (Konečný et al., 2016).**
1 initialize ω_0
2 for each client communication round, do
3 Evaluate $\nabla F(\omega_t) = \frac{1}{n}\sum_{x=1}^{n} \nabla F_x(\omega_t)$
4 for i = 1 to I Clients parallel do
5 Initialize $\omega_i = \omega_t$ **and** $h_i = \frac{h}{n_i}$
6 Assume $\{x_t\}_{t=1}^{n_i} \in P_i$
7 for t = 1 to n_i **do**

$$8\ \omega_i = \omega_t + h_i(N_i\,[\nabla F_{xt}(\omega_t) - \nabla F_{xt}(\omega_t)] + \nabla F(\omega_t))$$

$$9\ \ \omega_t = \omega_t + M\sum_{I=1}^{I}\frac{n_i}{n}(\omega_i - \omega_t)$$

10 End

As we know, federated learning works on the principle of data security, with improvement mechanisms as follows (Li et al., 2020):

Table 4. Privacy mechanisms in existing algorithms

S No	Federated Learning (FL) Privacy Mechanisms	Algorithm	Server Characteristics
1	FL without any extra privacy related mechanism	$\Delta\omega$ = aggregate. ($\Delta\omega_1 + \Delta\omega_2 + \Delta\omega_3$)	Server
2	FL with Global Privacy	$\Delta\omega$ = μ (aggregate. ($\Delta\omega_1 + \Delta\omega_2 + \Delta\omega_3$))	Trusted and Dependable server
3	FL with Local Privacy	$\Delta\omega'$ = aggregate. (μ ($\Delta\omega_1$) + μ ($\Delta\ \omega_2$) + μ ($\Delta\omega_3$))	Malicious server

- Privacy preserving Techniques

Table 5. Privacy algorithms and functions (Gosselin et al., 2022)

S No	Privacy technique	Description
1	Differential Privacy	Adds random noise to data or query responses to protect individual privacy while allowing statistical analysis.
2	Homomorphic Encryption	Enables performing computations on encrypted data without decrypting it, ensuring privacy during model training.
3	Secure Multi-Party Computation	Allows multiple parties to jointly compute a function on their private inputs without revealing them to each other.
4	Secure Aggregation	Aggregates model updates without exposing individual data by using cryptographic protocols like secure sum or mean.

4.4.1. Examples in Federated Averaging Algorithm

Imagine we have a simple scenario where three clients (Client A, Client B, and Client C) are participating in a federated learning process to train a model that predicts whether an email is spam or not. Each client has their own dataset of emails.

1. Local Training:

- Client A trains a model on their data and gets the parameters:
  ΩA= [0.3,0.4]
- Client B trains a model on their data and gets the parameters:
  ΩB= [0.5,0.1]

- Client C trains a model on their data and gets the parameters:
  ΩC= [0.6,0.6]

2. Sending to Server:
   - Each client sends their model parameters to the central server.

3. Averaging on Server:
   - The server computes the average of these parameters to get the global model. The averaging is done element-wise:

     Ωglobal= 3{ΩA + ΩB + ΩC}
     Ωglobal= 3{[0.3,0.4] + [0.5,0.1] + [0.6,0.6]}
     Ωglobal= [0.5,0.4]

4. Back to Clients:
   - The server sends the averaged parameters back to the clients.

5. Local Update:
   - Each client updates their local model with the new global parameters and continues training on their local data.

This process repeats for a set of rounds until the model converges or meets the desired performance criteria. The key advantage of federated averaging is that it allows for collaborative training without sharing the actual data, thus preserving privacy. It is particularly useful when working with sensitive information such as personal emails, medical records, or financial transactions.

4.5. Growth of Federated Learning

Federated learning is executed through the trained model's updates and variations rather than sharing all of the data. This property of federated learning increases the security of the users' data. This is what initially attracted researchers to federated learning, gradually leading to the emergence of more advanced techniques or algorithms to improve the good performance of applications. Google Gboard was initially introduced by a google organization which encrypts federated learning (Yu et al., 2022). We can say that Google was the first company to bring the federated learning theory to application. Gboard users, with variant and multiple smartphones, may share their contents without compromising their privacy. Federated learning has continued to evolve since its emergence. Federated learning is employed with various optimizing techniques to preserve privacy. In fact, helping build a more secure learning to not invade user's privacy and provide them with a safe and trusted environment is the main criterion for federated learning. The differential privacy technique used in federated learning has become more popular since the Apple company proposed it. In essence, this technique produces noise when the data is shared to ensure user privacy and data security. This works in letting the person know when the data is shared through the noise. Nevertheless, differential privacy could not eliminate the privacy concern completely. Data leakage is not completely prevented but it is helpful to some extent. Traditional learning methods, such as normally built machine learning applications, are not associated with this sound ejecting technology due to various model characteristics. Secure multi-party computation (MPC) allows multiple and variant clients to together function their updates without providing access to the original individual information. Software Guard Extensions (SGX) was introduced by

intel sources to provide honest and secure extension facilities in the federated learning process. Software Guard Extensions (SGX) is a technique that is applied on data that is encrypted. It works on the encrypted data by maintaining data security. The evaluated federated learning models are further put into action through the secured hardware encryption techniques produced by Intel secure. Microsoft Research team introduced an encrypting method that applies to a homomorphic encryption method for data privacy in federated learning known as Simple Encrypted Arithmetic Library (SEAL). By applying SEAL in federated learning, the homomorphic encryption gets stronger, therefore creating better encryption techniques to secure data privacy.

The mathematical formulae of the federated learning according to the researchers is,

$$f(\mathrm{a},\ldots\ldots\ldots,\mathrm{a}) = \frac{1}{\mathrm{n}}\sum_{i=1}^{n} 1\, f_{\mathrm{i}}(\mathrm{X}_{\mathrm{i}}) \tag{2}$$

There was tremendous research in the topics of federated learning and it has increased in the years of 2016 and 2017. The topic that was first researched was about its averaging algorithms and its communication standards. Later on, studies focused a lot on topics such as strategies from datasets and to reduce iterations in the techniques. The robustness of the model and algorithms was underscored. Some of the efficiency related learning approaches were meant to comprehend the deep learning strategies which help the models in processing. Federated learning and deep learning vary significantly in data privacy and security. Deep learning focusses on a centralized approach to collect and store data, which has a risk of data breaches and unauthorized access leading to privacy and security issues. On the other hand, federated learning solely shares the model gradients and keeps the data on local devices by following decentralized approach and distributed learning. Distributed learning is a multi-node comprising training model that improves upon the large amount of data which increases the scalability, and trains the model updates (Jiang et al., 2020b).

5. Performance Evaluation

5.1. Overview

The performance of a federated learning model was monitored and evaluated based on the MNIST dataset (LeCun et al., 1998), which is a reference dataset for handwritten digit identification. The FL systems was evaluated in terms of security, privacy, scalability and efficiency. The MNIST dataset encompasses 70,000 grayscale images of 28x28 pixels each, with 10,000 test images and 60,000 training images.

5.2. Performance Metrics

The following metrics were chosen to evaluate the performance of the federated learning model:

1. Data Confidentiality: Evaluates the system's capability to maintain the raw data within the user device and avoid reaching the central server using a measurement marked as «High», which determines resilient data protection
2. Anonymization: Indicates if the identity of the users is anonymous. «True» results in full anonymization.
3. Privacy: Measured through differential privacy variable. Strong data privacy is implied with low value.
4. Rigorous to Attacks: Evaluates the robustness of the system against various attacks such as data poisoning with a certain percentage under an attack.

5. Data Aggregation: Uses encryption in the process of aggregating system updates, while maintaining security of data throughout the transmission.
6. Authentication, Authorization and Integrity: Indicates if only authorised devices are allowed in the FL approach.
7. Model Reliability: The accuracy percentage of digits in the federated learning model, compared to the centralized approach.
8. Communication Efficiency: The efficiency percentage in rate of data transmission compared to the centralized approach.
9. System Scalability: The maximum number of devices that can be assisted by the federated learning system without any drop in performance.
10. System Latency: The average time to link the system updates with the central server.
11. Resource Limitations: Evaluates the memory and processing allocations in the user devices.
12. Regulatory Adherence: Represents the system's compliance to data security guidelines.
13. Accountability and Audit: Outlines the limit of logging and tracking for integrity and accountability.

5.3. Experimental Setup

The MNIST dataset was dispersed among 1000 virtual devices. The devices trained their data locally and model updates were sent to the central server and aggregated to develop the global model. Differential privacy algorithm was deployed to maintain data privacy. The main objective of this experimental setup is to simulate a federated learning model with the device and network settings.

6. Experimental Results

Table 6. Performance evaluation metrics and experimental results of federated learning on the MNIST dataset (LeCun et al., 1998)

S. No	Metrics	Value
1	Confidentiality	Strong
2	Anonymization	True
3	Privacy	$\varepsilon = 1.0$
4	Rigorous to attacks	90 %
5	Aggregation of Data	Encrypted
6	Authentication, Authorization and Integrity	True
7	Model Reliability	95 %
8	Communication Efficacy	75 %
9	System Scalability	1000 nodes capacity
10	System Latency	100ms
11	Resource Limitations	Average
12	Regulatory Adherence	Complaint (GDPR/HIPAA)
13	Verification and Evaluation	Complete

6.1. Analysis

The experimental results display the federated learning system's capacity to support high accuracy and privacy with system scalability and efficiency. Data confidentiality is secured by implementing the differential privacy and secure aggregation algorithms. The system is rigorous to attacks, maintaining consistent performance even in challenging situations. Efficiency in communication and low limitations in resources results, the federated learning system is ideal for large-scale applications, particularly when there is shortage of resources.

7. Conclusion

In this paper, we discussed various data security and privacy concerns and examined the types of attacks on federated learning applications. Through the review of a range of activities and models, this paper contributes to the knowledge on preserving and securing the data. We have seen major things happening across the fields of healthcare, IoT, robotics and others. This paper has covered the main aspects of FL, from applications to threats. We have discussed some of the major real-world applications that have been built by Google, Microsoft and Apple. They contribute a lot to the society and most of their aspects focus on ensuring privacy and security. Privacy and security are major aspects of federated learning and they have driven the growth of this particular field. We have pointed to the algorithms that make a significant change in the federated learning mechanisms and bring solutions to data privacy concerns. The algorithms that work in FL are essentially different from those used in decentralized approaches such as blockchain technologies. The homomorphic techniques used in FL encrypt the data using mathematical functions such as secret key. These techniques are useful in securing the data, ensuring the parameters and the data cannot be decrypted. Yet, the original encrypting functional techniques are far more efficient than the homomorphic approaches. The future work and the research challenges mainly regard privacy. Only if models are highly secure will it be possible to achieve the desired accuracy through training with different types of data sets. The latest research trend relates to federated learning which helps the models to constantly develop complexities and accuracies. Federated learning will play an important role in the fields of healthcare, smart homes, smart infrastructures, robotics, AI and many more. For now, federated learning has had a big impact on healthcare and is likely to continue on that path. The use of ML, and particularly FL, will mark a turning point in research aided diagnosis.

References

Cheng, K., Fan, T., Jin, Y., Liu, Y., Chen, T., Papadopoulos, D., & Yang, Q. (2021). Secureboost: A lossless federated learning framework. *IEEE Intelligent Systems*, *36*(6), 87-98.

Doku, R., Rawat, D. B., & Liu, C. (2019, July). Towards federated learning approach to determine data relevance in big data. In *2019 IEEE 20th international conference on information reuse and integration for data science (IRI)* (pp. 184-192). IEEE.

Gosselin, R., Vieu, L., Loukil, F., & Benoit, A. (2022). Privacy and security in federated learning: A survey. *Applied Sciences*, *12*(19), 9901.

Jatain, D., Singh, V., & Dahiya, N. (2022). A contemplative perspective on federated machine learning: Taxonomy, threats & vulnerability assessment and challenges. *Journal of King Saud University-Computer and Information Sciences*, *34*(9), 6681-6698.

Jiang, J. C., Kantarci, B., Oktug, S., & Soyata, T. (2020b). Federated learning in smart city sensing: Challenges and opportunities. *Sensors*, *20*(21), 6230.

Jiang, D., Shan, C., & Zhang, Z. (2020a, October). Federated learning algorithm based on knowledge distillation. In *2020 International Conference on Artificial Intelligence and Computer Engineering (ICAICE)* (pp. 163-167). IEEE.

Konečný, J., McMahan, H. B., Ramage, D., & Richtárik, P. (2016). Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527*.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). Federated learning: Challenges, methods, and future directions. *IEEE signal processing magazine*, *37*(3), 50-60.

McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017, April). Communicationefficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics* (pp. 1273-1282). PMLR.

Mosaiyebzadeh, F., Pouriyeh, S., Parizi, R. M., Han, M., & Batista, D. M. (2023, May). Intrusion Detection System for IoHT Devices using Federated Learning. In *IEEE INFOCOM 2023-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)* (pp. 1-6). IEEE.

Mothukuri, V., Parizi, R. M., Pouriyeh, S., Huang, Y., Dehghantanha, A., & Srivastava, G. (2021). A survey on security and privacy of federated learning. *Future Generation Computer Systems*, *115*, 619-640.

Niknam, S., Dhillon, H. S., & Reed, J. H. (2020). Federated learning for wireless communications: Motivation, opportunities, and challenges. *IEEE Communications Magazine*, *58*(6), 46-51.

Nilsson, A., Smith, S., Ulm, G., Gustavsson, E., & Jirstrand, M. (2018, December). A performance evaluation of federated learning algorithms. In *Proceedings of the second workshop on distributed infrastructures for deep learning* (pp. 1-8).

Wei, K., Li, J., Ding, M., Ma, C., Yang, H. H., Farokhi, F., ... & Poor, H. V. (2020). Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security*, *15*, 3454-3469.

Yaacoub, J. P. A., Noura, H. N., & Salman, O. (2023). Security of federated learning with IoT systems: Issues, limitations, challenges, and solutions. *Internet of Things and Cyber-Physical Systems*, *3*, 155-179.

Yang, F., Abedin, M. Z., & Hajek, P. (2023). An Explainable Federated Learning and Blockchain based Secure Credit Modeling Method. *European Journal of Operational Research*.

Yu, B., Mao, W., Lv, Y., Zhang, C., & Xie, Y. (2022). A survey on federated learning in data mining. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *12*(1), e1443.

Zhang, K., Song, X., Zhang, C., & Yu, S. (2022). Challenges and future directions of secure federated learning: a survey. *Frontiers of computer science*, *16*, 1-8.

Zhang, J., Zhu, H., Wang, F., Zhao, J., Xu, Q., & Li, H. (2022). Security and privacy threats to federated learning: Issues, methods, and challenges. *Security and Communication Networks*, *2022*.