

Decentralized Machine Learning Models to Preserve Data Privacy in Intrusion Detection Systems (IDS)

Cletus A. Sieh Jr, Edwin Hope Beh, Daniel Chibamb, Aditya Dayal Tyagi

Department of Computer Science and Application Sharda University, Greater Noida, India

DOI: <https://doi.org/10.51583/IJLTEMAS.2025.140400074>

Abstract: As Technology keeps advancing rapidly, the risk of data leakage is on the increase. centralized servers are often attacked in order to control the flow of data and divert the actual network direction and breach data privacy. Data can be manipulated on a higher scale once the centralized systems have been attacked and the existing IDS fails to recognize a breach in the network server.

In order to have a more robust privacy of data and accurate functioning of the IDS through ML and DL approaches to curb cyber-attacks and network attacks, the use of a decentralized ML model would be effective. This approach aims at introducing a decentralized network where data will be shared in protective nodes and each node will have an existing IDS that will not be exhausted with a huge workload.

Federated Learning, which is a sub domain of DML, would offer solutions by enhancing local models on edge devices without the need and reliance of a centralized server [16].

This approach will reduce the risk of data exposure by ensuring that data stays on the source device and can only be accessed and controlled from that source only.

The outcome of this research would vividly outline the effectiveness of decentralization of data and the efficiency of IDS on specialized points of the data network to protect the exchange and control of data without a maximum risk of data leak.

Key words: Intrusion Detection System (IDS), Anomaly, Decentralization, Central server, Data Privacy

I. Introduction

IDS is normally used in a centralized system that provides overall detection mechanism for intrusion. This method has a major weakness, if intrusion is done outside of the main controlled system, there's a high risk of that intrusion being undetected by the network. Since the IDS is situated at the central point of the network, other nodes of the network face threat of being intruded and manipulated with the lack of IDS available at their end of the network. Local nodes within the network relies on the central authority of the network to detect anomalies rather than being able to do so by themselves. This pose a significant risk for the system where intrusions may be seen as credible since they originate from an authentic node within the network. We have identified some crucial weaknesses in using IDS at the central point of a network rather than in various independent nodes of the network. Single point of failure is one of the most noticeable weakness of using IDS only at the central point of the network. By using at only a sing point of the network, a failure at that point will cause for failure and disruption throughout that entire system but if a distributed IDS is setup, it would allow redundancy within the system where in failure at one point of the system because of anomalies at that point won't entirely affect the system. Limited visibility is another key weakness of using IDS at the central point of the system. A centralized IDS may not have full of complete visibility into specific network segments, and without local IDS on each of the individual nodes of the network or system[5], some intrusions may go undetected especially in those attacks are designed to only target that individual node. The central IDS may only be on the alert for anomalies in individuals nodes if IDS are deployed that those nodes. Network Congestion may arise in a system with the use of a centralized IDS. These centralized IDS are responsible to analyze all traffic logs and if the traffic is of high volume, or once there are large number of devices on that network, it will be a serious issue. This is where decentralized IDS comes into play which would enable the system to alleviate congestion by processing data locally and sending only relevant alerts or only summaries to the central system. The central system will now have enough information about the anomaly, its point of entry and the system will easily get rid of those intrusions.

We aim to design the network in a way that each node has the ability to detect anomalies from their end which will significantly reduce the risk of intrusion in the network. The goal is to have decentralized architectures like block chain or peer-to-peer networks to detect and respond to intrusions. The implementation of various techniques such as Decentralized Detection Mechanism, Collaboration and coordination, Scalability of the IDS and Resilience in the networks can ensure that a system is more robust in detecting anomalies especially from their point of entry before getting to the central network[1].

To build a more robust system that will be effective in detecting anomalies, we will deploy decentralized machine learning(ML) which will help preserve data by ensuring that sensitive data is not centralized or exposed during the detection process.

Here are some key aspects of using decentralized machine learning process to preserve data privacy in IDS:

Data localization - by ensuring that data is kept and processed on local nodes, we can prevent sensitive data from being exposed or manipulated by third party or intruders, thus reducing the risks associated with centralized storage and transmission[4]. This technique if implemented would protect data privacy effectively. It will reduce the risk of data leakage and subsequently enabling a balanced system.

Federated Learning - This allows for multiple nodes to train ML models locally on their own data without transmitting sensitive data to a central server. The deployment of this technique would enable that only model updates are shared, and not necessarily the raw data itself which ensures that local data remains private and the risk is reduced.

Differential Privacy: this technique can be integrated into decentralized ML to ensure that the shared model updates do not inadvertently leak information about the individual data point. It encrypts data by adding noise to the updates to protect data from being influenced at any particular data item on the final model.

Literature review

IDS - Intrusion detection system is basically a network security technology built to detect anomalies and vulnerabilities against a computer system or network. IDS monitors traffic and reports to the system administrator for further actions. IDS alerts the system of possible threats based on certain known criteria. IDS analyze the system for suspicious activities and sends warning to the system. Centralized IDS refers to a system where all actions of monitoring, detecting anomalies and the decision making processes are done at a centralized point in a single entity like a server or a security management system. Centralized IDS deals with the aggregation of data from various sources on a network as well as logs. In contrast to centralized IDS, Decentralize IDS will enable extra layer of protection of data by enabling the uniqueness of data and protection from its source. Sensitive data won't be shared in a decentralized system.

Limitations of centralized IDS:

Single point failure - the central server becomes a single point of failure once the server becomes compromised, or crash or experience downtime. The entire system is affected by any breach and it becomes ineffective, leaving the network and system vulnerable to attacks. Due to the fact that all the data within the network is routed to a central server for processing, the central server becomes a single point of failure. The central server may become overwhelm with excess processes and might crash, leaving the entire network at a failed state.

Scalability issues - as number of monitored devices increase due to growth in the network, the central server may experience difficulties in processing large amounts of data in real-time. This can lead to a delay in detecting intrusions and may significantly upgrade to the server infrastructure in order to get the network functioning properly again. In this downtime, there will be significant risk to sensitive data.

High network traffic - within a centralized IDS, data is send and collected from multiple devices to the central server. These data may range from network traffic, logs and event data. In large networks, this can create a significant network congestion, increased bandwidth usage and potential performance bottlenecks.

Latency in detection response - latency arise when all data is send to the central server for analysis. And even if the server process those data quickly, transmission time at endpoints and the central server can delay the detection of intrusions, making the system less effective in real-time.

Privacy concerns - one of the most significant limitation of using Centralized IDS is the threat of data leakage due to less protection outside of the firewall of the central server. Centralized IDS collects and analyze data from various devices, which may be sensitive data, confidential information and etc. This is a very likely threat to the privacy of data, especially in areas like finance and health care. Once the central server gets penetrated, all those sensitive data becomes exposed which pose serious risk to the owners of those data.

Vulnerable to distributed attacks - A centralized IDS may encounter vulnerability like Distribute Denial of Service(D Dos),where malicious traffic is sent from multiple sources to overwhelm the central server. As a result, the system will become overloaded which will cause delays and failure of the IDS. Once the system fails, all data becomes vulnerable and at risk of being compromised.

Limited Flexibility - A centralized IDS system may not be as flexible or adaptable to different devices or network segments since collection of data is aggregated and processed centrally. Each different device or endpoints may have its own unique needs, and the centralized system might struggle to effectively and efficiently analyze diverse data sources.

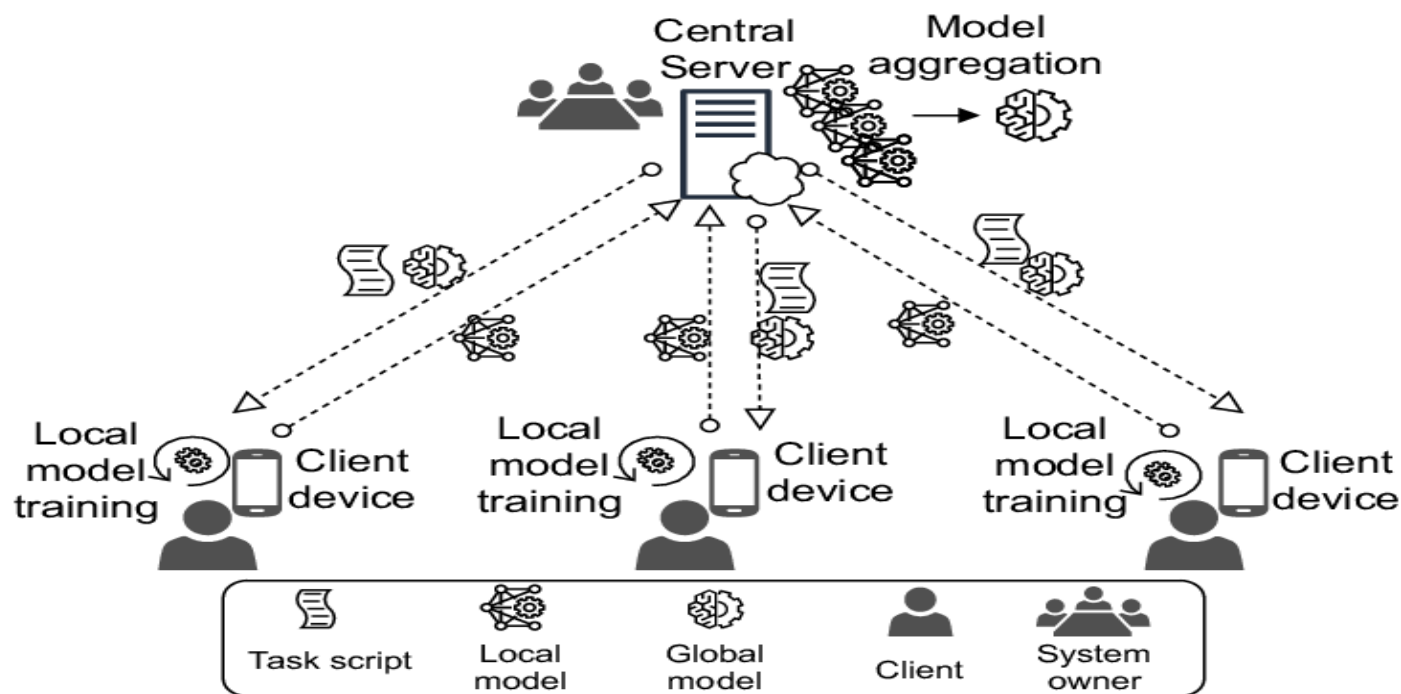
Lack of Anonymity - In a centralized server system, data traffic and the entire data identity are monitored and often can be easily traced back to users or specific devices. This lack of anonymity can be problematic for users who wish to keep their activities private or for organizations aiming to avoid risks associated with revealing user information to third party and intruders.

Decentralized Machine learning methods in IDS: Methods such as Federated learning(FL) and Distributed Learning, have emerged as potential solutions to the privacy challenges associated with the use of centralized IDS systems. These methods allow

learning and data analysis to occur at the edge and on local devices or at distributed nodes of the network without the need for sensitive data to be transferred to a central location.

Federated Learning - this is a decentralization approach to machine learning where the model is trained locally on the individual device or node of a network that generates the data. This allows for only model updates to be shared with the central server and not the raw data. Sensitive data never leaves the device and the privacy is preserved. Since raw data is not exchange in this model, the risk of exposing sensitive data is at a minimal. The central server only aggregates the weight of the model and updates but not the actual personal or private data[17]. The major challenge of using Federated Learning model is that communication overhead and heterogeneity of data across devices can complicate the learning process.

This figure depict an overview of federated learning(fig 1)



Distributed Learning - This model involves training machine learning model across multiple nodes in a network. The learning is divided across devices and they only share partial information without revealing the actual and complete data set. It keeps the data distributed across different nodes or devices, reducing the need for centralized data collection and processing. This decentralized model enhances data privacy. Although it is relatively safer, Distributed Learning might still require secure mechanisms to aggregate the information and may be prone to attacks such as model poisoning, where malicious updates are sent to intrude and manipulate the network.

Privacy-preserving Techniques

To make sure that the privacy of data is further enhanced in a decentralized IDS system, privacy preservation techniques can be applied.

Homomorphic encryption - allows computation on encrypted data without the need to decrypt it. This technique ensures that sensitive data can be processed while remaining encrypted which aids data integrity and makes sure that data is never exposed. In a decentralized IDS, the data can be encrypted before being sent for model training and the computations which involves anomaly detection can be performed on the encrypted data[11]. Only the result of the data is sent back, therefore maintaining its privacy. As a challenge, deploying Homomorphic encryption is expensive and it can cause a significant overhead to the network especially for large data sets or real-time IDS applications.

Secure Multiparty computation - Secure Multiparty computation enables multiple parties to collaboratively compute a function over their inputs while keeping those inputs private. In a Decentralized IDS system, multiple nodes can train and intrusion detection model collaboratively while keeping their respective data private. SMPC ensures that each party contributes to the model without having to expose its data to the other. SMPC faces a significant problem in its deployment. It requires significant computational resources and may not scale well to large number of parties. In addition to that, the coordination and communication between parties can introduce delays and complexity.

Differential Privacy - ensures that the inclusion and exclusion of a single data point does not significantly affect the output of a computation, thus it provides a guarantee and strong privacy of data. It involves adding noise to the results of analysis to mask

individual data points and adds extra layer of protection for the data. This technique can be applied in decentralized IDS systems on aggregated data and model updates to prevent the leakage of sensitive information about users or events and their sources. However, the noise applied to data can reduce the accuracy of the models or increase the difficulties of detecting certain types of attacks.

Previous use cases of Decentralized models in cyber security and IDS

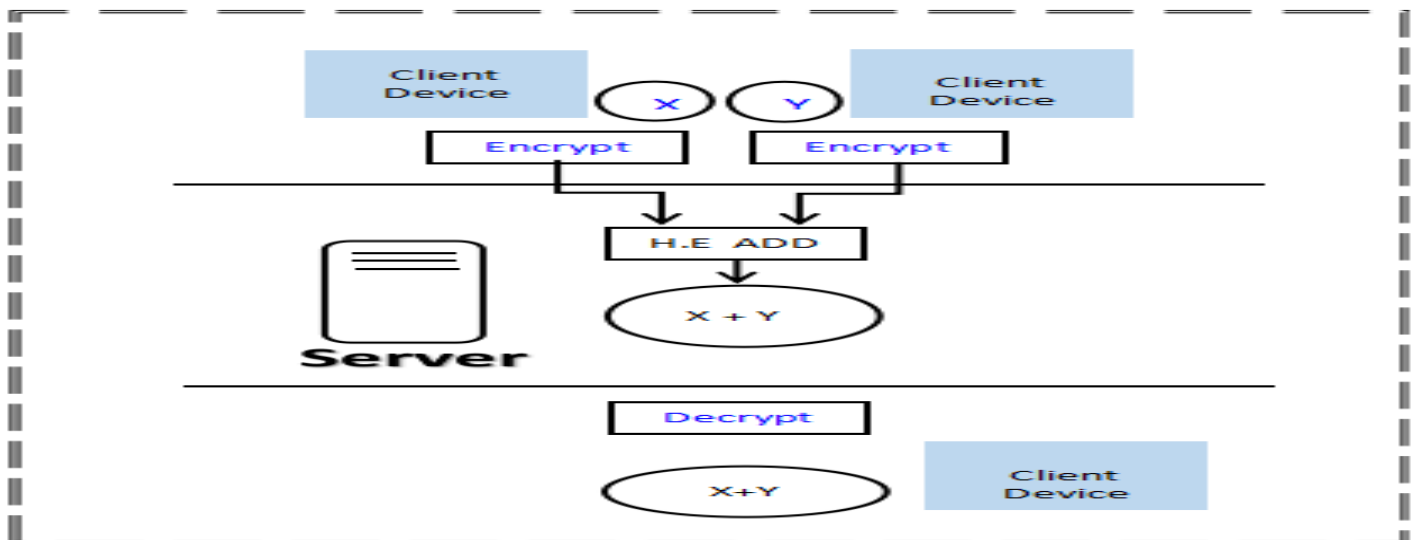
Federated Learning in Malware Detection: in mobile cyber-security, federated learning has been applied to train models for detecting Malware without uploading raw data to the central server. Each device trains a model on its local data and only shares aggregated model updates. This improves data privacy while enabling global detection capabilities.

Challenge - the major concern with this technique is to ensure model updates are secured and capable of preventing adversaries from poisoning the model updates.

Privacy-Preserving Intrusion Detection with Homomorphic Encryption: Researches have been carried out involving the use of a Homomorphic encryption in conjunction with IDS to ensure that sensitive data is encrypted during analysis by the central server of a network. This approach helps maintain the confidentiality of data even when algorithms are running on encrypted data.

Challenge - the high computational cost associated with Homomorphic encryption can reduce the real-time performance of IDS systems.

Fig. 2



(Fig 2) Homomorphic encryption being used in this Network for data integrity and confidentiality.

Secure multiparty computation for collaborative IDS: SMPC has been deployed in financial institutions where collaborations are done among multiple organizations to detect common threats such as DDOS attacks, while keeping their eternal data private. Those collaborative organizations share model updates and threat intelligence without having to expose their sensitive data.

Challenge - the scalability of SMPC in large networks with many participants and the complexity of managing secure computations across multiple parties pose a significant challenge.

Differential privacy for data Anonymity in IDS:

Differential privacy has been applied to anonymize traffic data in IDS systems, ensuring that individual users' information can't be extracted even if the IDS analysis is being compromised. This is made possible by adding noise to the data or aggregated statistics, differential privacy ensures privacy while still allowing threat detection algorithms to function effectively.

Challenge - there is always a trade off between the degree of privacy and the accuracy of threat detection. High noise levels can degrade the performance of IDS systems.

II. Methodology

This methodology outlines the approach for designing a decentralized machine learning-based Intrusion Detection System (IDS) while ensuring data privacy and robust data security. The system utilizes **Federated Learning (FL)** as the core decentralized learning model and integrates privacy-preserving techniques such as **Differential Privacy (DP)**, **Secure Multi-Party Computation (SMPC)**, and **Homomorphic Encryption (HE)**. Additionally, it defines key **performance and privacy evaluation metrics** to assess the effectiveness of the IDS[17].

Model Selection: Federated Learning (FL)

Why deploy FL

Federated Learning (FL) is chosen as the decentralized model due to its numerous abilities which include :

Enabling multiple IDS nodes to train a shared model without exposing raw data.

Reducing data exposure risk by exchanging only model updates (gradients) instead of raw traffic data.

Providing a balance between decentralization and structured coordination, ensuring effective model convergence.

Supporting privacy-enhancing mechanisms that protect against adversarial threats.

Privacy-Preserving Techniques

In order to maintain data confidentiality and security, the following privacy-preserving techniques are integrated into the FL-based IDS:

Differential Privacy (DP)

Objective: Protect individual data points from being inferred by adding controlled noise to model updates before sharing them.

Implementation: Apply **Gaussian or Laplacian noise** to gradients before sending updates to the central FL server.

Benefit: Enhances resistance against **membership inference attacks**.

Secure Multi-Party Computation (SMPC)

Objective: Allow multiple IDS nodes to compute shared model updates securely without revealing individual data.

Implementation: Utilize **Cryptographic protocols** to enable secure aggregation of model updates.

Benefit: Prevents exposure of model parameters and training data to suspicious aggregators.

Homomorphic Encryption (HE)

Objective: Ensure encrypted computations on model updates without decrypting them.

Implementation: Apply **fully or partially Homomorphic encryption** schemes to gradients before transmission.

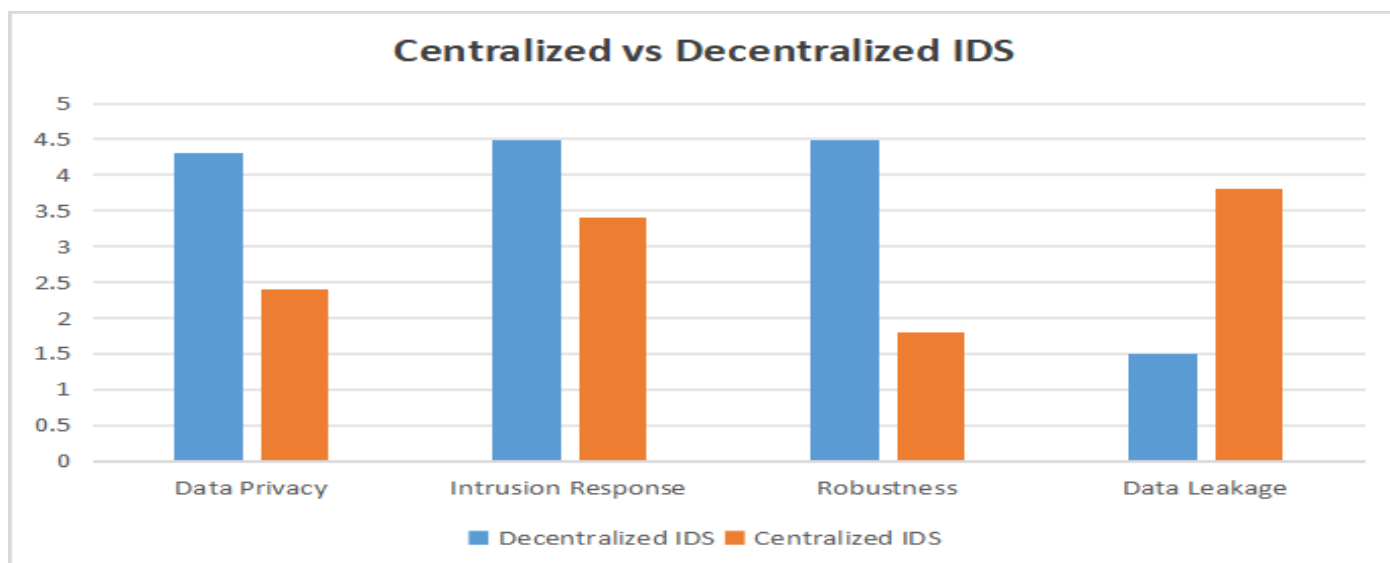
Benefit: Protects model updates from adversarial interception.

Federated Averaging with Local Differential Privacy

Objective: Ensure nodes only share encrypted or noised model updates.

Implementation: Each node applies **local differential privacy** before transmitting updates to the FL aggregator.

Benefit: Further reduces data leakage risk while preserving IDS accuracy.



(Fig3) depicts the contrast between using decentralized IDS and Centralized IDS

Evaluation of Metrics

To assess the IDS effectiveness and privacy, both **IDS performance metrics** and **privacy metrics** are considered.

IDS Performance Metrics

Detection Rate (DR) = (Correctly Detected Attacks) / (Total Attacks)

Evaluates the system's effectiveness in identifying intrusions.

Accuracy = $(TP + TN) / (TP + TN + FP + FN)$

Measures the overall correctness of intrusion classification.

Precision = $TP / (TP + FP)$

Assesses how many detected intrusions were actually true intrusions.

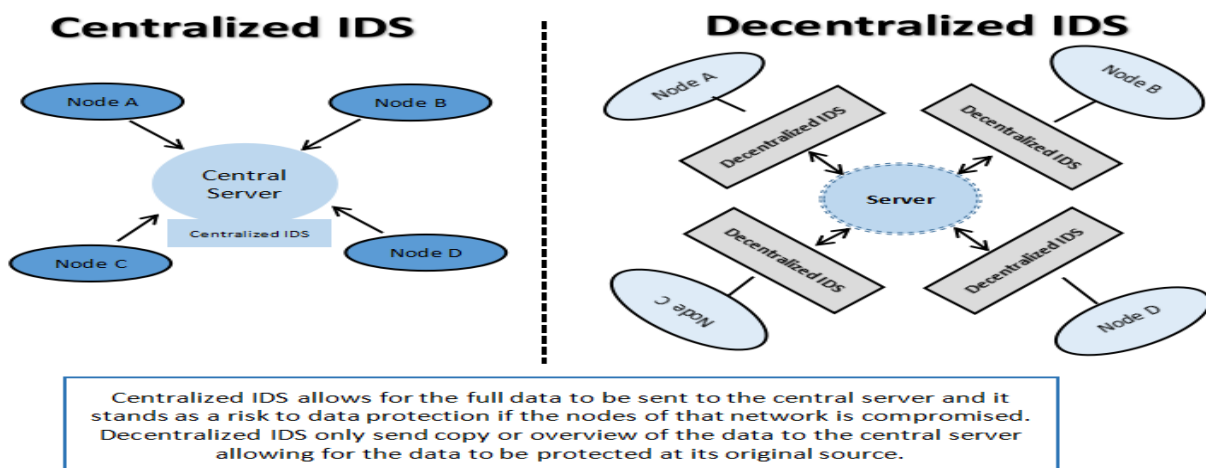
Recall (Sensitivity) = $TP / (TP + FN)$

Evaluates how well actual intrusions are detected.

Privacy Metrics

Data Leakage Risk

Measures the likelihood of sensitive information being inferred from model updates.



(Fig 4) illustrate the stack difference in data protection between centralized and decentralized IDS. Decentralized IDS ensures a better data protection.

Robustness to Membership Inference Attacks

Evaluates the system's resilience against adversaries trying to determine if specific data was used in training.

Model Robustness to Poisoning Attacks

Assesses the impact of adversarial data poisoning attempts on IDS performance.

Implementation Plan

Data Collection and Pre-processing

Use **benchmark IDS datasets** such as **CICIDS2017**, **NSL-KDD**, or **UNSW-NB15**.

Perform **feature selection**, **normalization**, and **encoding** to prepare the data-set for training.

Federated Learning Model Development

Implement an FL-based IDS using **TensorFlow Federated (TFF)** or **PySyft**.

Define a convolution **neural network (CNN)** or **recurrent neural network (RNN)** for traffic anomaly detection.

Train models across distributed nodes while integrating **privacy-preserving techniques**.

Privacy-Preserving Mechanisms Integration

Implement **differential privacy with noise injection** at the local model update stage.

Use **Secure Aggregation (SMPC)** to ensure privacy during model aggregation.

Apply **Homomorphic Encryption (HE)** to protect model gradients in transit.

Model Training and Evaluation

Train and evaluate the FL-based IDS with different privacy configurations.

Compare results against **centralized and non-private models**.

Assess privacy risks using membership inference and data leakage simulations.

III. Conclusion:

In this rapid growing technology world where data sensitivity is of a very high priority, we are keen to making sure to close most or all loop holes of data leakage. Deploying an IDS at the central point of the server is somehow protective but also very risky to the entire system. Once an intruder is aware that all data is at a central server, that portion of the network becomes the main target. And even if the central server is not breached, it might encounter some issues due to the overwhelming traffic targeted at the central server. The solution is quite simple, we can keep data secured at their various nodes in the network and send only reports to the central server for analysis. This way, if the central server is compromised, data will be safe at their various nodes and the rest of the network will remain functional. The deployment of a decentralized IDS will enable the extra safety of data sensitivity and avoid data leakage. Federated Learning makes this technique guaranteed, secured and risk free.

References:

1. A Decentralized Intrusion Detection System for Security of Generation Control Publisher: IEEE 2022 Siddhartha Deb Roy; Sanjoy Debbarma; Adnan Iqbal
2. Poster Abstract: Towards Scalable and Trustworthy Decentralized Collaborative Intrusion Detection System for IoT Publisher: IEEE Guntur Dharma Putra; Volkan Dedeoglu; Salil S Kanhere; Raja Jurdak
3. Securing Cyber-Physical Systems: A Decentralized Framework for Collaborative Intrusion Detection with Privacy Preservation Publisher: IEEE 2024 Zia Ul Islam Nasir; Adnan Iqbal; Hassaan Khaliq Qureshi
4. Kim, K.-J. Park and C. Lu, "A survey on network security for cyber-physical systems: From threats to resilient design", IEEE Commun. Surv. Tuts., vol. 24, no. 3, pp. 1534-1573, 2022.
5. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches", Trans. Emerg. Telecommun. Technol., vol. 32, no. 1, 2021.
6. A. Cheema, H. K. Qureshi, C. Chrysostomou and M. Lestas, "Utilizing block-chain for distributed machine learning based intrusion detection in Internet of Things", Proc. IEEE 16th Int. Conf. Distrib. Comput. Sensor Syst., pp. 429-435, 2020.
7. A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi and R. Ahmad, "CNN-LSTM: Hybrid deep neural network for network intrusion detection system", IEEE Access, vol. 10, pp. 99837-99849, 2022.
8. Zhao, Y. Yin, Y. Shi and Z. Xue, "Intelligent intrusion detection based on federated learning aided long short-term memory", Phys. Commun., vol. 42, 2020.
9. A review of applications in federated learning Author: Li Li a b, Yuxi Fan a, Mike Tse c, Kuo-Yi Lin 2020
10. Advances and Open Problems in Federated Learning © 2021 Peter Kairouz, H. Brendan McMahan, et al.
11. A survey on federated learning: challenges and applications Original Article Published: 11 November 2022
12. Federated Learning: Challenges, Methods, and Future Directions Publisher: IEEE 2020 Tian Li; Anit Kumar Sahu; Ameet Talwalkar; Virginia Smith
13. Enhancing Security with a Decentralized Intrusion Detection System for Sensor and Control Attacks J Khurana, SS, A Singla, GV Gaonkar... - 2024 IEEE 4th ..., 2024
14. Centralized and distributed intrusion detection for resource-constrained wireless SDN networks GAN Segura, A Chorti, CB Margi - IEEE Internet of Things ..., 2021
15. Tyagi, A. D., & Asawa, K. (2024). Influence Maximization in Social Network using Community Detection and Node Modularity. International Journal of Performability Engineering, 20(9).
16. Tomar, V., Sharma, S., Arora, S., & Tyagi, A. D. (2024, October). A Comprehensive Analysis of Techniques and Applications in Multimodal Deep Learning. In 2024 International Conference on Computing, Sciences and Communications (ICCS) (pp. 1-5). IEEE.
17. Tyagi, A. D., Garg, S., Sharma, S., Tomar, V., & Verma, K. (2024, November). Sarcasm Detection in X Data Using Node Embedding and Graph Convolutional Networks. In 2024 4th International Conference on Advancement in Electronics & Communication Engineering (AECE) (pp. 1336-1340). IEEE.