SecureDML:An Intelligent Framework for Preventing Poisoning Attacks in Distributed Machine Learning Systems

Archa A. T, Kartheeban K
Computer Science and Engineering, Kalasalingam Academy of Research and Education, Krishnankovil, TamilNadu, India

***Abstract*—The security and protection of models in distributed machine learning (ML) systems require high emphasis on adversarial threats, including poisoning attacks. This study contains a complete framework that integrates different advanced techniques to monitor poison attacks and prevent such attacks for the effective functioning of machine learning systems. The proposed system integrates hybrid encryption for security, and a subsequent anomaly detection method using autoencoders. SHapley Additive exPlanations-based interpretability method is used to enhance model transparency. Hybrid encryption combines the RSA and AES methods to keep data and model parameters secret, and autoencoders provide effective identification of poisoning attack patterns through abnormal data observations. This method is implemented using multimodal datasets such as CIFAR 100 and AG News datasets. Finally,the effectiveness of this method can be evaluated using confusion matrix, comparison graphs. It works as a comprehensive solution that benefits various ML applications, such as healthcare, autonomous vehicles, Large Language Models, etc., for enhancing security along with integrity protection.**

***Keywords*—Poisoning attacks; SHapley Additive exPlanations (SHAP); anomaly detection; federated learning; autoencoders**

I. Introduction

Nowadays, machine learning (ML) offers computational intelligence for a variety of applications. It has many applications in the area of image processing, natural language processing, cybersecurity, robotics, autonomous vehicles, recommendation systems, LLM's etc. Training data has an impact on the accurate prediction of output in such systems. Sometimes, adversaries are coming up with new ways to train ML models, which causes more security risks to clients' data.

Machine learning model security [1] is currently under risk from data poisoning and evasion threats. By altering a model's data, methods, or hyperparameters during the model training phase, poisoning attacks give attackers the ability to manipulate the model's behavior. However, by altering the test sample, an evasion attack is conducted. Adversaries are able to create valid inputs that are invisible to people, yet cause models to forecast incorrectly. There have been multiple significant and effective adversarial attacks [2] against machine learning systems. Highly trained attackers are driven to alter the outcomes of machine learning models by extremely powerful motivations.

Gu, Tianyu,et al. [3] investigated security threats that emerge when training programs are outsourced because an attacker can implement hidden access points in neural networks called BadNets that function normally, but compromise the chosen input data from attackers. The security threat can be illustrated through a toy model that modifies the handwritten digit recognition system. The targeted misuse of a U.S. street sign recognition system directs stop signs to be read as speed limits whenever specific stickers appear on the sign. The integrity of the system remains compromised after network retraining as a backdoor consistently functions regardless of the trained task. Adding training data that either compromises the model's functionality or have an adverse impact on its performance is known as "data poisoning". The architecture of a poisoning attack is shown in Fig. 1 [2]. The model is forced to learn and make predictions for attackers to corrupt information in the training sample. This type of poisoning has spread to all ML applications and is not restricted to any specific domain. The purpose of the label flipping attack is to reverse machine learning detector predictions. Label poisoning is used to switch the labels of the current training data to test the resilience of the ML models.

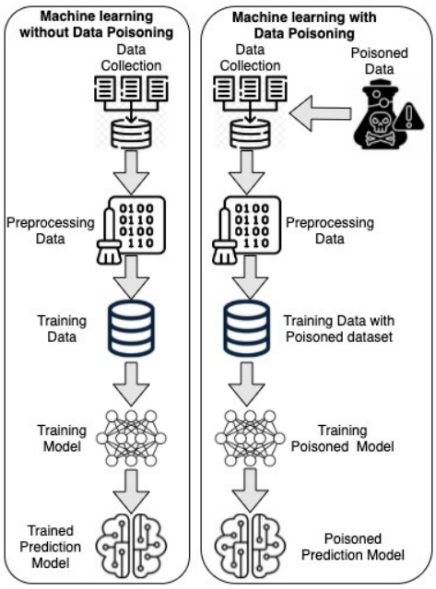

Fig. 1. ML architecture with and without poisoning [2].

Adversaries use training data manipulation to perform data poisoning attacks [4] against ChatGPT and similar LLMs [26] which leads to the inclusion of biases, vulnerabilities, or backdoors in the system. The attacks affect various points in the model training process and deployment stages which resulting in system outputs and behaviors becoming compromised. In

[7], the authors discussed the relevance of current and next-generation federated, secure, and privacy-preserving artificial intelligence methods in medical imaging. It also explores potential attack vectors and future prospects, emphasizing the need for advanced security measures in medical data processing. Transferable cleanlabel poisoning attacks were used in [8] to detect malicious training data without changing labels.

A. Types of Data Poisoning Attacks

1) Training data poisoning: During training, adversaries introduce modified information to manipulate or degrade the model capacity. Attackers embed toxic and false information or biased statements within publicly accessible datasets meant for fine-tuning.

2) Backdoor attacks [9]: The attack triggers a specific output whenever it is active through a hidden command placed in the training data. The model will produce harmful serving responses after receiving specific trigger phrases that appear during the training process on the poisoned data.

3) Exposure attacks: Attackers submit valuable information into the training sample data, which results in inaccurate predictions. The model unintentionally remembers numerous API keys and passwords together with private user communications.

4) Fine-tuning poisoning: During the adjustment of the external dataset, an opponent can develop deceptive data to modify the operational parameters of the model. The utilization of poisoned sentiment data in fine-tuning a chatbot may result in an unfair preference for particular subjects.

5) Prompt injection attacks (inference-time manipulation): The system receives adversarial prompts from the users, which alter the generated responses. The importance of the proposed model is to find direct applications within secure machine learning frameworks used to protect distributed and federated learning setups against data poisoning attacks. Integrity protection of training data is vital for critical sectors deploying AI models such as healthcare, finance, and cybersecurity.

While distributed ML security encompasses multiple dimensions, including encryption, privacy preservation, explainability, and multimodal robustness, this study focuses specifically on poisoning attack detection and mitigation. There are also practical deployability that is illustrated by this work, but are not stand-alone theoretical works.

The main contributions of this proposed work are as follows:

- Abnormal updates of a malicious client: To detect adversarial client updates in distributed learning, this study suggested representation-level anomaly scoring scheme, which is an autoencoder reconstruction. Contrary to distance-based defenses, including KRUM [36] which is based on geometric proximity, SecureDML learns the latent update behavior, making it possible to detect coordinated and low-magnitude stealth attacks.
- Anomaly-sensitive robust aggregation to mitigate: SecureDML uses adaptive anomaly scores in a weighted aggregation scheme, which eliminates the effect of adversarials, but averages update differently. Unlike FedAvg [37] variants, which do not involve adversarial discrimination, the presented approach has a high level of non-IID data distribution resistance and constrained Byzantine adaptation.
- Deterministic compatibility of decoupled robustness: This framework separates the mechanisms of poisoning resilience and privacy, and has no anomaly-masking effect as with DPFL-based [15] defenses, and is also compatible with secure communication and distributed deployment constraints.

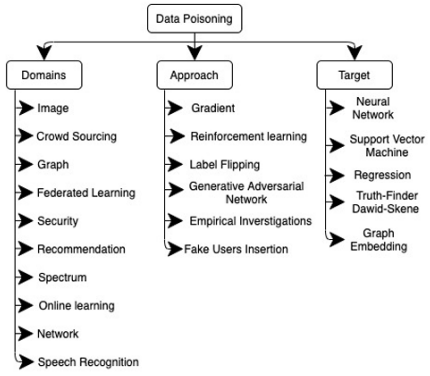

Fig. 2. Attack taxonomy [2].

II. Literature Review

There are different data poisoning attacks emerged in various attack domains. Existing literature on data poisoning attacks can be taxonomized in terms of attack domains, approach, and target (victim), as illustrated in Fig. 2. Recently, trending technologies, such as crowdsourcing and federated learning are always vulnerable, as the authenticity of individual data can never be verified. The most recent victims of poisoning attacks have spread in the security, network, speech recognition, and healthcare. The existing data poisoning approaches have targeted almost all the machine learning algorithms, ranging from traditional algorithms like regression to modern deep neural network architectures.

Researchers and adversaries use a variety of tactics to change data in data poisoning, including the two major data poisoning attacks such as Label Flipping and Feature Poisoning(FP). Lin et al. [10] assessed the effectiveness of LF attacks and suggested a defense mechanism using two genuine datasets from the UCI repository such as MNIST [11], and Spambase [12], which are frequently used benchmarks for classification tasks.Munoz et al. [13]trained a CNN on a labelled dataset that was harmless and manipulated using an LF attack on the MNIST dataset. The researchers determined that adding poisoned training points increased the number of classification mistakes.

A Multiclass Logistic Regression classifier increased the classification error during random LF attacks while replacing the CNN. Fisher et al. [14] proposed approaches to support labeling constraints whereby hackers can apply predefined prediction transformations versus outputting incorrect

results altogether. The repetition of experiments on different datasets, including CIFAR-10, along with a reduced version of ImageNet, demonstrated the effectiveness of the proposed technique. Gradient ascent strategy is used in [1], in which the gradient is computed based on the properties of the optimal solution of SVMs. This method can be kernelized and enables the attack to be constructed in the input space even for nonlinear kernels.

Homomorphic-based transfer learning [15] is used to protect the client's privacy in training tasks by encrypting client data. A detection mechanism is carried out to locate sizable attack groups and their tactics, and to infer their capabilities. The ability of Robust-DPFL to concurrently guarantee task correctness, model robustness, and privacy protection is confirmed by extensive testing on three benchmark datasets: MNIST, FEMNIST, and CIFAR 10. Robust-DPFL updates the global model on detected clean gradients after first differentiating poisoned gradients from clean gradients using their unfair patterns in the components.

Paudice et al. [16] investigated how attackers manipulate training labels through flipping attacks because they weaken the model's performance. Label sanitization, detection and correction methods have been proposed by the authors to minimize the damage from attacks where attackers modify training labels intentionally. This study details the significant accuracy reduction triggered by label flipping attacks, which primarily affects binary classification tasks. To fight such scenarios, the authors introduced a labeling sanitization system.

The k-nearest neighbors (k-NN) method acts as the foundation for this technique, which evaluates suspicious data points through comparison with their nearest neighbors. The algorithm detects infected samples by performing a check that results in either elimination or correction of those samples. The defense mechanism specifically targets label flipping attacks but cannot prevent other poisoning attacks, such as backdoor attacks [17] or gradient manipulation in the same way. The k-NN sanitization method depends on the data set characteristics. In data sets with numerous dimensions, the k-NN algorithm shows reduced effectiveness, whereas the sanitization process leads to possible clean data removal which minimally affects the model prediction accuracy.

Luis Mu˜noz-Gonz´alez et al. [13] evaluated the vulnerability of deep learning model to data poisoning attacks through their proposed novel execution method. The authors extended data poisoning attacks from binary classifiers to multiclass classifiers while demonstrating wider threat applicability. The authors created a back-gradient optimization method that uses automatic differentiation to compute efficient gradients, allowing adversaries to create deceptive training samples that affect multiple models within neural networks and deep learning systems. A new attack method has been successfully applied across different applications, including both spam filtering systems and malware detection systems, and handwritten digit recognition systems. This research demonstrates that adversaries can create transferable attacks through adversarial training, which enhances the destructive capacity of such attacks and defense strategies are not addressed in this study.

Different types of poisoning attacks and their prevention strategies [18] are discussed to improve the efficiency of ML models.Some real time poisoning attack scenarios and detection methods [19] to determine the relevance of using secure ML systems. Jodayree,et al. [5] proposed a robust prevention scheme to enhance the security of federated learning. Three algorithms are proposed here to enable the creation of encrypted and decrypted verification keys for federated learning data files on a training node. Malicious participants are eliminated by the federated learning server and prevent backdoor attacks with the help of an encrypted verification scheme.It can be useful only on low-processing devices. Differentially Private Federated Learning is discussed to perturb the communicated data via independent random noise to pose challenges to user privacy identification.

Ali,et al. [20] proposed two-step defense methods such as global filtering, selffiltering, and adaptive threshold strategy.To address more complex attack patterns, there is a need for various other anomaly detection methods. The need for security of FL systems in IOHT applications [21]. Jia,et al. [22] proposed a generic poison detection approach named CodeDetector to automatically detect poison samples in a suspicious dataset. This study discussed security issues and the need for a more defensive layer.

A. Explainable AI (XAI) and SHAP

A model's predictions become transparent through explainable AI methods (XAI) which ensures that the predictions can be trusted and interpreted. SHAP [23]is a commonly used XAI method that assigns importance values to input variables according to their influence on the model outputs. Anomaly detection benefits from SHAP values because they reveal important influencing features when flagging irregular feature contributions during decision-making processes. The theoretical basis of SHAP relies on Shapley values from cooperative game theory, which provides equitable feature contribution assessments. Lundberg,et al. [24]created an approach that computes SHAP values effectively for complex ML models. Learning professionals used SHAP to demonstrate its success in interpreting tree-based models [24] together with deep learning architectures and alternative machine learning techniques.

SHAP was employed to identify poisoning attacks in adversarial machine learning through the identification of inconsistent feature attributions. Molnar (2022) explained that the interpretability technique SHAP functions to enhance security because it reveals manipulated features present in datasets. SHAP enables the strong protection of ML models against adversarial threats because it delivers clear explanations of individual predictions. Table I shows some recent methods used to detect and prevent poisoning attacks.

III. Problem Statement

Data poisoning corrupts training data, which degrades model functionality and establishes unauthorized access points. These attacks create major threats to datasets containing both images and texts because they specifically target distributed systems. This leads to undesirable outcomes that affect the performance of the distributed ML systems. To handle such systems efficiently, a better data poisoning attack detection and prevention method is required.

TABLE I. SOME RECENT METHODS USED TO DETECT AND PREVENT POISONING ATTACKS AND THEIR CHALLENGES.

Year	Methods Used	Challenges
2025	Global model aggregation algorithm designed to mitigate poisoning attacks [39]	Need more defensive method to address jamming and poisoning attacks in wireless FL networks.
2025	Two-step defense methods,such as global filtering, self-filtering, and adaptive threshold [20]	In order to address more complex attack patterns, there is a need for various other anomaly detection methods to be used.
2024	A generic poison detection approach named CodeDetector to automatically detect poison samples in a suspicious dataset [22]	Security issues, there is a need for a defense layer as extended work.
2023	An encrypted verification scheme in the federated learning model [5], preventing poisoning attacks without requiring specific attack detection programming	Useful only for only low processing devices. Difficult to handle complex data.

Assume that the clean training dataset be:

$$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n}$$

Calculate the model parameters through empirical risk minimization:

$$\theta^* = \arg\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(f_\theta(x_i), y_i)$$

A. Formulation of Poisoning Attack

m malicious samples are injected by adversary:

$\mathcal{D}_p = \{(x_j^p, y_j^p)\}_{j=1}^{m}$.

Then, the poisoned dataset becomes:

$$\mathcal{P}' = \mathcal{P} \cup \mathcal{P}_d$$

The attacked model parameters are calculated as:

$$\theta_p^* = \arg\min_{\theta} \frac{1}{n+m} \sum_{(x,y)\in\mathcal{D}'} \mathcal{L}(f_\theta(x), y)$$

B. Security Goal

The machine learning system is robust, if:

$$\|\theta_p^* - \theta^*\| \leq \epsilon$$

where, the poisoning ratio is:

$$\alpha = \frac{m}{m+n}$$

IV. ADVERSARY ASSUMPTIONS AND THREAT MODEL

A. Distributed Learning Environment

Consider a distributed learning environment that consists of a central server S, which coordinates different participating clients. Each client contains a private local dataset. A local optimization occurs on its own dataset, and each client sends the updates on the model to the server which summarizes them to create the following global model.

B. Adversary Assumption

Consider a limited Byzantine adversary that may fail at most fraction. Bad users can organize and interfere with their local information or sent updates. The adversary acts in either a white-box environment (information about model architecture, global parameters and aggregation rule) or a black-box environment (only has access to compromised local information); unless or otherwise, the more powerful white-box assumption is used.

The attack budget is limited by local data poisoning rate and an update magnitude to have realistic adversarial capability. We measure availability attacks, targeted attacks and backdoor attacks.

The opponent is not allowed to attack the central server, corrupt the mechanism of aggregation, or corrupt validation or test data or to surpass the specified fraction of attack clients. The outlining framework identifies abnormal updates of clients, and suppresses their effects by using powerful aggregation, which ensures that convergence is stable with the influence of attacks being limited under these adversarial conditions.

C. System Model

Here, the model is a federated learning system where all participants are synchronized and where the weighted aggregation of models is done by a central server at every round. Clients calculate local updates on personal data and send this to the server where it is aggregated. The proposed method is applied in the server side which identifies and blocks suspicious updates prior to parameter fusion.

V. PROPOSED METHODS

This method includes different sequential steps for operation. The data ingestion phase enables various clients to send their processed data to the system. Transfer learning-based systems have been used to detect GAN poisoning attacks. Multimodal data, such as image,text data enters the processing system through ViT and BERT [26] before being efficiently processed. Encryption is performed by AES [6,25,27] using RSA Signing method, which provides data security during communication and storage. The distributed learning method allows clients to perform local training and receive securely encrypted information through security protocols. The system detects anomalies through autoencoder. Feature changes can be identified by the SHAP Explainability system when the system explains the prediction models.

The system combines different advanced methods to build a distributed machine learning (ML) [29] model training framework that delivers strong performance and security features

while adapting to changing conditions.This system comprises four main processes, including data ingestion, hybrid RSA and AES encryption-decryption, autoencoder anomaly detection, and SHAP-based interpretability functions.

The operation of proposed system begins with a data ingestion stage. The initial step combines the task of collecting raw data from several sources with the preprocessing operations. Multiple data sources are feed into the system to produce standardized, clean data that are ready for machine learning operations. The system features multiple data type compatibility along with various data source compatibility to accommodate the needs of different distributed environments.

For efficient operation of the ML algorithm, data ingestion methods need to maintain both quality and data consistency because these criteria establish a foundation for the model.The proposed system shields data through a dual encryption solution that joins the Rivest-Shamir-Adleman RSA [30] keys with AES[24] (Advanced Encryption Standard). This approach leverages the strengths of both encryption methods and utilizes RSA encryption for safe encryption key transfers between its components. Data encryption under RSA requires two cryptographic keys that function in public and private form for security. Encryption takes place using the public key but the private key is responsible for the decryption operations. The system maintains secure information transfer of sensitive content such as model parameters and distributed node data through its configuration that defends against unauthorized access.

Large amounts of data undergo secure encryption and decryption using AES (Symmetric Encryption). The encryption process using AES occurs with shared secret keys, which leads to quicker and more efficient encryption of large datasets compared to RSA encryption. During the training and inferential processes, AES operates on bulk data to secure data during transmission and storage.

The anomaly detection capability plays a vital role in such systems because distributed ML environments handle inputs from multiple diverse sources with different quality levels. The system utilizes autoencoders which are unsupervised neural networks, to identify anomalies within the input data. Training an autoencoder enables the network to master a reduced format of data while simultaneously building its reconstruction. Anomalies are detected by measuring the reconstruction error, which establishes the difference between the initial input data and the recreated version. Any data point exceeding the specified thresholds in its reconstruction error was identified as an anomaly. Distributed ML systems identify anomalies as problems, including data attacks and sensor faults that potentially harm model functionality or integrity. The system tracks anomalies in real-time to modify its operation based on changing data patterns while spotting problems that could disrupt the proper model training.

Machine learning model interpretation and trustworthiness increase through the utilization of SHAP (SHapley Additive exPlanations) within the system. The explanation system SHAP performs an unbiased analysis of separate feature impacts on the predictive outcomes for any model type. SHAP values are used to understand model predictions because this interpretability method reveals specific features that influenced

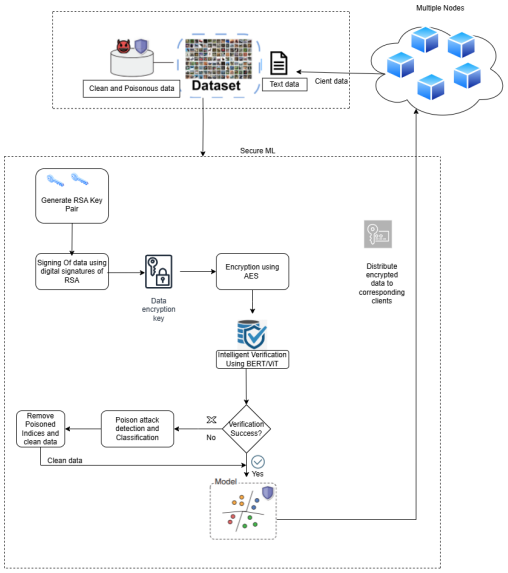

Fig. 3. Secure DML system.

outcomes the most. The interpretability of distributed ML systems becomes critical throughout multiple node training because it enables better debugging, performance improvements, and compliant privacy regulation implementations.

The incorporation of SHAP into the proposed system provides stakeholders with transparent decision-making capabilities to assess the prediction reliability and fairness of the model. The capacity to provide explanations makes this approach essential for industries that must follow explainability regulations in the field of healthcare and finance. The proposed system (Secure DML system) in Fig. 3 consists of data ingestion, encryption, and decryption using the hybrid approach of the RSA and AES methods, anomaly detection using autoencoders and integrity verification using BERT,ViT and SHAP-based interpretability to ensure robust and adaptive model training in distributed ML systems. The detailed steps are shown in Fig. 4.

VI. Experimental Evaluation

The experiments were performed in a distributed machine learning environment using Google Colab while utilizing GPUs to enhance the computational speed. The evaluation was conducted on two multimodal datasets: CIFAR-100 for images and AG News for texts. The development was performed through Python using TensorFlow along with PyTorch to accomplish AES-RSA encryption and BERT [31,32]verification through cryptographic libraries.

A. Datasets

The CIFAR-100 [33] benchmark includes 60,000 images divided into 100 distinct categories. The news articles in AG News [28] consist of 120,000 items divided across four classes. Here,10,000 samples were selected from which 3,000 consisted of clean data points and 7,000 contained poisoned samples with diverse attack methods such as misclassification, label flipping, backdoor, and perturbations. The level of poisoning

Algorithm 1 Secure Distributed ML System

Input: Clients $\{C_1, C_2, ..., C_N\}$, Global Model θ, Streaming Data $\mathcal{D}_t^{stream}$
Output: Securely Trained Global Model θ^* with Updates
Step 1: System Initialization
Generate RSA Key Pair (PK, SK)
Generate AES Key K_{AES}
for each client C_i **do**
  Encrypt K_{AES} with RSA: $K_{AES}^{enc} = Encrypt(PK, K_{AES})$
  Securely store K_{AES}^{enc}
  Initialize local model $\theta_i \leftarrow \theta$
end for
Step 2: Data Ingestion and Feature Extraction
for each client C_i receiving new data stream $\mathcal{D}_t^{stream}$ **do**
  Extract features from images using ViT and text using BERT
  Normalize and encrypt features using AES: $E_i = Encrypt(K_{AES}, X_i)$
end for
Step 3: Federated Learning with Secure Updates
for each global round $t = 1, 2, ..., T$ **do**
  for each client C_i processing real-time data **do**
    Train local model $\theta_{i,t}$ on encrypted data E_i
    Compute anomaly score $A_i = ||X_i - \hat{X}_i||^2$ using Autoencoder
    if A_i exceeds threshold **then**
      Mark data as suspicious, reject update
    else
      Send $\theta_{i,t}$ update to server
    end if
  end for
  Server aggregates valid updates: $\theta_{t+1} = \frac{1}{N}\sum_{i=1}^{N}\theta_{i,t}$
end for
Step 4: Model Adaptation
while system is running **do**
  Receive new client data
  Compute SHAP values for interpretability
  if SHAP indicates feature drift or anomaly **then**
    Adjust model training with adaptive learning rate
  end if
  Update global model θ^* dynamically
  Deploy updated model for inference
end while
End of Algorithm

Fig. 4. Steps for secure DML system.

is on the client level. A fraction of clients are assumed to be compromised, and inject malicious updates every time they undergo local training. The assumptions of Byzantine federated learning assume 30% of the clients are malicious. A variety of poisoning attack strategies have been employed to evaluate this approach. An attack strategy based on GAN provides data poisoning through adversarial samples that deceive classification models. The implementation of specific triggers through Backdoor Attacks allows predictions to be manipulated in their target directions. The process of label flipping was used to create inconsistencies within the dataset. Slight modifications of the data cause perturbation-based attacks. AES-256 [34] operated as the encryption method for data protection, whereas key exchange and digital signature operations relied on RSA-4096.

B. Results

The effectiveness of this cryptographic method can be implemented by comparing the method with benchmark methods. Fig. 5 shows the classification report and confusion matrix on multiple clients without a defense mechanism. As the number of clients increases,the attack detection rate decreases, as shown in Fig. 6. This demonstrates the need for a defense layer in ML systems. Adding a hybrid defense mechanism helps to secure sensitive data. The classification methods fail to provide interpretable explanations thus making it complex to understand decision-making reasons. This system integrates SHapley Additive exPlanations (SHAP) to generate detailed explanations for detected poisonous data samples. Through SHAP analysis, consistent features are discovered which led to the detection of anomalies among different clients.

The scatterplot showing anomalies are shown in Fig. 7 and the anomaly detection on multiple clients is shown in Fig. 8.

The BERT-ViT hybrid model functions as an integrity

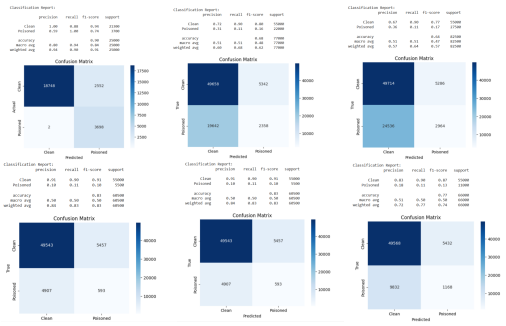

Fig. 5. Confusion matrix without having defense mechanism.

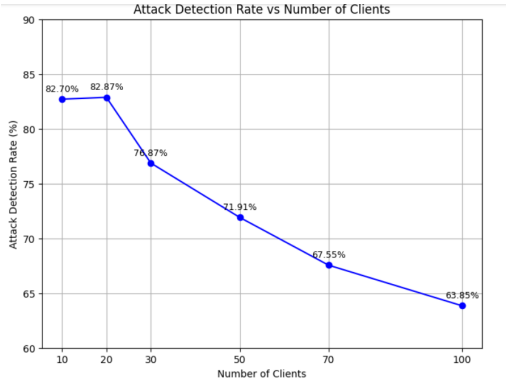

Fig. 6. Attack Detection rate without having defense mechanism.

verification system that verifies the authenticity of the data while detecting poisonous samples. The Attack Success Rates based on an increasing number of poisonous points are shown in Fig. 9. ASR can be calculated as:

$$ASR = \frac{Npsuccess}{Np} * 100 \quad (1)$$

where, $Npsuccess$ denotes the number of data samples and

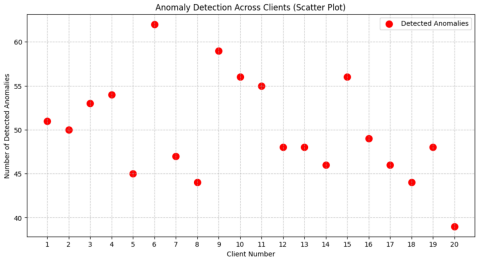

Fig. 7. Scatterplot showing anomalies.

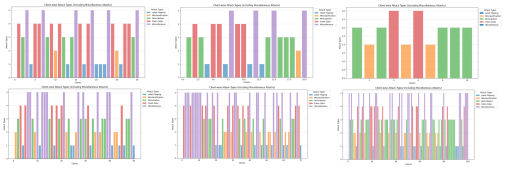

Fig. 8. Different poisoning attacks on muiltiple clients.

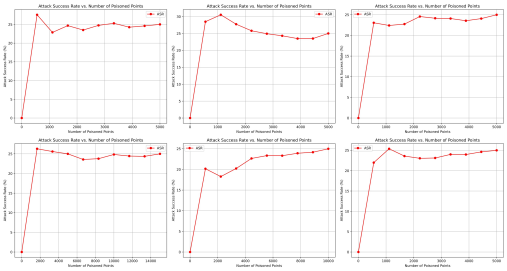

Fig. 9. Attack success rates.

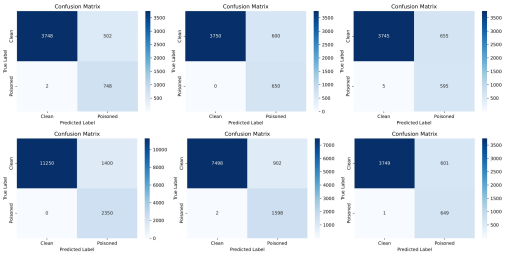

Fig. 10. Confusion matrix.

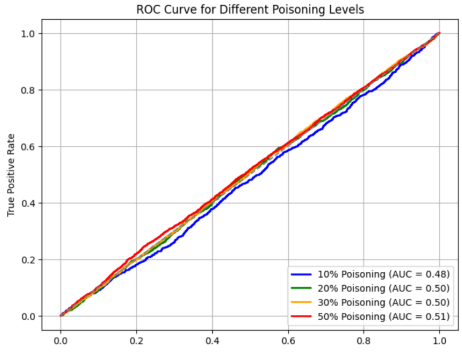

Fig. 11. ROC curve.

Fig. 12. Attack detection rate.

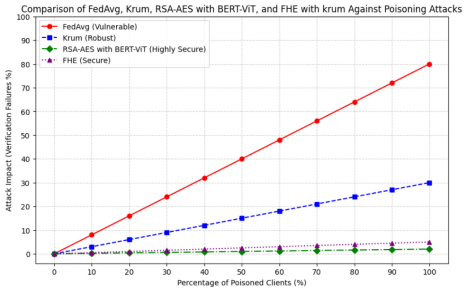

Fig. 13. Comparison with other methods.

Np denotes total no. of poisoned points.

The SHAP-based classification demonstrated better performance based on its high precision and recall values, as confirmed by the ROC curve analysis. The SHAP analysis delivered saturated results through its confusion matrix where poisonous and clean data categories showed distinct separation.

The effectiveness of SHAP based methods can be achieved using a confusion matrix which is shown in Fig. 10.

The ROC curve in Fig. 11 shows the variation of AUC values across different levels of poisoning.

The poisoning attack detection rate across an increased number of clients (number of clients 10,20,30,..100) are shown in Fig. 12. As the number of client increases, the rate of attack detection becomes relatively stable, indicating that the system is highly robust. The detection performance of the system was high, which demonstrates the scalability of the system.

VII. Discussion

This proposed defense method is compared with other benchmark methods such as KRUM [36],FedAvg [38], and FHE [37] methods.The verification failures are less in the proposed method as compared with other methods.This indicating that the hybrid defense method is highly secure.The analysis graph is shown in Fig. 13.

VIII. Applications and Future Enhancements

The proposed method is used to maintain secure federated learning models [35] by protecting them from distributed machine learning attacks. Hybrid RSA-AES encryption allows the approach to provide protected model updates thereby proving ideal for sensitive applications, including healthcare and finance. The detection capabilities of the system are improved through nDPI-based poisoned packet detection in network environments. The system permits the immediate disclosure of adversarial network traffic while Streamlit operates an intrusion detection dashboard that delivers an easy-touse interface to security analysts to track potential threats in real time and network intrusion detection systems are enhanced through the analysis of poisoned packets by combining nDPI with deep learning classifiers.

Data manipulation must not impact AI systems operating in finance, autonomous systems, or healthcare applications. This system can be applied to the management of IoT system security and edge environments. The system can be expanded to perform better multimodal verification when applied to security applications such as Deepfake detection in video environments. The system's resistance to adversarial modifications in voice-based AI systems increases when the integration of Whisper or Wav2Vec models is included. This system is highly secure, achieves privacy and protection, and is scalable for

moderate users. Its explainability feature helps to detect and interpret anomalous features. Although the system achieves better performance, the computation overhead is still an issue, which is further enhanced in the future.

IX. Conclusion

Security threats from poisoning attacks create major problems in the performance and integrity of distributed machine learning models. The attackers inject deceptive or harmful information into model training, which damages performance and quality, and generates incorrect predictions. This system offers a unified solution that uses various cutting-edge methods to detect and protect against poisoning. The proposed system offers security through robust distributed ML models, enhancement, and protective measures to combat poisoning attacks. The system utilizes anomaly detection together with secure encryption and model interpretability to discover suspicious data while stopping malicious inputs, thereby maintaining the uninterrupted integrity of distributed models. Thus, distributed ML systems are secure against adversarial attacks and maintain good performance.

X. Statements and Declaration

There is no potential conflict of interest reported by the author(s).

References

[1] Biggio, Battista, Blaine Nelson, and Pavel Laskov. "Poisoning attacks against support vector machines." arXiv preprint arXiv:1206.6389 (2012).

[2] Aryal, Kshitiz, Maanak Gupta, and Mahmoud Abdelsalam. "Analysis of label-flip poisoning attack on machine learning based malware detector." In 2022 IEEE International Conference on Big Data (Big Data), pp. 4236-4245. IEEE, 2022.

[3] Gu, Tianyu, Brendan Dolan-Gavitt, and Siddharth Garg. "Badnets: Identifying vulnerabilities in the machine learning model supply chain." arXiv preprint arXiv:1708.06733 (2017).

[4] Archa, A. T., and K. Kartheeban. "A Review on Privacy Enhanced Distributed ML Against Poisoning Attacks." In International Conference on Cyber Security, Privacy in Communication Networks, pp. 173-186. Singapore: Springer Nature Singapore, 2023.

[5] Jodayree, Mahdee, Wenbo He, and Ryszard Janicki. "Preventing image data poisoning attacks in federated machine learning by an encrypted verification key." Procedia Computer Science 225 (2023): 2723-2732.

[6] Zodpe, Harshali, and Ashok Sapkal. "An efficient AES implementation using FPGA with enhanced security features." Journal of King Saud University-Engineering Sciences 32, no. 2 (2020): 115-122.

[7] Kaissis, Georgios A., Marcus R. Makowski, Daniel Rückert, and Rickmer F. Braren. "Secure, privacy-preserving and federated machine learning in medical imaging." Nature Machine Intelligence 2, no. 6 (2020): 305-311.

[8] Zhu, Chen, W. Ronny Huang, Hengduo Li, Gavin Taylor, Christoph Studer, and Tom Goldstein. "Transferable clean-label poisoning attacks on deep neural nets." In International conference on machine learning, pp. 7614-7623. PMLR, 2019.

[9] Zhang, Jie, Chen Dongdong, Qidong Huang, Jing Liao, Weiming Zhang, Huamin Feng, Gang Hua, and Nenghai Yu. "Poison ink: Robust and invisible backdoor attack." IEEE Transactions on Image Processing 31 (2022): 5691-5705.

[10] Lin, Jing, Long Dang, Mohamed Rahouti, and Kaiqi Xiong. "ML attack models: Adversarial attacks and data poisoning attacks." arXiv preprint arXiv:2112.02797 (2021).

[11] Deng, Li. "The mnist database of handwritten digit images for machine learning research [best of the web]." IEEE signal processing magazine 29, no. 6 (2012): 141-142.

[12] Hopkins, Mark, Erik Reeber, George Forman, and Jaap Suermondt. "Spambase, UCI Machine Learning Repository (1999)." DOI: https://doi.org/10.24432/C53G6X (2024).

[13] Muñoz-González, Luis, et al. "Towards poisoning of deep learning algorithms with back-gradient optimization." Proceedings of the 10th ACM workshop on artificial intelligence and security. 2017.

[14] Fisher, Ronald Aylmer. "Iris. UCI machine learning repository." DOI: https://doi. org/10.24432/C56C76 (1988).

[15] Lee, Seewoo, Garam Lee, Jung Woo Kim, Junbum Shin, and Mun-Kyu Lee. "HETAL: Efficient privacy-preserving transfer learning with homomorphic encryption." In International conference on machine learning, pp. 19010-19035. PMLR, 2023.

[16] Paudice, Andrea, Luis Muñoz-González, and Emil C. Lupu. "Label sanitization against label flipping poisoning attacks." In Joint European conference on machine learning and knowledge discovery in databases, pp. 5-15. Cham: Springer International Publishing, 2018.

[17] Chen, Xinyun, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. "Targeted backdoor attacks on deep learning systems using data poisoning." arXiv preprint arXiv:1712.05526 (2017).

[18] Archa, A. T., and K. Kartheeban. "Real time poisoning attacks and privacy strategies on machine learning systems." In 2024 3rd International Conference on Sentiment Analysis and Deep Learning (ICSADL), pp. 183-189. IEEE, 2024.

[19] AT, Archa, and K. Kartheeban. "SecureTransfer: A Transfer Learning Based Poison Attack Detection in ML Systems." International Journal of Advanced Computer Science & Applications 15, no. 7 (2024).

[20] Ali, Yasir, Kyung Hyun Han, Abdul Majeed, Joon S. Lim, and Seong Oun Hwang. "An Optimal Two-Step Approach for Defense Against Poisoning Attacks in Federated Learning." IEEE Access (2025).

[21] Coelho, Kristtopher K., Michele Nogueira, Alex B. Vieira, Edelberto F. Silva, and Jose Augusto M. Nacif. "A survey on federated learning for security and privacy in healthcare applications." Computer Communications 207 (2023): 113-127.

[22] Li, Jia, Zhuo Li, HuangZhao Zhang, Ge Li, Zhi Jin, Xing Hu, and Xin Xia. "Poison attack and poison detection on deep source code processing models." ACM Transactions on Software Engineering and Methodology 33, no. 3 (2024): 1-31.

[23] Molnar, C. "Interpretable machine learning, 2nd edn. christophm. github. io/interpretable-ml-book." (2022).

[24] Lundberg, Scott M., and Su-In Lee. "A unified approach to interpreting model predictions." Advances in neural information processing systems 30 (2017).

[25] Lalem, Farid, Abdelkader Laouid, Mostefa Kara, Mohammed Al-Khalidi, and Amna Eleyan. "A novel digital signature scheme for advanced asymmetric encryption techniques." Applied Sciences 13, no. 8 (2023): 5172.

[26] Ferrag, Mohamed Amine, Mthandazo Ndhlovu, Norbert Tihanyi, Lucas C. Cordeiro, Merouane Debbah, Thierry Lestable, and Narinderjit Singh Thandi. "Revolutionizing cyber threat detection with large language models: A privacy-preserving bert-based lightweight model for iot/iiot devices." IEEe Access 12 (2024): 23733-23750.

[27] Guitouni, Zied, Mohammed Ali Ghaieb, and Mohsen Machhout. "Security analysis of medical image encryption using AES modes for IoMT systems." IJCA Int J Comput Appl 14, no. 2 (2023): 8887.

[28] Rothman, Denis. Transformers for Natural Language Processing: Build, train, and fine-tune deep neural network architectures for NLP with Python, Hugging Face, and OpenAI's GPT-3, ChatGPT, and GPT-4. Packt Publishing Ltd, 2022.

[29] Jiang, Zoe L., Jiajing Gu, Hongxiao Wang, Yulin Wu, Junbin Fang, Siu-Ming Yiu, Wenjian Luo, and Xuan Wang. "Privacy-preserving distributed machine learning made faster." In Proceedings of the 2023 secure and trustworthy deep learning systems workshop, pp. 1-14. 2023.

[30] Rivest, Ronald L., Adi Shamir, and Leonard Adleman. "A method for obtaining digital signatures and public-key cryptosystems." Communications of the ACM 21, no. 2 (1978): 120-126.

[31] Lyu, Weimin, Songzhu Zheng, Lu Pang, Haibin Ling, and Chao Chen. "Attention-enhancing backdoor attacks against bert-based models." arXiv preprint arXiv:2310.14480 (2023).

[32] Schaerf, Ludovica, Eric Postma, and Carina Popovici. "Art authentication with vision transformers." Neural Computing and Applications 36, no. 20 (2024): 11849-11858.

[33] Zheng, YuYu, HaoXuan Huang, and JunMing Chen. "Comparative analysis of various models for image classification on Cifar-100 dataset." In Journal of Physics: Conference Series, vol. 2711, no. 1, p. 012015. IOP Publishing, 2024.

[34] Daemen, Joan, and Vincent Rijmen. The design of Rijndael. Vol. 2. New York: Springer-verlag, 2002.

[35] Qi, Tao, Huili Wang, and Yongfeng Huang. "Towards the robustness of differentially private federated learning." In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, no. 18, pp. 19911-19919. 2024.

[36] García-Márquez, M., et al.,"Krum Federated Chain (KFC): Using Blockchain to Defend Against Adversarial Attacks in Federated Learning," *arXiv preprint arXiv:2502.06917*, 2025.

[37] Jiang, S., et al., "Towards compute-efficient Byzantine-robust federated learning with fully homomorphic encryption," *Nature Machine Intelligence*, vol. 7, no. 10, pp. 1657–1668, 2025.

[38] Gonçalves, J. P. S.,"Robustness of AI models in software vulnerability detection," 2025.

[39] Barkatsa, Sofia, et al. *Coordinated Jamming and Poisoning Attack Detection and Mitigation in Wireless Federated Learning Networks.,IEEE Open Journal of the Communications Society (2025)*