International Journal of Software Science and Computational Intelligence
Volume 15 • Issue 1

A Novel Deep Federated Learning-Based Model to Enhance Privacy in Critical Infrastructure Systems

Akash Sharma, Chandigarh College of Engineering and Technology, Chandigarh, India
https://orcid.org/0000-0002-5879-7353

Sunil K. Singh, Chandigarh College of Engineering and Technology, Chandigarh, India*
https://orcid.org/0000-0003-4876-7190

Anureet Chhabra, Chandigarh College of Engineering and Technology, Chandigarh, India

Sudhakar Kumar, Chandigarh College of Engineering and Technology, Chandigarh, India

Varsha Arya, Department of Business Administration, Asia University, Taiwan & Department of Electrical and Computer Engineering, Lebanese American University, Beirut, Lebanon & Center for Interdisciplinary Research, University of Petroleum and Energy Studies (UPES), Dehradun, India

Massoud Moslehpour, Department of Business Administration, Asia University, Taiwan & California State University, San Bernardino, USA
https://orcid.org/0000-0001-8808-2407

ABSTRACT

Deep learning (DL) can provide critical infrastructure operators with valuable insights and predictive capabilities to help them make more informed decisions, improving system's robustness. However, training DL models requires large amounts of data, which can be costly to store in a centralized manner. Storing large amounts of sensitive critical infrastructure data in the cloud can pose significant security risks. Federated learning (FL) allows several clients to share learning data and train ML models. Unlike centralized models, FL does not require the sharing of client data. A novel framework is presented to train a VGG16 based CNN global model without sharing the data and only updating the local models among clients using federated averaging. For experimentation, MNIST dataset is used. The framework achieves high accuracy and keep data private using FL in critical infrastructures. The benefits and challenges of FL along with security vulnerabilities and attacks have been discussed along with the defenses that can be used to mitigate these attacks.

KEYWORDS

Cyber Attacks, Cyber Security, Deep Learning, Federated Learning, Machine Learning, Malicious server, Privacy Protection

DOI: 10.4018/IJSSCI.334711

*Corresponding Author

1. INTRODUCTION

Critical infrastructures are systems which are essential for the smooth functioning of a society and its economy and these include such as transportation systems, communication networks, and financial systems. These infrastructures are often complex, interdependent, and vulnerable to disruptions, which can have serious repercussions for public safety, economic stability, and national security. Deep Learning (DL) (Aggarwal et al., 2022; Mengi et al., 2023) provides critical infrastructure operators with valuable insights and predictive capabilities that can help them make more informed decisions, improve system resilience, and enhance public safety. By analyzing data from sensors and other sources, DL models can identify patterns and anomalies that may indicate equipment failure or maintenance needs so that it could be repaired before failures occur, reducing downtime (Mandle et al., 2022). Deep learning in critical infrastructure requires large amounts of training data for accurate and reliable modeling of the task. Traditionally, training data for these systems has been collected in data centers or on a single machine, which can be costly and time-consuming. Storing large amounts of critical infrastructure data in the cloud can pose significant risks and responsibilities, such as the potential for data breaches and cyber attacks. To address these challenges, training data from these systems must be collected and managed, in a decentralized manner. These approaches could help ensure the accuracy and quality of the training data while also reducing the risks and responsibilities associated with storing large amounts of sensitive data in the cloud. Federated Learning (FL) (D. Li et al., 2023) can be used to secure critical infrastructure making it possible to achieve the benefits of improved efficiency and performance while utilizing DL without compromising the safety and security of the system. Enhancing privacy(M. Singh et al., 2023) in critical infrastructure systems is crucial for safeguarding sensitive information and ensuring the reliability of essential services. Developing different prediction algorithms for critical infrastructure system (I. Singh, Singh, Singh, et al., 2022) (Peñalvo, Maan, et al., 2022) (S. Gupta et al., 2023) with sustainable development (Chopra et al., 2022; Peñalvo, Sharma, et al., 2022) (Bouncken et al., 2022; M. Singh et al., 2023) is an art which involves a deep understanding of the underlying systems, the potential risks they face, and a creative approach to designing algorithms with minimum overheads (S. Kumar et al., 2021) (S. Kumar et al., 2022) (P. S. Kumar, 2022; S. Kumar et al., 2023).

Federated deep learning is a technique and architecture which allows multiple clients to interact and train a deep learning model without having to share their raw data with each other. In federated learning, the training data is distributed across multiple clients or devices and the model is trained locally on each client using its own data. The updates from the local models are then aggregated to create a global model that is more accurate and robust. In this approach, since the data remains on the local devices or servers, and only the model updates are exchanged between the clients or with a central server. Therefore, federated deep learning helps to preserve the privacy and security (A. Sharma et al., 2023; I. Singh, Singh, Kumar, et al., 2022) of sensitive data and reduces the risk of data breaches and cyber attacks. In this paper a novel framework using federated deep learning for critical infrastructure has been proposed in which multiple devices or clients collaboratively train a global model without sharing their raw data. The global model is a convolutional neural network (CNN) (Kaur et al., 2021) that is trained initially on a centralized dataset. The framework is demonstrated on the MNIST dataset, a commonly used benchmark dataset for image classification.

In this paper, the authors briefly review all the important components of federated learning in privacy protection in section 2 they described previous contributions and use cases of federated learning in cyber security. In section 3 of this paper, authors described motivation to use federated learning in privacy protection and simultaneously achieving security and privacy in FL. Section 4 of the paper introduces a secure FL based deep learning framework for critical infrastructure. In the 5th section of this paper the authors elaborated security vulnerabilities and attacks in the FL domain. Continuing these attacks from section 6, they described defenses against these attacks. Since FL

is a new and recent technology, section 7 throws light on limitations of federated learning and the application of FL are discussed in section 8.

2. LITERARY WORK

While there has been some research on this subject, progress has been achieved; nonetheless, basic research in some areas lags behind when it comes to understanding FL and its security and privacy implications. This study differs from previous efforts in that it aims to provide a comprehensive analysis of FL security in terms of a formal introduction, dangers, countermeasures or defenses (Pan et al., 2022), applications, and issues.

FL might be used to create highly high-end ML based, multi-domain, real-time applications by letting users keep their data private. FL Models are commonly applied in a variety of fields, including smart cities(Chopra et al., 2021; R. Singh, Singh, Kumar, et al., 2022), education, finance (Lee & Suh, 2022; Marinakis & White, 2022), edge technology (Saini et al., 2020; R. Singh, Singh, Kumar, et al., 2022), healthcare and insurance, and various other intelligent applications(Khade et al., 2012). This section goes over a few notable use cases that have been deployed in real-time utilizing FL technology (Alazab et al., 2022).

For monitoring risk in small and micro firm loans, most banks traditionally employed a process known as whitelisting. Whitelisting is a screening technique that employs some rules and requires user involvement. For this, data should be obtained from banks that hold all credit reports. Encryption utilizing the RSA encryption technique is currently utilized when conveying sensitive information. This process is likewise restricted to approved agents and banks. To eliminate the need for manual intervention, machine learning (ML) and artificial intelligence (AI) models must be developed (A. Gupta et al., 2022a). However, owing to the sensitivity of the data, acquiring initial data on which the ML model is formed, as well as testing, is one of the most difficult issues. Previously, a Chinese bank called WeBank attempted to overcome the issue by deploying an AI-based risk-control strategy. However, owing to cyber-attacks, sending sensitive data via the Internet was a huge difficulty. To address this issue, FredeRick Ctrl, a FL-based risk control system for SME loan applications, has been introduced. The FL-AI Technology Enabler platform is used to implement the application. The application is based on the heterogeneous FL idea. When compared to a traditional ML strategy, Hetero-LR improves prediction accuracy by roughly 12% during the screening step.

Anti-money laundering efforts are typically seen as critical in the banking sector. Based on traditional methods, if a transaction must be determined as money laundering activity, rule-based models are used to filter the right records, requiring a significant amount of manual intervention to review and classify any transactions of money laundering which is a time-consuming process. Though most banks utilized ML algorithms on a regular basis to determine these sorts of transactions, the performance is not as high as expected due to the lack of training data. Multiple banks must be connected to collect large amounts of data, but this is impractical since banks are reluctant to reveal their private information. Even if they do share, the attackers may misunderstand the data, causing the action to be misunderstood. As a result, an online bank of China constructed a platform employing FL for virtually merging the data of all banks and generated a model called Homo-LR to train the data and to avoid cyber-attacks. Each client bank trained the model locally using the global model without publicly disclosing the data, and then global model was updated based on the training values of each individual client with the latest parameters. This notion aided in resolving the data island problem without jeopardizing data privacy. Due to its usage, the no. of transactions to be reviewed manually decreased from thousands to less than 50.

3. FEDERATED LEARNING AND PRIVACY

3.1 Motivations to Use Federated Deep Learning for Security of Critical Infrastructure

The traditional approach to cybersecurity (Bhardwaj & Kaushik, 2022) makes it more difficult to acquire and share data in a privacy-invading manner. Similarly, data aggregation from several data providers is a difficult task. FL might be used to reduce cyberattacks while also achieving data privacy and security (Arafeh et al., 2022). The paper (Jalali & Chen, 2023) discusses security vulnerabilities in federated learning under IoT critical infrastructure and proposes security models and algorithms as solutions. The paper (Rathod et al., 2023) proposes an AI and onion routing-based secure architecture for IoT-enabled critical infrastructure to address security risks and protect user privacy, confidentiality, and integrity. Figure 1 displays the many elements that impact the usage of FL for cybersecurity, as well as the strategies employed by FL to attain these advantages. The steps followed to simulate federated learning are shown in figure 2. The following are the reasons for using FL for cybersecurity (Alazab et al., 2022).

- **Data privacy:** Critical infrastructure systems are often highly sensitive and proprietary, and their data cannot be shared openly. Federated learning enables organizations to train models collaboratively without sharing their data, thus maintaining data privacy and confidentiality.
- **Distributed computation:** Federated learning distributes the computation workload among multiple parties, reducing the burden on any one organization. This can be particularly useful for critical infrastructure systems that require real-time response times.
- **Enhanced security:** Federated learning can improve the security of critical infrastructure systems by reducing the attack surface area. By keeping data and models decentralized, hackers have fewer points of attack, making it more difficult to compromise the system.
- **Improved accuracy:** Federated learning can also improve the accuracy of models by using more diverse data from multiple sources. This can help detect and prevent attacks more effectively.
- **Faster model deployment:** With federated learning, models can be trained more quickly, as multiple parties are collaborating on the training process. This can help organizations respond more quickly to new threats and vulnerabilities.
- **Flexibility**: Federated learning can be applied to a wide range of critical infrastructure systems, including power grids, transportation systems, and industrial control systems.

3.2 Simultaneously Achieving Security and Privacy in FL for Critical Infrastructures

Ensuring the security and privacy of critical infrastructure while minimizing computational costs and accuracy loss in Federated Learning (FL) poses a significant challenge. The privacy of clients must be safeguarded without compromising the accuracy of learned models or adding complexity to the network while ensuring strict privacy assurances. Techniques used for privacy protection should not result in unacceptably high overhead on the network or complexity in training.

While a multiparty solution that aggregates local updates using significant encryption before applying them to the global model can enhance privacy, it can be costly and obscure individual changes from the FL server. As a result, the server cannot compute accuracy metrics and weight statistics on individual updates, which may result in rejected updates. To ensure consistency with security protection, any plan to protect clients from privacy attacks should avoid aggregating updates. Combining encryption, differential privacy, secure aggregation, access control, anonymization, and model verification techniques can ensure both security and privacy in FL.

To enhance privacy, an alternative approach is to sever the relationship between updates and their creators, for instance, through anonymous communications. Encrypted communications such as SSL/TLS alone may not be adequate, as the FL server could be considered a potential adversary

despite securing communications against third parties. Additionally, the removal of identifiers from the communication payload may not be sufficient to ensure privacy, as metadata such as IP addresses or timestamps can reveal personal information about individuals and be used by the server to track specific, identifiable individuals. By using anonymous communication channels that make it impossible for contact parties to understand the metadata of message messages, unlinkability between sender-messages can be achieved. In conclusion, implementing these techniques can provide a secure and privacy-preserving approach to collaborative machine learning in critical infrastructure.

Figure 1. Motivation to use federated learning in cybersecurity

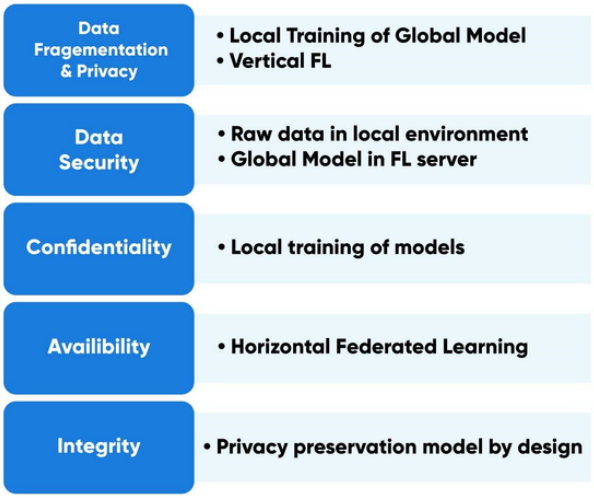
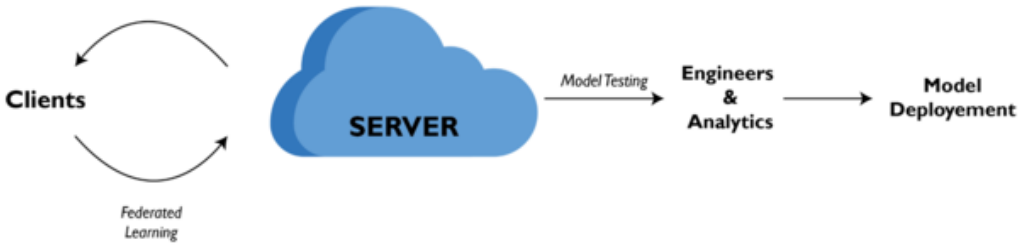


Figure 2. Federated learning system's life cycle, as well as the various actors involved



4. SECURE FL BASED DEEP LEARNING FRAMEWORK FOR CRITICAL INFRASTRUCTURE

In critical infrastructure, the privacy and security of data are of utmost importance. Thus, the Federated Learning (FL) algorithm can be utilized to protect sensitive data while training machine learning models. The steps involved in the FL algorithm remain the same as follows:

1. **Initialize the model:** Start by defining the model architecture and hyperparameters.
2. **Split the dataset:** Divide the dataset into multiple non-overlapping subsets based on the privacy requirements of the clients. The sample representation of the dataset used is shown in figure 3.

3. **Select clients:** Randomly select a subset of clients from the pool of all clients that have enough data to participate in the training process.
4. **Transmit the model:** Send the initial model to the selected clients.
5. **Train locally:** Each selected client trains the model on their local data without transmitting their data or model parameters.
6. **Aggregate the model:** Collect the model updates from the selected clients and aggregate them using secure aggregation methods like Federated Averaging.
7. **Update the global model:** Update the global model with the aggregated updates.
8. **Repeat the process:** Repeat steps 3-7 for multiple rounds until convergence or stopping criteria are met.
9. **Evaluate the model:** Evaluate the model on a separate test dataset to measure its performance.
10. **Privacy protection techniques:** Use privacy-enhancing techniques like differential privacy, secure multi-party computation, homomorphic encryption, and secure aggregation to protect the privacy of clients' data and model parameters during the training process. There are several privacy protection techniques that can be used in Federated Learning, including:
    A. **Differential privacy:** This technique adds noise to the aggregated model updates to ensure that individual clients' data remains private.
    B. **Secure multi-party computation:** This technique uses cryptographic protocols to allow multiple parties to jointly compute a function without revealing their private inputs.
    C. **Homomorphic encryption:** This technique allows the central server to perform computations on encrypted data without decrypting it, preserving the privacy of the data.
    D. **Federated Transfer Learning:** This technique allows the model to learn from other models and data sources without exposing the client's data or the global model.

    By using privacy protection techniques, Federated Learning can be used in sensitive domains such as healthcare, finance, and government where the privacy of the data is of utmost importance.
11. **Fine-tune the model:** Fine-tune the global model on the centralized data to improve its performance.
12. **Deploy the model:** Deploy the final model on the centralized server or clients' devices based on the privacy requirements and resource constraints.

    The deployment process involves several steps, including:
    A. **Converting the model to a deployable format:** The model is converted to a format that can be used by the target deployment platform or application.
    B. **Testing the model:** The model is tested in a controlled environment to ensure that it works as expected and performs well on real-world data.
    C. **Integration with the application:** The model is integrated with the larger system or application and made available for use by end-users.
    D. **Monitoring and maintenance:** The model is monitored over time to ensure that it continues to perform well and is updated as needed.

Algorithm 1: Implementing Federated Learning with a CNN model for critical infrastructure: Training a Global Model with Multiple Clients

```
import tensorflow and numpy
Global_model ←CNN Based neural network model
# Compile the global
global_model.comile(optimizer ←adam, Loss, Metrics ←accuracy)

 # Define the number of clients and the number of training rounds
num_clients ←number of clients
num_rounds ←number of rounds
# Define the dataset
Dataset ←Load the dataset
(x_train, y_train), (x_test, y_test) = Dataset
# Data pre-processing according to application
str ← input of sub-string to be present in main string

 for i in range num_clients do
    Starting_index_features ←(i ×(len(x_train)/num_clients)
     Ending_index_features ←((i + 1) ×(len(x_train)/num_clients)
     Starting_index_labels ←(i ×(len(y_train)/num_clients)
     Ending_index_labels ←((i + 1) ×(len(y_train)/num_clients)
     client_data ←x_train[Starting_index_features : Ending_index_features]
     client_labels ←y_train[Starting_index_labels : Ending_index_labels]
    clients ←append((client_data, client_labels))
end
# Train the global model
num_epochs ←number of epochs for the model
batch_size ←batch size for the model
global_model.fit(x_train, y_train, num_epochs, batch_size)
```

Algorithm 2: Implementing Federated Learning Loop for Distributed Training of a Global Model with Local Updates and Model Aggregation

```
# Federated learning loop
for rounds in range num_rounds do
    selected_client_indices ←select client indices
    selected_client ←select clients from above indices

     # Transmit the global model to the selected clients
    for rounds in range num_rounds do
        client_model ←clone global_model
        # set weight in client model
        client_model.set_weight( )
        # Compile the client model
        client_model.comile(optimizer ←adam, Loss, Metrics ←accuracy)
        # Train locally
        client_model.fit(client[0], client[1], epochs, batch_size)
        # Aggregate the model
        new_weights ←initialise empt list [ ]
        length ←len(global_model.get_weights( ))
        for i in range length do
            mean ←mean(client_model_weight[i],global_model_weight[i])
            new_layer_weights ← mean
            new_weights ←append(new_layer_weights)
        end
        # set new weights in global models
        global_model ←set_weights(new_weights)
    end
    # Evaluate the global model
    test_loss, test_acc ←global_model.evaluate(x_test, y_test)
    # .0print testing accuracy for each round
end
# Fine-tune the model
global_model ←fit(x_train, y_train, epochs, batch_size)
# Deploy the model
global_model.save('name_of_model.h5')
```

The algorithms 1 and 2 allows for training ML models using distributed data while maintaining the privacy of the individual clients. Federated learning and deep learning algorithms can work together to leverage the benefits of deep learning while addressing the privacy concerns associated with centralized data storage. The achieved graph for training and testing accuracy is shown in figure 4. It can be seen that accuracy is increasing with epochs. This framework is designed to be used in

critical infrastructure where high accuracy is required. The model has achieved testing accuracy of 98.4% and training accuracy of 99.8%. Furthermore, the proposed model is not overfitting even with these accuracies.

Figure 3. Sample representation of MNIST dataset used to implement the model

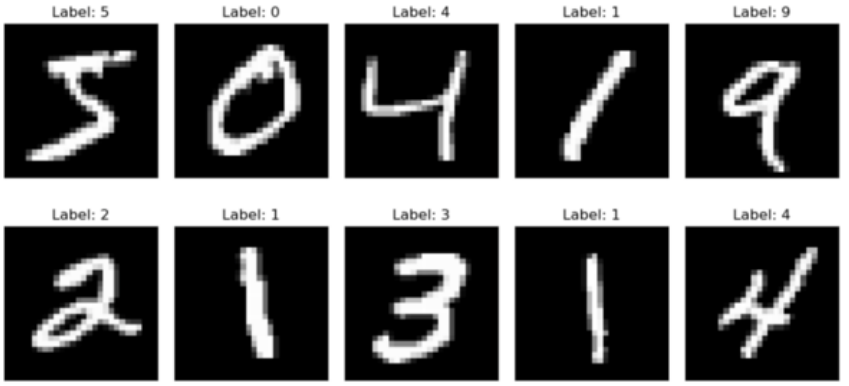
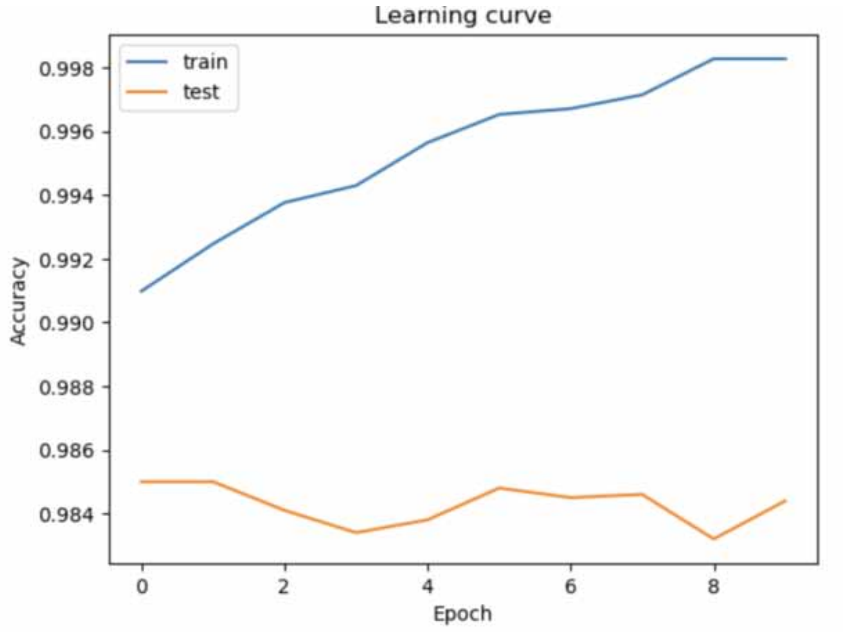


Figure 4. Accuracy of train and test vs Epoch curves of the model during training



5. SECURITY VULNERABILITIES AND ATTACKS IN FL DOMAIN

5.1 Sources of Vulnerabilities

Critical infrastructure vulnerabilities can be exploited by attackers, making it essential to understand and manage these vulnerabilities to defend against potential attacks. Open vulnerabilities can arise from non-secure channels, and homomorphic encryption is the current widely-used safeguard for

data in critical infrastructure. However, attackers can still obtain sensitive information, such as leaked gradients, which can be used to defeat defenses. Attackers can exploit vulnerabilities in critical infrastructure to obtain sensitive information about individuals. Gradient leakage can occur, allowing attackers to gain access to critical information about local data. Bad actors are challenging to detect, and they can use knowledge of benign participants to customize their updates, making Byzantine attacks more powerful than other attacks. It is crucial to identify and address vulnerabilities in the critical infrastructure to defend against potential attacks.

A server or a cluster of servers is frequently used in a cloud-based architecture for critical infrastructure. Cloud computing and hacking attacks, as well as distributed denial of service (DDoS) attacks (Devi & Bharti, 2022), can all be used to attack cloud computing infrastructures or physical servers. The server's security should be audited on a regular basis for any vulnerabilities that attackers might exploit. During Federated Learning in critical infrastructure, clients receive the global model during each training cycle, and local gradients are calculated and sent to a central aggregator. However, clients may encounter system issues, leading to a low-quality, skewed model if successful processes fail. The design of the training pipeline must integrate with the Federated Learning environment and address data privacy restrictions, making it challenging to detect bugs. It is essential to audit the Federated Learning process to detect any vulnerabilities that attackers could exploit.

In critical infrastructure, an adversary can intentionally create failures such as limited bandwidth or processing power to compromise the quality of the model being trained in Federated Learning. Technical difficulties, noisy feedback from clients, and network issues can also distort model updates. The dispersed nature of Federated Learning enables collusion and dispersed attacks, where clients from past, present, and future can work together to launch attacks on global model improvements. A recent example is the Dispersed Back-door attack (DBA), which takes advantage of the distributed properties of Federated Learning to create a unique conspiracy attack. DBA decomposes trigger patterns into local patterns altered for usage by adversarial parties, making it significantly more subtle and long-lasting when deployed against various datasets, including finance and image data. In non-homogeneous data, skewed Federated Learning can make it more challenging to detect false positives.

5.2 Attacks in Federated Learning

Critical infrastructure can also be vulnerable to data poisoning attacks in ML algorithms (A. Gupta et al., 2022b). Attackers can exploit vulnerabilities in the training process of ML algorithms used in critical infrastructure to compromise security and privacy standards. For example, if a ML algorithm is used to monitor and control a power grid, an attacker can inject malicious data to manipulate the system's behavior, resulting in power outages or other serious consequences.

Federated Learning (FL) is a decentralized machine learning approach that allows multiple clients to participate in data processing and model parameter sharing. However, the decentralized environment also creates the potential for malicious clients to interfere with the training process and poison the global model. This poses a significant threat to critical infrastructure as data poisoning attacks can affect the accuracy and reliability of the ML algorithm used in critical infrastructure (D. Li et al., 2019). In FL, data poisoning attacks are defined as the use of malicious examples to train the global model with the intention of obtaining global model parameters and sending them to the server. Data injection is a type of data poisoning where a rogue client injects dangerous data into the local model processing, allowing the malicious agent to take control of multiple clients' local models and eventually alter the global model with harmful information. As FL involves model updates from multiple clients, the risk of data poisoning attacks originating from one or more clients' training data is high, making it a significant threat to critical infrastructure that relies on accurate and reliable ML algorithms.

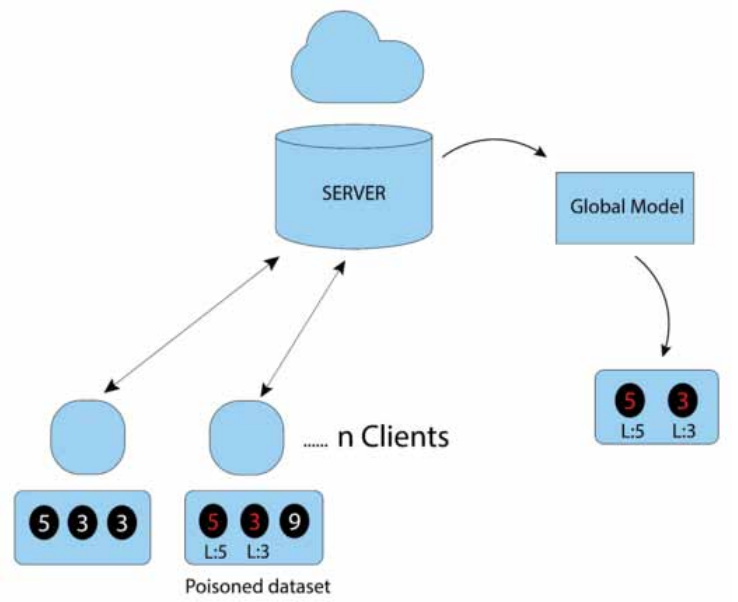
5.2.1 Dirty-Label Attacks

In critical infrastructure, data poisoning attacks can occur when attackers insert fictitious data into the training process of ML algorithms to manipulate their behavior, compromising the security and

privacy standards. For instance, a malicious actor could use label-flipping, a common type of dirty-label poisoning attack, to misidentify data samples with the desired label, causing the ML algorithm to produce inaccurate results. Model poisoning is another type of attack that involves directly attacking the global model rather than using fictitious data. The working of data poisoning attack is shown in figure 5. These attacks can have severe consequences, such as power outages, in critical infrastructure systems (Lim et al., 2020).

Data tampering/modification attacks, such as feature collision, can also be used to perform data poisoning attacks in critical infrastructure systems. These attacks can be accomplished by altering the training dataset, causing the ML algorithm to produce misleading results. In some cases, solutions can apply a shade or pattern from another class to a certain class, causing the ML algorithm to become confused. Random labeling of the training dataset is another option for attackers to carry out data poisoning attacks. Data injection and data modification attacks are examples of ML data poisoning threats in Federated Learning, which can significantly affect the accuracy and reliability of the ML algorithm used in critical infrastructure (Xie et al., 2020).

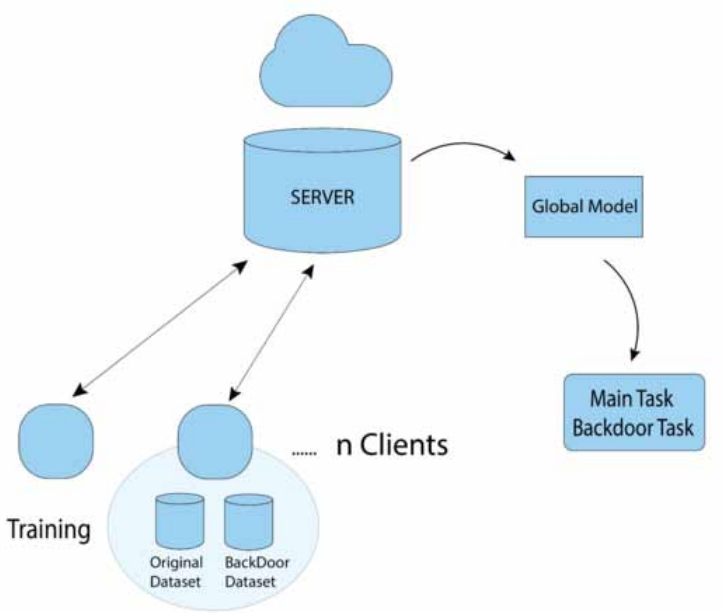
Figure 5. Data poisoning in federated learning



5.2.2 Backdoor

In the context of critical infrastructure, backdoor attacks can pose a serious threat to the security and reliability of ML algorithms used in critical infrastructure. Backdoor attacks can be used by attackers to introduce malicious code into the system, which can be triggered to cause a major security breach or system failure as shown in figure 6. This type of attack is particularly difficult to detect and prevent because it does not immediately affect the accuracy of the ML model. Researchers have investigated backdoor attacks in the context of FL, and have highlighted the significant impact that these attacks can have on the accuracy and reliability of the global model. In a study by (Hynes et al., 2018), the authors demonstrated how backdoor attacks can be used to predict ML algorithms and accurately forecast false positives. This highlights the potential danger of backdoor attacks in critical infrastructure and the need for robust security measures to prevent and detect these types of attacks.

Figure 6. BackDoor attack in federated learning



5.2.3 Evasion Attacks

In an avoidance attack (Biggio et al., 2013), (Szegedy et al., 2014), an adversary attempts to keep away from a sent model via cautiously controlling the information tests given to it. One normal technique for avoidance attack is the utilization of purported "ill-disposed examples," which are changed copies of test tests that appear to a human to be practically tradable with the first information tests. Ill-disposed information is made by extending blemish onto true information. The attacker attempts to affect the FL model's strength utilizing altered data. Adversarial preparing is a proactive guard approach that endeavors all changes of an attack from the start of the preparation to make the FL worldwide model impervious to known antagonistic attacks. (Tramèr et al., 2018) examines how to make the learning model powerful to antagonistic preparing attacks. The aftereffects of the assessment uncover that antagonistic preparation is as yet helpless against black-box attack. Therefore, they likewise give Ensemble Adversarial Training, a method that adds bothers to preparing information. Antagonistic Training diminishes the danger of delivering real preparing information through surmising by using ill-disposed examples, which advances client information protection.

5.2.4 Algorithms Oriented Attacks

It's respected to be a more modest arrangement of risks than the ones referenced already. The aggressor compromises the respectability of the conglomeration calculation (in the event that he approaches the server or the aggregator overall) or the neighborhood preparing pipeline, potentially by adjusting the enhancer's solid activity. The hyper-parameters of the nearby preparing plan might possibly be controlled by the attacker. This attack vector is more normal when the assailant has unlimited authority over at least one member at the FL framework's edge. It has the likelihood to slant determined figures inside the learning framework unintentionally (e.g., model updates).

5.2.4.1 Non-Robust Aggregation

Non-robust aggregation mechanisms in FL can result in compromised models in the event of adversarial attacks (Mani et al., 2020), according to (Fu et al., 2019). Additionally, the assumptions

of i.i.d data distribution and gradient uniformity among different clients are often shattered in real-life FL scenarios, making it difficult to assure the long-term viability of the aggregation algorithm. Moreover, poor reweighting procedures combined with aggregation strategies can lead to anomalous behavior in the global model.

5.2.4.2 Training Rules Manipulation

Attackers can also manipulate model training rules to sabotage FL computation, as discussed by (Tan et al., 2021). For instance, attackers with access to participating devices can change hyperparameters such as the no. of epochs, LR, and batch size so that the model can properly trained. Even small modifications to the training rules can result in the optimizer failing to converge, making the global model compromised.

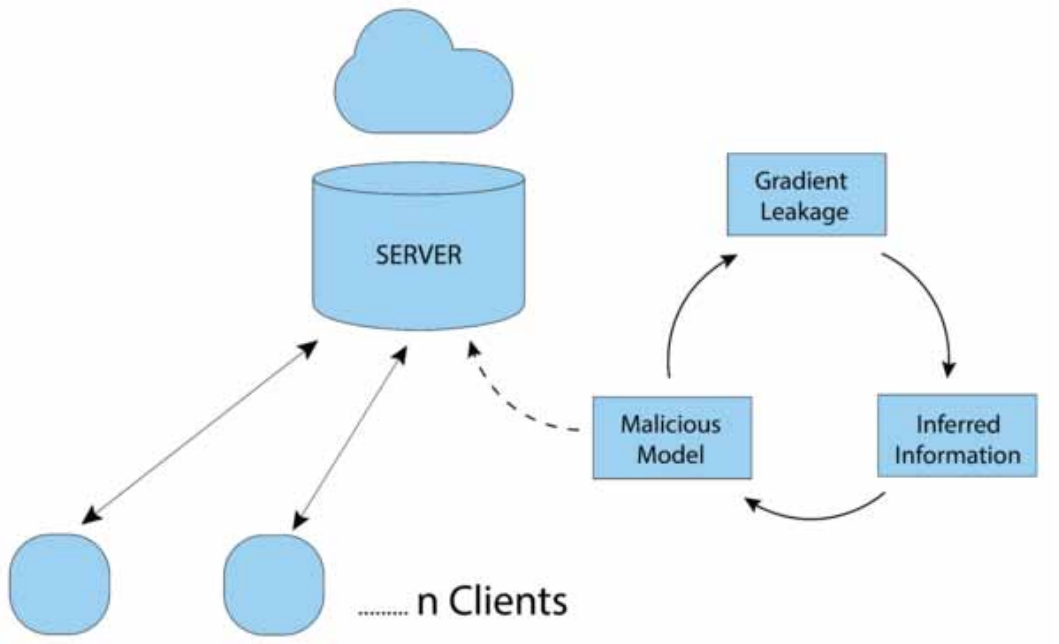
5.2.4.3 Compromised Distributed Computation

FL figuring, when in doubt, plans to survey a limit on an appropriated client dataset (by and large a significant neural association getting ready estimation, but it will in general be something clear, similar to a central computation). The current structure's information design can influence center outcomes, making them vulnerable against pernicious performers. There's an issue called irregularity, which insinuates a client's or a server's ability to show various individuals that they completed the arranged development without uncovering any of the limited intel on which they were acting.

5.2.5 Attacks Focused on Federation

Federated learning aims to protect the security of clients and players by providing model boundaries based on the results of local training. However, there are still risks such as enrollment inference attacks (shown in figure 7), accidental data leaks through GAN-based and inference-based attacks. Creating a federated system with optimal security characteristics is a challenging task due to multiple attacks targeting the FL system itself and factors such as data segmentation, network topologies, security settings, and goals.

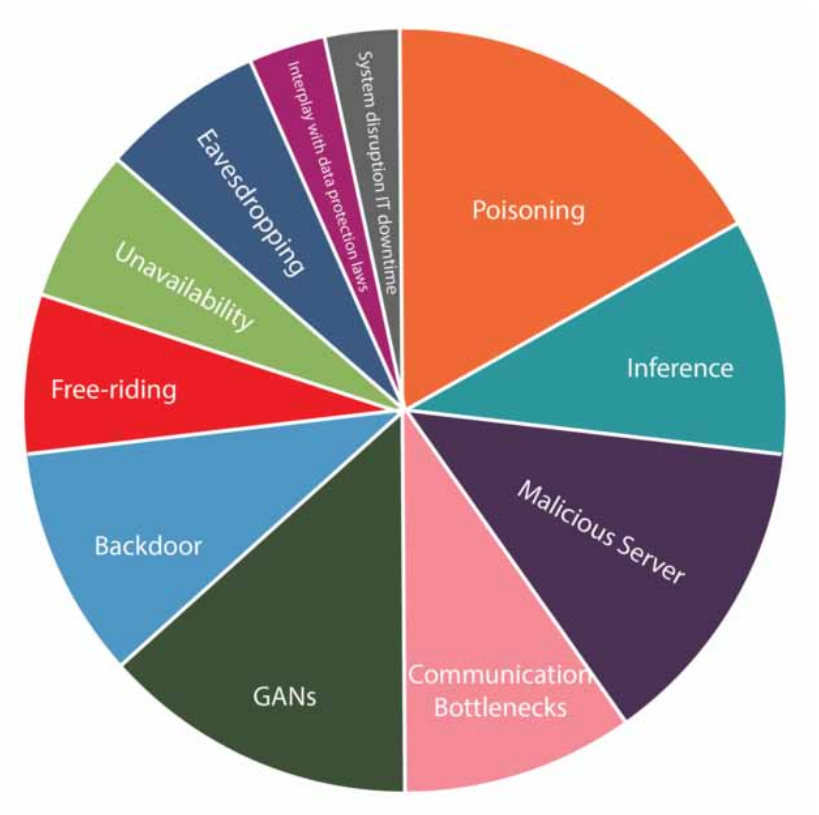
Figure 7. Inference attack in federated learning



Federated Learning (FL) was created to enhance user privacy, but there are still risks such as unintended data breaches, membership inference attacks, and GAN-based inference attacks. GAN-based client-side attacks can generate prototype samples from the targeted training set and endanger the privacy of the training set owner during real-time training. The mGANAI attack is a GAN-based server-side FL attack that boosts sample quality without altering the shared model or collaborative learning. The attack operates on the FL server side and is undetectable to the FL approach. During the GAN training phase, the attack relies on completing additional tasks and allows the retrieval of sensitive data at the client level, as the FL server can identify client identities.

In Federated Learning, the central server manages most of the cross-device network effort, including initial model setups, receiving updates, and deploying the global model. A malicious server, however, can cause significant damage to the global model if it is infected. It is possible for them to gain access to private client data or to update the global model in a harmful manner using shared processing power. In contrast, the server selects the clients' perspective on the joint model, which has a significant effect on the trained model. This presents an interesting problem in FL, where novel sustainable approaches are needed to explore a model's worst-case attack vulnerability, since only the server can control when clients can update the model during FL training. FL is particularly useful in gathering training data from client interactions with mobile apps. However, communication cost is a significant bottleneck, and free-riding attacks, where clients feign participation in the FL process, can have severe consequences, especially in a smaller FL environment with scarce data and high communication value models as shown in figure 8.

Figure 8. Severity of attacks in federated learning



Man-in-the-Middle attacks capture attacks traded among customers and servers and supplant them with malignant model changes (Bhushan et al., 2018). These attacks can be conveyed either by meddling with real organizations or by developing fake organizations that the assailant controls. Compromised correspondence is regularly deprived of encryption to take, change, or reroute passed on model updates to the aggressor's ideal objective. Enemies might be unobtrusively observing or rescrambling taken traffic to its expected objective at whatever point it is put away or adjusted, making an attack hard to distinguish. FL relies on the participation of customers during the training process, but some customers may drop out due to various reasons, which can result in inadequate preparation of the global model. This problem is similar to the free-riding attack, but customers may miss training due to unforeseeable obstacles rather than deliberately avoiding participation and has been presented in table 2. This can affect the flexibility of the aggregation process, and it is important to use asynchronous aggregation methods to address this issue. The risk of this problem is moderate, as the chance of customer dropouts is low, but it can still impact the quality of the final model.

Table 1. Conclusion of all the attacks, solution, and solution challenges

Type of Attack	Description of Attack	Current Solution	Current Solution's Challenge
Inference attack	Unintentional Data Leakage By Reconstruction Through The Inference Models And Generative Adversarial Network	The enhancement of FL model privacy involves combining Differential Privacy (DP) with conventional shuffling techniques. Additionally, an algorithm named the invisibility cloak algorithm is employed to obscure user data.	This methodology might reduce the training performance of the FL model as there might be uncertainty in the parameters that are uploaded.
Backdoor attacks	Attackers try to inject a task that is malicious within the existing FL model without affecting the rate of accuracy when the actual task is performed.	A strategy called SAFELearning is introduced to empower FL model users in identifying backdoor attacks during the aggregation of model parameters. This method accomplishes the goal of backdoor attack detection through the application of two techniques known as oblivious random grouping and partial parameter disclosure. Additionally, the effectiveness of PoisonGAN and DataGen models is highlighted.	The process of detecting such attacks is a highly time-consuming process as there is no impact on the accuracy of the model.
Adversarial attack	Some clients try to deduce the information of other clients in the landscape.	Two attack models, namely PoisonGAN and DataGen, are introduced to iteratively regenerate the victim's samples using the global model parameters.	N.A.
Free-riding attacks	In some situations, the passive client might try to inject a few dummy parameters to update the global model without performing training with their local data.	Introducing the BytoChain framework, a proposal aimed at enhancing model verification through a parallel process involving dedicated verifiers. These verifiers extensively assess the models, employing a concept known as Proof of Accuracy. The framework is constructed based on blockchain-powered Federated Learning (FL).	Slow process Harder to Scale

Table 2. Conclusion of current defenses, description, and attacks

Defenses	Description	Attacks Defended Against
Anomaly Detection	Identification of patterns or instances that deviate significantly from the expected behavior in any given system	● Free-Riding Attacks ● Data Poisoning ● Model Poisoning
Differential Privacy	Protection of sensitive data by adding noise in a controlled manner, preventing adversaries from distinguishing if a specific information is present or not	● Data Poisoning ● Inference Attack ● Model Poisoning ● Evasion Attack ● Distributed Computation
Trusted Execution Environment	Prevention of attacks by creating a secure and isolated ecosystem in a processor, ensuring confidential computations and data integrity.	● Malicious Server ● Training Rules Manipulation ● Compromised FL
Robust Aggregation	It combines and summarizes data from multiple sources, preventing manipulation or compromise during the aggregation process, enhancing the integrity and reliability of the aggregated information.	● Dropout of Clients ● Data Poisoning ● Backdoor Attacks ● Model Poisoning ● Non-Robust Aggregation
Pruning	Involves the selective removal of potentially vulnerable elements of a system, reducing attack surfaces and minimizing the risk of exploitation by malicious actors.	● Communication Bottlenecks ● Backdoor Attacks
Zero-Knowledge Proofs	Cryptographic protocols are used to validate the possession of specific knowledge without disclosing the actual information, thus safeguarding data integrity and confidentiality during authentication or verification, thereby mitigating exposure risks in cybersecurity.	● Model Poisoning ● Data Poisoning ● Backdoor Attacks ● Man-in-the-Middle Attacks
Adversarial Training	Optimizing a ML model by iteratively adjusting both adversarial samples and model parameters. This involves training on both original and crafted adversarial data, enhancing the model's ability to withstand and counter adversarial inputs for heightened security and performance.	● Evasion Attacks
Federated Multi-Task Learning	Training of models for interconnected tasks, boosting fault tolerance and creating a resilient system capable of handling challenges associated with individual tasks. Thus, contributing to overall system versatility and reliability.	● Dropout of Clients ● GAN Attacks
Moving Target Defense	Enhance cybersecurity by introducing dynamic randomness to system modules in order to minimize successful attacks and decrease attack duration in Moving Target Defense.	● Reconstruction Attacks ● Inference Attacks ● Communications Bottlenecks ● Man-in-the-middle Attacks
Recognizing Legitimate Clients	Determine whether a client is genuine in the presence of multiple adversaries to significantly mitigate the effectiveness of poisoning attacks.	● Model Poisoning ● Backdoor Attacks ● Data Poisoning
Federated Distillation	Adversaries exploit the knowledge transfer process between a fully trained model and a student model, capitalizing on knowledge sharing's emphasis over weights sharing. This undermines the robustness of FL and can compromise the efficiency gains in communication and computation costs.	● Communication ● Reconstruction Attack ● Bottlenecks ● Inference Attack ● Man-in-the-middle Attacks

6. DEFENSES IN FEDERATED LEARNING

Various issues with joined learning systems can be considered as ensuring power: clean data gets degraded or interfered with, whether or not purposely or unexpectedly. Insurance has actually been depicted as a strength in data security research, very differential security (DP). As an assurance system, differential security has a lot of drawing in components. In any case, it gives strong, most negative situation confirmation against a wide extent of risks. Second, different private techniques are known, and the defense may be used to settle different AI issues.

Poisoning attacks are handled using a model upgrade that includes:

(1) putting a norm restriction on the client model update (e.g., by clipping the client updates)
(2) combining the updates that have been cut
(3) By injecting Gaussian noise into the aggregate, the service provider can limit the overall model contribution of any individual client.

Overfitting to a single update or a small number of troublesome individuals is avoided using this method, which is comparable to differential privacy training. Recently, researchers looked into this technique and discovered preliminary success in using differential privacy as a defense against targeted attacks. (Sun et al., 2019) recommend that they broaden the scope of their targeted attacks and tests to include more generic adversarial tactics. Huang et al. (Huang et al., 2022) suggested that edge case backdoors, created by low probability data samples in the distribution underlying those samples, can be used to exploit differential privacy measures. Further, table 2 has summarized all the defense and description of attacks.

Protecting yourself from data poisoning attacks Data poisoning is the incapacity of a learning system to be resilient, and it arises when a few attacked training samples have a big impact on the learnt model. As a result, making the learning mechanism differentially private, which increases resistance, is a natural way to prevent these attacks. Data poisoning precautions such as differential privacy have been studied recently (Ma et al., 2019).

6.1 Anomaly Detection

FL systems may use anomaly detection techniques to detect attacks like data poisoning and model poisoning. Anomaly detection is a type of proactive security that detects and stops harmful updates. One popular strategy is to calculate the test error rate of a single update and reject it if it fails to improve the global model (Carminati et al., 2020). Although these anomaly detectors are successful against untargeted adversarial attacks, when trained on backdoor data, they are more likely to fail because poisoned model updates look and act like infected models.

6.2 Differential Privacy

Differential privacy was created to protect against data poisoning, but it may also be used to protect against intrusions of privacy (Ma et al., 2019). It works by introducing a degree of unpredictability into the updates. The goal of differential privacy is to ensure that no single data record can be reliably distinguished from others (Abadi et al., 2016). An attacker with only a few training samples should theoretically be unable to have a significant impact on the distribution of learned models.

The most serious problem in differential privacy is that the clatter introduced by the learning approach is added to the noise generated by the computational approach. The noise that has built up has the potential to taint the model that has been learned.

6.3 Recognizing Legitimate Clients

For many years, researchers have investigated numerous poisoning attacks in a centralized FL environment, including inclusive model poisoning and data poisoning aggressive attacks. The

effectiveness of distributed poisoning with multiple attacks over traditional poisoning is questionable despite scattered poisoning being a greater threat in FL. Even in the presence of a high number of enemies, this defense system detects legitimate users, greatly lowering the success rate of poisoning attempts.

6.4 Defending Against Inference Attacks

Differential Privacy (DP) is integrated into the FL model with traditional shuffling, and they employ an algorithm called the invisibility cloak algorithm to hide users' data (Ghazi et al., 2019). This practice, however, may reduce the FL model's training performance because the parameters that are uploaded may be unknown. VerifyNet (Xu et al., 2020) is a privacy-protection system that uses a double-mask approach to prevent attackers from randomly generating train data. Researchers should focus on developing novel frameworks that reduce communication overhead while preserving privacy.

6.5 Defending Against Backdoor Attacks

During the collection of model parameters, a technique called SAFELearning is being created to allow FL model users to discover backdoor threats (Z. Zhang et al., 2021). The technology detects the backdoor attack by employing 2 strategies: partial parameter disclosure and oblivious random grouping.

6.6 Defending Against Adversarial Attacks

PoisonGAN and DataGen are two attack models proposed in (J. Zhang et al., 2021) for iteratively replicating the victim's data using the parameters of the global model. The testing of these models was based on a FL prototype, and the results revealed in addition to adversarial attacks, they are also successful at backdoor attacks and label flipping.

6.7 Defending against Free Riding Attacks

For better model verification, a framework called BytoChain is presented (Z. Li et al., 2021). The verification is carried out parallelly by adding verifiers who will thoroughly test the models. The technique called Proof of Accuracy is employed. The framework is based on FL, which is a blockchain-based technology.

7. LIMITATIONS OF EXISTING SOLUTIONS

Since the objective of FL is for the server to show the populace example of customer information, the objective of regular security is to measure and maybe oblige the server's capacity to reproduce the info information of individual customers. is. This incorporates

(a) Officially characterizing what is a perspective on customer information presented to the server because of FL execution
(b) What is the security loss of such a view? increment. In FL, we are especially keen on permitting the server to total reports from customers while simultaneously concealing the commitments of individual customers here and there. This should be possible in an assortment of ways.

You commonly utilize the idea of delta information insurance. There are a wide range of ways of doing this, particularly in FL, each with its own shortcomings. For instance, as referenced over, the issue is that the focal DP needs to be confided in the central server. This prompted other promising private divulgence strategies that are already portrayed in this paper. Here we diagram a portion of the shortcomings of these strategies.

As previously said, LDP eliminates the need for a presumed central server by requiring each customer to execute a delta private modification of the report before sending it to the central server. The LDP anticipates that client protection will arise only from the customer's own addition of irregularity. As a result, the client information insurance guarantees that it is free of any further arbitrariness added by any lingering clients. In greater layered information settings, the LDP convention is convincing and hypothetically supported for carrying out information insurance while achieving nearby differential protection and protecting the utility. Numerous results indicate that it is a test. Because of the method that the amount of irregular commotion familiar should be similar with the size of the sign in the information, this problem is worsen. This can need combining reports from many consumers. equivalent to the central setup, equivalent benefits with LDP need a substantial customer base or boundary selection.

The hybrid differential information assurance model diminishes the size of the client base needed by dividing clients dependent on trust settings. Be that as it may, it is hazy which application regions and calculations can best utilize the half breed trust model information. Also, current work on half breed models ordinarily expects that the information are from a similar circulation, paying little heed to the client's trust inclinations. Loosening up this supposition that is particularly significant for FL, as the connection between trust settings and real client information can be precarious.

This strategy, however, has several drawbacks. Sparse vector aggregation is inefficient, (A) assuming a semi-honest server (private key infrastructure phase only), (B) allowing the server to recognize a group of peripherals (which may lose information), (C) and (D) does not have the capacity to enforce well-formed expressions on client input. It is still unclear how to develop an effective, reliable, and secure aggregation system that handles all of these issues.

8. APPLICATIONS OF FL IN CYBERSECURITY

There are numerous applications of FL in cybersecurity that can benefit critical infrastructure systems which are discussed below in the section.

8.1 Federated Learning for Authentication

A potential method for developing user authentication models for vital infrastructure is FL. A decision problem called UA makes use of an embedding space to examine input similarity. Authentication models must be trained on a variety of data sources in order to successfully reject forgers. However, centralized user data collection and model training presents privacy issues. FL eliminates the need for a central server and restricts access to other users' data, minimizing privacy problems. FL can therefore aid in enhancing the efficacy of authentication methods by thwarting assaults like evasion and poisoning.

8.2 Learning for Privacy

In the context of critical infrastructure, traditional machine learning models send sensitive data to a centralized cloud (Stergiou et al., 2022) for training, which poses a security risk. Federated learning (FL) provides personalized predictions on local hosts while keeping sensitive data locally on mobile or edge devices. FL only transmits parameters from these devices to train the global model, ensuring data privacy and security. FL can be used in various critical infrastructure applications such as social media, medical applications (Vijayakumar P. et al., 2022), traffic management, and smart city apps.

8.3 Federated Learning for Trust Management

Massive amounts of data generated by IoT and smart phones (R. Sharma & Sharma, 2022) are utilized to train machine learning models for improved mobile services. Storing user data on a centralized server for model training, on the other hand, increases communication costs, storage space requirements,

and the danger of data privacy leaks. FL solves these concerns by running predictive algorithms on local devices and communicating only the parameters needed to run the global ML model for larger-scale predictions, while preserving sensitive data locally. Despite this, FL is still subject to security threats and requires trust measures to detect untrustworthy local model changes. Traditional machine learning includes transferring data to a centralized cloud for model training, exposing sensitive data to intruders/hackers.

8.4 Federated Learning for Attack Detection

The security of sensitive data from prospective attackers is crucial in critical infrastructure. The danger of data being accessible to unauthorized users has grown as digitalization and online transactions have expanded. An intrusion detection system (IDS) (Yadav et al., 2021) and anomaly detection can help identify and detect malicious network users or intruders. Machine learning-based models have shown potential in spotting intruder patterns, which can assist in the construction of an effective intrusion detection and anomaly detection system. Critical infrastructure must have strong security measures in place to avoid any possible assaults on sensitive data.

9. CONCLUSION AND FUTURE WORK

Federated Learning is a new learning paradigm that addresses the increased processing capacities of devices such as wearables, smartphones, and self-driving cars, as well as privacy concerns. Federated learning was proposed as a way to extend the advantages of machine learning to places where sensitive data is stored. In a range of areas, such as education, finance, and health insurance, learning from private data while respecting user privacy has immense promise. However, it provides a number of open concerns and ideas for future federated learning research, such as how to prevent or safeguard the vulnerabilities in FL systems. This research came to a conclusion after looking into a number of vulnerabilities and attacks. The most common dangers in FL are model poisoning, backdoor attacks, and inference attacks. Continuous vulnerability assessment is critical for ensuring the security of FL systems. Current detection and mitigation methods are often limited in their effectiveness, and new methods need to be developed that are tailored to the unique challenges of FL systems. Existing FL protocols often have security vulnerabilities, and new protocols need to be designed that are more robust to attack. Promising new cryptographic techniques, such as trusted execution environments (TEEs) and zero-knowledge proofs (ZKPs), have the potential to significantly improve FL security. Using cryptographic protocols like trustworthy Execution Environments and Zero-Knowledge Proofs to build and ensure agreement among cooperative parties may give crucial protection against failures and attacks that must be addressed in this privacy protection through federated learning.

REFERENCES

Abadi, M., McMahan, H. B., Chu, A., Mironov, I., Zhang, L., Goodfellow, I., & Talwar, K. (2016). Deep learning with differential privacy. *Proceedings of the ACM Conference on Computer and Communications Security*. ACM. doi:10.1145/2976749.2978318

Aggarwal, K., Singh, S. K., Chopra, M., Kumar, S., & Colace, F. (2022). Deep Learning in Robotics for Strengthening Industry 4.0.: Opportunities, Challenges and Future Directions. In N. Nedjah, A. A. Abd El-Latif, B. B. Gupta, & L. M. Mourelle (Eds.), *Robotics and AI for Cybersecurity and Critical Infrastructure in Smart Cities* (pp. 1–19). Springer International Publishing. doi:10.1007/978-3-030-96737-6_1

Alazab, M., Rm, S. P., Parimala, M., Maddikunta, P. K. R., Gadekallu, T. R., & Pham, Q. V. (2022). Federated Learning for Cybersecurity: Concepts, Challenges, and Future Directions. *IEEE Transactions on Industrial Informatics*, *18*(5), 3501–3509. doi:10.1109/TII.2021.3119038

Arafeh, M., Hammoud, A., Otrok, H., Mourad, A., Talhi, C., & Dziong, Z. (2022). Independent and Identically Distributed (IID) Data Assessment in Federated Learning. *2022 IEEE Global Communications Conference, GLOBECOM 2022 - Proceedings*. IEEE. doi:10.1109/GLOBECOM48099.2022.10001718

Bhardwaj, A., & Kaushik, K. (2022). Predictive Analytics-Based Cybersecurity Framework for Cloud Infrastructure. *International Journal of Cloud Applications and Computing*, *12*(1), 1–20. doi:10.4018/IJCAC.297106

Bhushan, B., Sahoo, G., & Rai, A. K. (2018). Man-in-the-middle attack in wireless and computer networking—A review. *Proceedings - 2017 3rd International Conference on Advances in Computing, Communication and Automation (Fall), ICACCA 2017*. IEEE. doi:10.1109/ICACCAF.2017.8344724

Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G., & Roli, F. (2013). *Evasion attacks against machine learning at test time*. Lecture Notes in Computer Science. Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics., doi:10.1007/978-3-642-40994-3_25

Bouncken, R. B., Lapidus, A., & Qui, Y. (2022). Organizational sustainability identity: 'New Work' of home offices and coworking spaces as facilitators. *Sustainable Technology and Entrepreneurship*, *1*(2), 100011. doi:10.1016/j.stae.2022.100011

Carminati, M., Santini, L., Polino, M., & Zanero, S. (2020). Evasion attacks against banking fraud detection systems. *RAID 2020 Proceedings - 23rd International Symposium on Research in Attacks, Intrusions and Defenses*.

Chopra, M., Kumar, S., Madan, U., & Sharma, S. (2021). Influence and Establishment of Smart Transport in Smart Cities. *International Conference on Smart Systems and Advanced Computing (Syscom-2021)*. IEEE.

Chopra, M., Singh, D. S. K., Gupta, A., Aggarwal, K., Gupta, B. B., & Colace, F. (2022). Analysis & prognosis of sustainable development goals using big data-based approach during COVID-19 pandemic. *Sustainable Technology and Entrepreneurship*, *1*(2), 100012. doi:10.1016/j.stae.2022.100012

Devi, S., & Bharti, T. S. (2022). A Review on detection and Mitigation Analysis of distributed denial of Service Attacks and Their effects on the Cloud. *International Journal of Cloud Applications and Computing*, *12*(1), 1–21. doi:10.4018/IJCAC.311036

Gupta, A., Bansal, A., Mamgain, K., & Gupta, A. (2022a). *An Exploratory Analysis on the Unfold of Fake News During COVID-19 Pandemic*. Smart Innovation, Systems and Technologies., doi:10.1007/978-981-16-2877-1_24

Gupta, A., Bansal, A., Mamgain, K., & Gupta, A. (2022b). *An Exploratory Analysis on the Unfold of Fake News During COVID-19 Pandemic*. Smart Innovation, Systems and Technologies., doi:10.1007/978-981-16-2877-1_24

Gupta, S., Agrawal, S., Singh, S. K., & Kumar, S. (2023). A Novel Transfer Learning-Based Model for Ultrasound Breast Cancer Image Classification. In S. Smys, J. M. R. S. Tavares, & F. Shi (Eds.), *Computational Vision and Bio-Inspired Computing* (pp. 511–523). Springer Nature. doi:10.1007/978-981-19-9819-5_37

Huang, W., Li, T., Wang, D., Du, S., Zhang, J., & Huang, T. (2022). Fairness and accuracy in horizontal federated learning. *Information Sciences*, *589*, 170–185. doi:10.1016/j.ins.2021.12.102

Hynes, N., Cheng, R., & Song, D. (2018). Efficient and Private Federated Learning using TEE. *arXiv*.

Jalali, N. A., & Chen, H. (2023). Security Issues and Solutions in Federate Learning Under IoT Critical Infrastructure. *Wireless Personal Communications*, *129*(1), 475–500. doi:10.1007/s11277-022-10107-3

Kaur, P., Singh, S. K., Singh, I., & Kumar, S. (2021). Exploring Convolutional Neural Network in Computer Visionbased Image Classification. *CEUR Workshop Proceedings* in the Proceedings of International Conference on Smart Systems and Advanced Computing (Syscom 2021), Vol.3080, pp 147-155, 2021.

Khade, G., Kumar, S., & Bhattacharya, S. (2012). Classification of web pages on attractiveness: A supervised learning approach. *2012 4th International Conference on Intelligent Human Computer Interaction (IHCI)*. IEEE. doi:10.1109/IHCI.2012.6481867

Kumar, P. S. (2022). Computationally Simple and Efficient Method for Solving Real-Life Mixed Intuitionistic Fuzzy 3D Assignment Problems. [IJSSCI]. *International Journal of Software Science and Computational Intelligence*, *14*(1), 1–42. doi:10.4018/IJSSCI.309425

Kumar, S., Singh, S., Aggarwal, N., & Aggarwal, K. (2021). Evaluation of automatic parallelization algorithms to minimize speculative parallelism overheads: An experiment. *Journal of Discrete Mathematical Sciences and Cryptography*, *24*(5), 1517–1528. doi:10.1080/09720529.2021.1951435

Kumar, S., Singh, S. K., & Aggarwal, N. (2023). Speculative Parallelism on Multicore Chip Architecture Strengthen Green Computing Concept: A Survey. In *Advanced Computer Science Applications*. Apple Academic Press. doi:10.1201/9781003369066-2

Kumar, S., Singh, S. K., Aggarwal, N., Gupta, B. B., Alhalabi, W., & Band, S. S. (2022). An efficient hardware supported and parallelization architecture for intelligent systems to overcome speculative overheads. *International Journal of Intelligent Systems*, *37*(12), 11764–11790. doi:10.1002/int.23062

Lee, M. T., & Suh, I. (2022). Understanding the effects of Environment, Social, and Governance conduct on financial performance: Arguments for a process and integrated modelling approach. *Sustainable Technology and Entrepreneurship*, *1*(1), 100004. doi:10.1016/j.stae.2022.100004

Li, D., Deng, L., Bhooshan Gupta, B., Wang, H., & Choi, C. (2019). A novel CNN based security guaranteed image watermarking generation scenario for smart city applications. *Information Sciences*, *479*, 432–447. Advance online publication. doi:10.1016/j.ins.2018.02.060

Li, D., Lai, J., Wang, R., Li, X., Vijayakumar, P., Gupta, B. B., & Alhalabi, W. (2023). Ubiquitous intelligent federated learning privacy-preserving scheme under edge computing. *Future Generation Computer Systems*, *144*, 205–218. doi:10.1016/j.future.2023.03.010

Li, Z., Yu, H., Zhou, T., Luo, L., Fan, M., Xu, Z., & Sun, G. (2021). Byzantine Resistant Secure Blockchained Federated Learning at the Edge. *IEEE Network*, *35*(4), 295–301. doi:10.1109/MNET.011.2000604

Lim, W. Y. B., Luong, N. C., Hoang, D. T., Jiao, Y., Liang, Y. C., Yang, Q., Niyato, D., & Miao, C. (2020). Federated Learning in Mobile Edge Networks: A Comprehensive Survey. *IEEE Communications Surveys and Tutorials*, *22*(3), 2031–2063. doi:10.1109/COMST.2020.2986024

Ma, Y., Zhu, X., & Hsu, J. (2019). Data poisoning against differentially-private learners: Attacks and defenses. *IJCAI International Joint Conference on Artificial Intelligence*. IEEE. doi:10.24963/ijcai.2019/657

Mandle, A. K., Sahu, S. P., & Gupta, G. P. (2022). CNN-Based Deep Learning Technique for the Brain Tumor Identification and Classification in MRI Images. *International Journal of Software Science and Computational Intelligence*, *14*(1), 1–20. doi:10.4018/IJSSCI.304438

Mani, N., Moh, M., & Moh, T.-S. (2020). Defending Deep Learning Models Against Adversarial Attacks. *International Journal of Software Science and Computational Intelligence*, *13*(1), 72–89. doi:10.4018/IJSSCI.2021010105

Marinakis, Y. D., & White, R. (2022). Hyperinflation potential in commodity-currency trading systems: Implications for sustainable development. *Sustainable Technology and Entrepreneurship*, *1*(1), 100003. doi:10.1016/j.stae.2022.100003

Mengi, G., Singh, S. K., Kumar, S., Mahto, D., & Sharma, A. (2023). Automated Machine Learning (AutoML): The Future of Computational Intelligence. In N. Nedjah, G. Martínez Pérez, & B. B. Gupta (Eds.), *International Conference on Cyber Security, Privacy and Networking (ICSPN 2022)* (pp. 309–317). Springer International Publishing. doi:10.1007/978-3-031-22018-0_28

Pan, X., Yamaguchi, S., Kageyama, T., & Kamilin, M. H. B. (2022). Machine-Learning-Based White-Hat Worm Launcher in Botnet Defense System. *International Journal of Software Science and Computational Intelligence*, *14*(1), 1–14. doi:10.4018/IJSSCI.291713

Peñalvo, F. J. G., Maan, T., Singh, S. K., Kumar, S., Arya, V., Chui, K. T., & Singh, G. P. (2022). Sustainable Stock Market Prediction Framework Using Machine Learning Models. [IJSSCI]. *International Journal of Software Science and Computational Intelligence*, *14*(1), 1–15. doi:10.4018/IJSSCI.313593

Peñalvo, F. J. G., Sharma, A., Chhabra, A., Singh, S. K., Kumar, S., Arya, V., & Gaurav, A. (2022). Mobile Cloud Computing and Sustainable Development: Opportunities, Challenges, and Future Directions. [IJCAC]. *International Journal of Cloud Applications and Computing*, *12*(1), 1–20. doi:10.4018/IJCAC.312583

Rathod, T., Jadav, N. K., Tanwar, S., Polkowski, Z., Yamsani, N., Sharma, R., Alqahtani, F., & Gafar, A. (2023). AI and Blockchain-Based Secure Data Dissemination Architecture for IoT-Enabled Critical Infrastructure. *Sensors (Basel)*, *23*(21), 8928. doi:10.3390/s23218928 PMID:37960626

Saini, T., Kumar, S., Vats, T., & Singh, M. (2020). *Edge Computing in Cloud Computing Environment: Opportunities and Challenges*.

Sharma, A., Singh, S. K., Kumar, S., Chhabra, A., & Gupta, S. (2023). *Security of Android Banking Mobile Apps: Challenges and Opportunities*. Lecture Notes in Networks and Systems. doi:10.1007/978-3-031-22018-0_39

Sharma, R., & Sharma, N. (2022). Attacks on Resource-Constrained IoT Devices and Security Solutions. *International Journal of Software Science and Computational Intelligence*, *14*(1), 1–21. doi:10.4018/IJSSCI.310943

Singh, I., Singh, S. K., Kumar, S., & Aggarwal, K. (2022). Dropout-VGG Based Convolutional Neural Network for Traffic Sign Categorization. In Lecture Notes on Data Engineering and Communications Technologies. doi:10.1007/978-981-16-9416-5_18

Singh, I., Singh, S. K., Singh, R., & Kumar, S. (2022). Efficient Loop Unrolling Factor Prediction Algorithm using Machine Learning Models. *2022 3rd International Conference for Emerging Technology (INCET)*. IEEE. doi:10.1109/INCET54531.2022.9825092

Singh, M., Singh, S. K., Kumar, S., Madan, U., & Maan, T. (2023). Sustainable Framework for Metaverse Security and Privacy: Opportunities and Challenges. In N. Nedjah, G. Martínez Pérez, & B. B. Gupta (Eds.), *International Conference on Cyber Security, Privacy and Networking (ICSPN 2022)* (pp. 329–340). Springer International Publishing. doi:10.1007/978-3-031-22018-0_30

Singh, R., Singh, S. K., Kumar, S., & Gill, S. S. (2022). SDN-Aided Edge Computing-Enabled AI for IoT and Smart Cities. In *SDN-Supported Edge-Cloud Interplay for Next Generation Internet of Things*. Chapman and Hall/CRC. doi:10.1201/9781003213871-3

Stergiou, C. L., Psannis, K. E., & Gupta, B. B. (2022). InFeMo: Flexible Big Data Management Through a Federated Cloud System. *ACM Transactions on Internet Technology*, *22*(2), 1–22. doi:10.1145/3426972

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2014). Intriguing properties of neural networks. *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*. IEEE.

Tan, J., Liang, Y. C., Luong, N. C., & Niyato, D. (2021). Toward Smart Security Enhancement of Federated Learning Networks. *IEEE Network*, *35*(1), 340–347. doi:10.1109/MNET.011.2000379

Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., & McDaniel, P. (2018). Ensemble adversarial training: Attacks and defenses. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*. IEEE.

Vijayakumar, P., Jegatha Deborah, L., & Rajkumar, S. C. (2022). Deep Reinforcement Learning-Based Pedestrian and Independent Vehicle Safety Fortification Using Intelligent Perception. *International Journal of Software Science and Computational Intelligence*, *14*(1), 1–33. doi:10.4018/IJSSCI.291712

Xie, C., Huang, K., Chen, P. Y., & Li, B. (2020). DBA: DISTRIBUTED BACKDOOR ATTACKS AGAINST FEDERATED LEARNING. *8th International Conference on Learning Representations, ICLR 2020*. IEEE.

Xu, G., Li, H., Liu, S., Yang, K., & Lin, X. (2020). VerifyNet: Secure and Verifiable Federated Learning. *IEEE Transactions on Information Forensics and Security*, *15*, 911–926. doi:10.1109/TIFS.2019.2929409

Yadav, K., Gupta, B. B., Hsu, C. H., & Chui, K. T. (2021). Unsupervised Federated Learning based IoT Intrusion Detection. *2021 IEEE 10th Global Conference on Consumer Electronics, GCCE 2021*. IEEE. doi:10.1109/GCCE53005.2021.9621784

Zhang, J., Chen, B., Cheng, X., Binh, H. T. T., & Yu, S. (2021). PoisonGAN: Generative Poisoning Attacks against Federated Learning in Edge Computing Systems. *IEEE Internet of Things Journal*, *8*(5), 3310–3322. doi:10.1109/JIOT.2020.3023126