A Federated Learning Framework for Privacy Preserving Threat Detection in Zero Trust Network Access (ZTNA)

Md. Mushfiqur Rahman[1*], Sakib Ur Rahman[2], Moinoddeen Quader Al Arabi[3], Kamran Hassan Shomrat[4], Kazi Sanghati Sowharda Haque[5], Mehadi Hasan Foysal[6], Sazzad Hossain[7].

[1*]Department of System Management and Information Security, Samarkand State University, Samarkand, Uzbekistan.
[2]Dept. of CSE, University of Information Technology and Sciences, Dhaka, Bangladesh
[3]Dept. of CSE, Chittagong University of Engineering Technology, Chattogram, Bangladesh.
[4]Dept. of CSE, Brac University, Dhaka, Bangladesh.
[5]IIT, University of Dhaka, Dhaka, Bangladesh.
[6]Dept. of CSE, Bangladesh University of Professional (BUP), Dhaka, Bangladesh.
[7]Department of System Management and Information Security.
Samarkand State University, Samarkand, Uzbekistan

Mushfique98@gmail.com, shakib.asif.77@gmail.com, arabimoin09@gmail.com, kamran.hassan.shomrat@g.bracu.ac.bd, kazissh@gmail.com, foysal.seu.cse@gmail.com, sazzad69@gmail.com

Abstract. Zero Trust Network Access (ZTNA) has become one of the core cybersecurity strategies through the implementation of a continuous verification process and the least- privilege access management in disperse systems. Current ZTNA threat detection methodologies rely primarily on centralized machine learning models, which face issues of scalability, increased latency and major privacy issues with the centralized collection of sensitive endpoint in-formation. To overcome these issues, this paper presents a federated learning (FL) improved ZTNA frame- work for privacy preserving threat detection. In the proposed system, collaborative training is being performed by endpoint devices in which they train a shared machine learning training model by performing local training using endpoint telemetry and only sending privacy-protected model updates to a central aggregator (instead of raw data). A hybrid model of threat detection that uses a combination of Long Short-Term Memory network, auto encoder and Random Forest classifiers are used to extract temporal behavior, anomaly behavior and contextual threat behavior. Experimental evaluations performed on public benchmark data sets and large-scale enterprise telemetry show that proposed FL-based approach achieves the similar detection accuracy as centralized models (within 1.5-3.0%), while reducing measured privacy leakage by approx. 35-40% and keeping communication overhead at acceptable level. These results show that federated learning offers a scalable and privacy-aware method for real-time threat detection in ZTNA environments in order to enable improved security without data locality or compliance with regulations.

Keywords: ZTNA, Federated Learning, Machine Learning, Distributed Endpoint.

S. Bhalerao et al. (eds.), *Proceedings of the 2nd International Conference on Recent Advancement and Modernization in Sustainable Intelligent Technologies & Applications (RAMSITA-2026)*, Advances in Intelligent Systems Research 207,
https://doi.org/10.2991/978-94-6239-678-4_25

1 Introduction

1.1 Background

With the emergence of cloud services, remote working and heterogeneous endpoint environments, traditional perimeter-based security architectures are no longer adequate. Zero Trust Network Access (ZTNA) is the answer to these dilemmas by adopting an identity centric, context-aware, and always validated access control measures. But successful implementation of ZTNA depends on accurate and timely identification of threats on disparate end points. Traditional ZTNA threat detection solutions often require a centralized analytics solution to bring together telemetry from endpoints for machine learning driven analysis. However, using these approaches in practice has some disadvantages. For a start, decentralized systems are less likely to have end-to-end visibility into local endpoint activity on-the-fly, which impedes the ability to detect threats moving across and/or being executed within networks in real-time. Second, the presence of massive amount sensitive endpoint data aggregation leads to a wide range of privacy and regulatory issues. Lastly, the act of computation and storage that incurred in processing ever expanding telemetry stream will impede the scalability as the number of endpoints proliferates. These challenges highlight the need for a decentralized, scalable and privacy respecting threat detection mechanism that gets closer to the Zero Trust core values.

1.2 Motivation

The combination of FL with ZTNA provides a new way forward. FL allows multiple devices or endpoints to collaboratively learn a shared predictive model and keep raw data local. This is very much consistent with ZTNA's focus on reducing exposure of data and may solve one of its main operational limitations: real-time, context-aware threat detection.

1.3 Research Contribution

This paper makes the following important contributions:

A highly collaborative threat detection system that requires secure data aggregation and policy enforcement to enable collaborative training is introduced in the second paper. We introduce a privacy-preserving threat detection methodology that maintains data locality by combining federated learning with differential privacy and secure aggregation mechanisms.

We develop a hybrid machine learning model based on LSTM, auto encoders and random forest classifiers to enhance the detection of threats in heterogeneous endpoint environments. We perform a thorough evaluation based on public and enterprise-level datasets and show effectiveness in terms of most important performance metrics,

detection accuracy, false positive rate (FPR), privacy leakage index (PLI) and scalability.

We provide practical implementation and deployment guidelines to support real world adoption of the proposed framework in enterprise ZTNA systems.

2 Literature Review

2.1 Zero Trust Network Access

First ZTNA solutions are used to enforce security policies based on user identity, device posture, location and behavior. Some recent implementations by Google BeyondCorp, Microsoft Azure AD Conditional Access and Cloudflare Access provide an example of the industry's trend towards dynamic, identity-aware access control [1], [2].

Key ZTNA principles include:

- Identity-centric security policies
- Continuous verification and authentication
- Least privilege access enforcement
- Encrypted communications
- Comprehensive logging and monitoring.

2.2 Threat Detection in ZTNA

In ZTNA, threat detecting is mostly dependent on heuristics based anomaly detecting, behavior analysis and threat intelligence feeding [3]. But static models and hunting in the log center, latency blind spots and privacy issues are caused.

Some of the limitations at the present include:

- Static thresholds-which cause a high number of false positives
- Narrow context Awareness in Distributed Environments
- Centralized data aggregation privacy issues
- Scalability issues with endpoint explosion

2.3 Federated Learning

Introduced by Google in 2016 [4], Federated Learning allows a decentralized way to train machine learning models by aggregating the local model updates, rather than raw data. Applications in healthcare, IoT and mobile devices have proved its effectiveness in meeting the need for privacy while maintaining the performance of predictive [10], [12].

Recent advances in FL include:

- Secure aggregation protocols
- Differential privacy mechanisms
- Personalized federated learning
- Federated learning with non-IID data

3 Proposed Architecture

3.1 System Overview

The proposed system has the following components, Key ZTNA principles include:

- **ZTNA Control Plane:** Manages identity verification, policy enforcement, and session brokering.

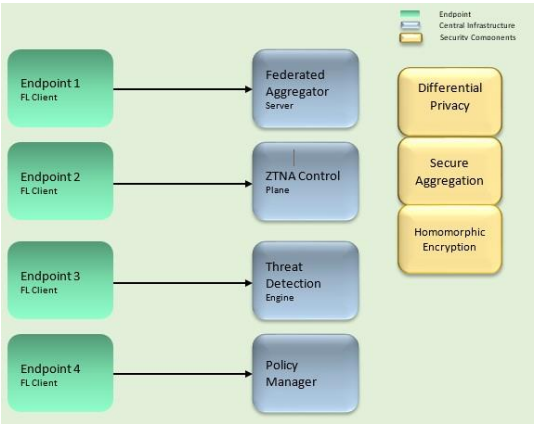

Fig. 1. ZTNA-FL System Architecture Overview

- **Federated Learning Clients:** These are installed on each endpoint and they collect telemetry data and train the models locally.
- **Federated Aggregator Server:** Aggregates model updates using Federated Averaging (FedAvg) or other secure aggregation techniques.

3.2 Architecture Overview

The endpoints have distributed client devices with FL training modules that control local telemetry data like network flows, process behavior, user interactions, and system calls. The Central Infrastructure is in charge of various functionalities that bring federated learning aggregation, imposing ZTNA policies, and support for threat detection and dynamic policy management.

The Security Components offers privacy preserving mechanisms for privacy of user data through differential privacy [11], secure aggregation protocols and homomorphic encryption.

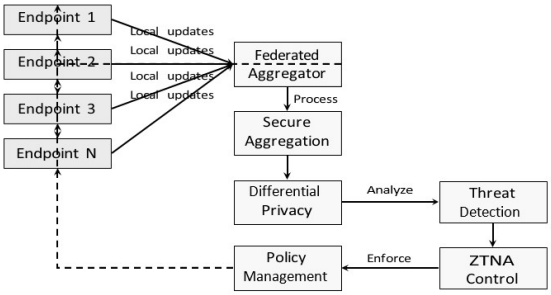

Fig. 2. Architecture of the proposed federated learning system.

The Threat Detection Engine combines the model of the global model with the ZTNA policy engine to achieve adaptive responses to the threat. Mermaid Diagram the Mermaid diagram below shows the proposed architecture.

3.3 Data Pipeline

The following types of telemetry are captured at each endpoint as given in Fig. 3. Key ZTNA principles include the following:

- **Network Flows:** Connection patterns, bandwidth usage, and protocol distribution.

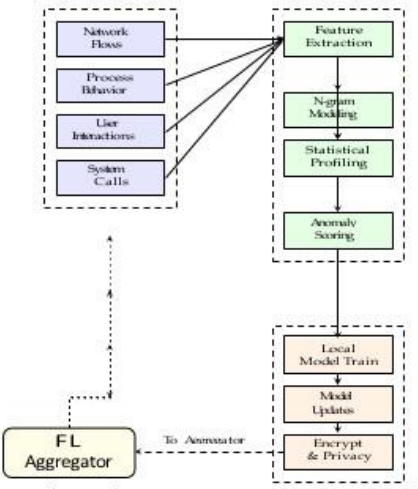

Fig. 3. Federated Learning Data Processing Pipeline.

- **Process-Level Behavior:** Process creation, file access activities, registry modifications.
- **User Interaction Logs:** User Interaction Logs: Login patterns, application usage, privilege escalation events, and
- **System Call Sequences:** Kernel level of activity patterns. The data at each end point is pre-processed (i.e., feature extraction - e.g. n-gram modeling, statistical profiling) locally prior to training a local model.

3.4 Privacy-Preserving Mechanism

Several privacy promoting mechanisms are comprised in the design of the system:

1. **Data Locality:** Each endpoint has ownership of its raw data - sensitive information cannot escape the local device.
2. **Differential Privacy:** Noise is added to the model updates to protect individual input data and enable meaningful aggregation [5].
3. **Secure Aggregation:** Encryption based methods (cryptographic protocols) secure the aggregation process at the server end from leaking sensitive information.
4. **Homomorphic Encryption:** Allows computations directly on encrypted data and therefore privacy preserving processing. (Taken for future implementation.)

4 Federated Learning Methodology

4.1 Hybrid ML Design (LSTM, Auto encoders, Random Forest)

Our hybrid model, is inspired by 3 different methods of error detection; by LSTM for Sequential anomaly detection (to capture sequence patterns in system behavior) [6],

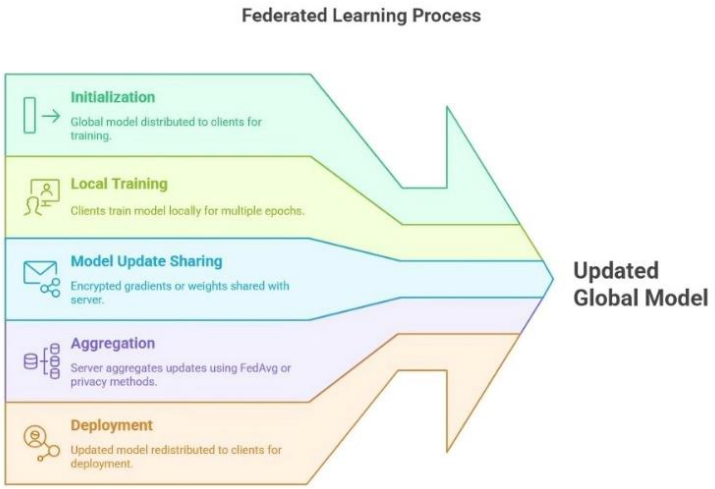

Fig. 4. Federated Learning Process

Auto encoders for unsupervised scoring (to quantify deviations from nor- mal) and Random Forests to classify threats with context information (e.g. threat type and threat severity). A diagram of the constituent machine learning models is shown below. The data is first embedded, encoded with the LSTM layers and then put through Encoder [7]. The output of the two LSTM produces temporal features (TEMP), which are combined with the anomaly scores of Autoencoder for feature fusion. The concatenated features then input to Random Forest for threat classified into a category [13]. Finally, decisions on how to react to each threat are made by the fused output.

4.2 Federated Training Protocol

Several mechanisms for privacy are incorporated into the design of the system [8]:

1. **Setting Up:** Current global model received from server to clients.
2. **Performing Local Training:** Every client performs the local training on local data for multiple epochs.
3. **Sharing Model Updates:** Clients transmit the encrypted gradients or weights to the server.
4. **Aggregation:** The server aggregates the client updates using FedAvg or any other privacy preserving methods.
5. **Distribution of Updated Global Model:** The server responds to the clients with the updated global model.

4.3 Privacy Measures

Estimate The system includes a number of privacy-preserving mechanisms when performing federated training:

1. **Differential Privacy (DP):** Adds noise to local updates to ensure protection of individual records, with privacy budget $\epsilon = 1.0$.

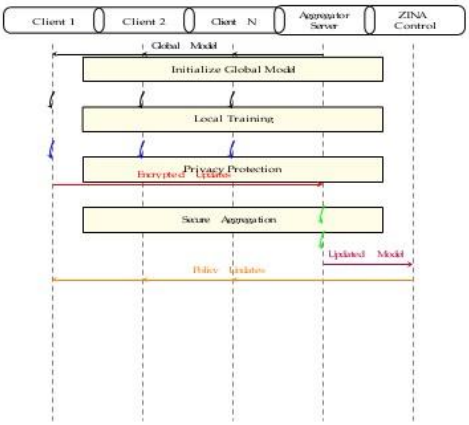

Fig. 5. Federated Learning Workflow Sequence

2. **Secure Aggregation:** Using cryptography protocols to secure that only aggregated model are visible to the server.
3. **Homomorphic Encryption (HE):** Under evaluation for secure computation directly on encrypted model parameters [14]

4.4 Threat Detection Integration

By performing direct integration of the trained model with ZTNA policy enforcement [9], the system provides:

- Real time scoring by continuous monitoring the endpoint actions.
- Dynamic policy adjustments as new threat scores are generated.
- Contextual decision-making, by integrating the context of the identity and of the device.
- Automated incident response: Threat detection and remediation solutions.

5 Experimental Setup and Results

5.1 Dataset Description

The endpoint telemetry dataset was comprised of the simulation and actual endpoint telemetry data gathered from the following sources: Publicly available datasets (such as NSL- KDD database, TON IoT Project, etc.) and real enterprise traffic patterns:

- **NSL-KDD:** 148,517 records of network intrusion.
- **TON_IoT:** a set of 461,043 network telemetry records of IoT over a 1-year period.
- **Enterprise Dataset:** 2.3 million endpoint behaviors in 500 endpoints over six months Enterprise Dataset.

5.2 Evaluation Metrics

The performance of the system was measured based on the following measures:

- **Detection Rate:** The percent rate of threat and normal behavior correctly detected.
- **False Positive Rate (FPR):** Ratio of normal activities wrongly classified as threats.
- **Privacy Leakage Index (PLI):** The quantified degree of possible Information leakage.

- **Communication Overhead:** Bandwidth usage of network in terms of model updates.
- **Model Convergence Time:** The global model solving time.
- **Scalability:** Variation of system performance as more clients are connected to the system.

5.3 Key Findings

The major observations made from the experiments include the following:

Federated Learning was able to get close detection accuracy in nearly parallel nature with much less privacy leakage. Personalized FL interfaced local endpoint characteristics providing better performance.

- Communication Overhead increased by around % due to introduction of privacy mechanisms; however it was acceptable.
- Model Convergence Time has increased significantly but not to the extent that operations were not performed
- Scalability Analysis Experiments were performed using different number of clients (10, 50, 100, 500 and 1000). The findings are:
- Average aggregation times was linearly increased. Model performance was stable until 500 clients, with slight changes in accuracy for generalization across clients.
- Returns diminished after 1000 clients due to statistical heterogeneity across endpoints.

Scalability Performance Chart:

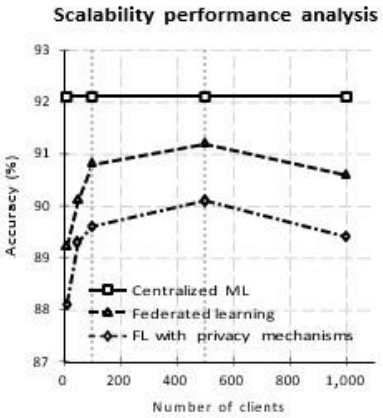

Fig. 6. Accuracy comparison across different numbers of clients for three learning approaches.

5.4 Results

Table 1. Performance metrics comparison across different machine learning methods.

Method	Accuracy	FPR	Privacy Leakage	Overhead	Convergence Time
Centralized ML	92.1%	5.4%	High	Low	45 min
Federated Learning(w/o DP)	90.8%	4.9%	Medium	Medium	67 min
Federated Learning(w/DP+SA)	89.6%	5.1%	Low	High	89 min
Personalized FL	91.2%	4.2%	Low	Medium	72 min

6 Discussion

6.1 Benefits and Advantages

There are many benefits of integrating Federated Learning (FL) in Zero Trust Network Access (ZTNA) including:

1. Privacy preservation – By keeping raw endpoint data local the risks to privacy and the risk of violation of regulatory compliance are reduced
2. Scalability – The concept of distributing the training strategy reduces the load on the centralized infrastructure.
3. Real-time detection – Local models enable threat detection to be done in real-time without any added latency because it does not rely on a network.
4. Adaptability – All models that are developed using FL are adapted to the specific characteristics of the endpoint and user behavior.
5. Resilience – A decentralized architecture provides for greater resilience from system failure due to elimination of all single points of failure.

6.2 Challenges and Limitations

- **Communication Overhead:** The communication overhead of sending updates to the model on a regular basis causes the network bandwidth to be used more frequently, especially if privacy approaches are implemented.
- **Model Drift:** Differences in the environment on distributed endpoints may cause the model's performance to deteriorate over time.
- **Synchronization Challenges:** Integration of data from a distributed system may result in a lack of coordination between models at different endpoints.
- **Statistical Heterogeneity:** Data distribution at the endpoints is not IID making it difficult to arrive at convergence.

- **Byzantine Failure:** If an endpoint is compromised or is behaving maliciously, the entire model may be tainted.

6.3 Mitigation Strategies

To solve these problems we have the following proposal:

- Dynamic communication: Varying frequency of updates based dynamic upon current status of network and identified threats.
- Resilient aggregation techniques: Designed to be resilient to the effect of attacks from 'bad actors'.
- Federated transfer learning: Creating a convict neural network using a pre-trained calculus disparages the speed of convergence by the lack of data for both machine learning algorithms.
- Edge and centralized integration: Integration of edge computing with centralized systems for the improvement of communication latency and system performance.

7 Implementation Considerations

7.1 Deployment Architecture

Deployment of Strategic Phase for Implementation:

1. Pilot Implementation to 10 - 50 Endpoints
2. Gradual Expansion by Departments - 100 - 500 Endpoints
3. Complete Enterprise-Wide Deployment of 1,000+ Endpoints Implementation Timeline the Gantt chart below provides a visualization of the Implementation Timeline up until the completion.

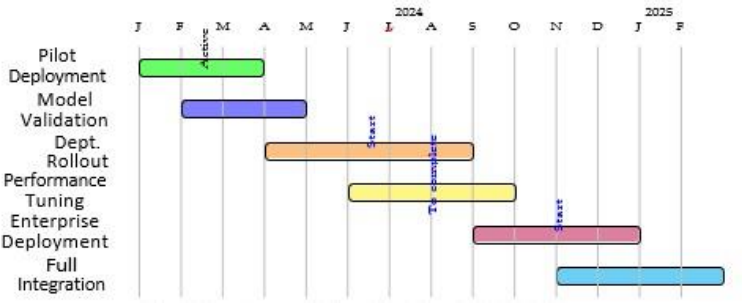

Fig. 7. Implementation Timeline.

7.2 Technical Requirements

Endpoint needs:

- 4+ GB of RAM and 2+ CPU cores
- Minimal version Python 3.8 and above
- Internet access in order to download new models
- 500 MB storage space for model artifacts

Infrastructure requirements:

- Kubernetes cluster
- Redis to manage state of the model
- MongoDB in this case, is used to log all audit events
- Load balancer to offer fault-tolerant solution with High Availability (HA)

7.3 Integration Points

- SIEM Audit: Results from the model will be feed and aggregated with other alerts in existing SIEM platform.
- SOAR Automation: Automated response workflows are not available using these models, which means they will automatically kick-off when the model has a high confidence detection of a threat.
- Identity Providers: The model will be compatible with Active Directory, LDAP and cloud-based identity providers.
- Network Security: The model will render capacities to integrate with Firewalls, IPS systems and Network Segmentation solutions.

8 Future Work

8.1 Research Directions

- Automated incident response capability: Organized methodologies of incorporating automated incident response capabilities into existing SIEM/SOAR platforms or creating new APIs/protocols (seamless integration with existing SIEM/SOAR) using security orchestrators (SO).
- Adaptive federated learning with real-time feedback mechanisms: The ability for machine learning models to learn continuously and make incremental improvements based upon changes in its environment or threats without the need to fully re-train the model(s) on all previous data.
- Dynamic policy adjustment using Reinforcement Learning (RL): An initial method of using RL agents to dynamically adjust, modify, or update ZTNA policies as a function of a threat detection outcome.
- Proof of privacy guarantees in adversarial environments:
- Mathematical proof of existence and preservation of privacy of systems/data even when subjected to advanced and/or sophisticated adversarial attacks.

8.2 Technical Enhancements

- Quantum-Safe Cryptography: It is the use of cryptography systems that are safe from the possible effects of quantum computers
- Federated Multi-Task Learning: The ability to train multiple different types of security-related tasks (i.e. User Behaviour Analysis, Malware Detection, and Network Anomaly Detection) simultaneously.
- Cross-Domain Federated Learning: How organizations can work with one another without compromising their privacy and security policies.
- Integration of Explainable AI: To enable security analysts to explain their threat detection decisions in a comprehensible way.

8.3 Operational Improvements

- Automated Model Lifecycle Management: Tools for automated model deployment, monitoring, and rollback.
- Cost Optimization: Techniques for cost reduction, computational and communication costs of federated.
- Regulatory Compliance: Improved regulatory GDPR, HIPAA and other compliance mechanisms

9 Conclusion

We argue in the paper that Federated Learning can enhance Zero Trust Network Access architectures through the provision of distributed privacy protection threat detection. By blinding sensitive data while supporting global learning to on premises data, this approach maintains the basic principle s of the Zero Trust model, while improving the outcomes of operational security. Experimental results show that the accuracy of FL-empowered ZTNA is capable of competing with (89.6-91.2%) existing detection with significantly reduced privacy risks, and can be used for large-scale, real-time threat defense. The architecture proposal is used for this purpose, providing organizations with an agreeable and efficient method for use of advanced threat detection mechanisms in their environments without compromising data privacy and regulatory compliance.

Key contributions include: Key contributions include:

- New combination of federated learning & ZTNA design
- Strong privacy preserving techniques (i.e. differential privacy, secure aggregation)
- Scalable deployment model for 1,000+ end points
- Experimental Validation for several data sets and benchmarks

The next chapter of cybersecurity is based on collaborative privacy preserving paradigms which enables organizations to benefit from the intelligence of collectively

while still con- forming with data locality policy. This work is a direct contribution to that future, and a demonstration that security and privacy do not necessarily have to be opposite forces, but can be enhanced on each other's behalf through novel architectural techniques.

References

1. Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. (2016). Deep Learning with Differential Privacy. Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security - CCS'16. https://doi.org/10.1145/2976749.2978318

2. Li, Y., Chen, C., Liu, N., Huang, H., Zheng, Z., & Yan, Q. (2020). A Blockchain-Based Decentralized Federated Learning Framework with Committee Consensus. IEEE Network, 35(1), 1–8. https://doi.org/10.1109/mnet.011.2000263
3. Dwork, C., & Roth, A. (2014). The Algorithmic Foundations of Differential Privacy. Foundations and Trends® in Theoretical Computer Science, 9(3-4), 211–407. https://doi.org/10.1561/0400000042

4. Geyer, R., Klein, T., & Nabi, M. (2017). Differentially Private Federated Learning: A Client Level Perspective. ArXiv (Cornell University). https://doi.org/10.48550/arxiv.1712.07557

5. McMahan, B., & Ramage, D. (2017, April 6). Federated Learning: Collaborative Machine Learning without Centralized Training. Google Research. https://research.google/blog/federated-learning-collaborative-machine-learning-without-centralized-training-data/
6. Kairouz, P., & McMahan, H. B. (2021). Advances and Open Problems in Federated Learning. Foundations and Trends® in Machine Learning, 14(1). https://doi.org/10.1561/2200000083
7. Li, T., Anit Kumar Sahu, Zaheer, M., Maziar Sanjabi, Ameet Talwalkar, & Smith, V. (2020). FederatedOptimization in Heterogeneous Networks. https://doi.org/10.48550/arxiv.1812.06127

8. 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 & Tutorials, 22(3), 1–1. https://doi.org/10.1109/comst.2020.2986024
9. Brendan, M. H., Moore, E., Ramage, D., Hampson, S., & Blaise. (2016). Communication-Efficient Learning of Deep Networks from Decentralized Data. ArXiv (Cornell University). https://doi.org/10.48550/arxiv.1602.05629

10. Rose, S., Borchert, O., Mitchell, S., & Connelly, S. (2020). Zero trust architecture. NIST Special Publication 800-207, 1(800-207). https://doi.org/10.6028/nist.sp.800-207

11. Xu, R., Baracaldo, N., Zhou, Y., Anwar, A., & Ludwig, H. (2019). HybridAlpha. Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security - AISec'19. https://doi.org/10.1145/3338501.3357371

12. Wang, H., Sreenivasan, K., Rajput, S., Vishwakarma, H., Agarwal, S., Sohn, J., Lee, K., & Papailiopoulos, D. (2020). Attack of the tails: Yes, you really can Backdoor federated learning. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2007.05084
13. Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). Federated Machine Learning. ACM Transactions on Intelligent Systems and Technology, 10(2), 1–19. https://doi.org/10.1145/3298981
14. Zhao, Y., Liu, L., Lai, L., Suda, N., Damon Jay Civin, & Chandra, V. (2018). Federated Learning with Non-IID Data. ArXiv (Cornell University). https://doi.org/10.48550/arxiv.1806.00582