

# **A Hybrid Encryption Framework for Scalable and Secure Federated Learning**

**A PROJECT REPORT**

**Mini Project (01CE0609)**

*Submitted by*

**AUMANSH VIJAYENDRA GUPTA**

**92200103209**

**BACHELOR OF TECHNOLOGY**

*in*

**Computer Engineering**



**Faculty of Engineering & Technology**

**Marwadi University, Rajkot**

**April, 2025**



## **Mini Project (01CE0609)**

Department of Computer Engineering

**Faculty of Engineering & Technology**

**Marwadi University**

**A.Y. 2024-25**

## **CERTIFICATE**

This is to certify that the project report submitted along with the project entitled **A Hybrid Encryption Framework for Scalable and Secure Federated Learning** has been carried out by **Aumansh Vijayendra Gupta** (92200103209) under my guidance in partial fulfilment for the degree of Bachelor of Technology in Computer Engineering, 6<sup>th</sup> Semester of Marwadi University, Rajkot during the academic year 2024-25.

Dr. Deepak Kumar Verma

Associate Professor

Department of Computer Engineering

Dr. Krunal Vaghela

Professor & Head

Department of Computer Engineering



## **Mini Project (01CE0609)**

Department of Computer Engineering

**Faculty of Engineering & Technology**

**Marwadi University**

**A.Y. 2024-25**

## **DECLARATION**

We hereby declare that the **Mini Project (01CE0609)** report submitted along with the Project entitled **A Hybrid Encryption Framework for Scalable and Secure Federated Learning** submitted in partial fulfilment for the degree of Bachelor of Technology in Computer Engineering to Marwadi University, Rajkot, is a bonafide record of original project work carried out by me / us at Marwadi University under the supervision of **Dr. Deepak Kumar Verma** and that no part of this report has been directly copied from any students' reports or taken from any other source, without providing due reference.

Name of the Student

Sign of Student

Aumansh Vijayendra Gupta

## Acknowledgement

I would like to express our heartfelt gratitude to everyone who contributed to the successful completion of this project, “*A Hybrid Encryption Framework for Scalable and Secure Federated Learning*”.

First and foremost, I extend my sincere appreciation to our guide Prof. Dr. Deepak Kumar Verma for their invaluable guidance, continuous support, and encouragement throughout the project. Their expert insights and constructive feedback have greatly enhanced our understanding of the subject.

I would also like to thank Marwadi University for providing us with the necessary resources, facilities, and academic support to carry out this research effectively.

A special note of appreciation goes to our peers and fellow researchers for their constructive discussions and encouragement, which helped refine our approach.

Lastly, we extend our gratitude to our families and friends for their unwavering support, patience, and motivation throughout this journey.

This project has been a great learning experience, and we hope that our findings contribute to further advancements in traffic forecasting and intelligent transportation systems.

## Abstract

*In this work, we refer to the novel cryptographic framework that integrates BFV-Brakerski, Fan, and Vercauteren, as well as CKKS Cheon-Kim-Kim-Song homomorphic encryption with adaptive privacy control, as FLIP (Federated Learning Implementation with Privacy). FLIP incorporates a privacy practitioner within the model framework to dynamically adjust the encryption parameters, optimizing security and performance in proportion to system demands. The framework achieves secure model aggregation at a total computation cost that enables scalable FL to be deployed in large-scale installations with minimal cost. Our evaluation on the MNIST and CIFAR-10 datasets under adversarial data poisoning and model inversion attack scenarios shows that FLIP achieves 99.23% accuracy on MNIST and 66.35% on CIFAR-10, with lowered leakage privacy of  $\varepsilon = 6$  and 8, respectively, outperforming the protected FL leak lower bound results. FLIP reduces the computational burden, further enabling unrestricted device scaling while maintaining resource constraints. The proposed strategy addresses the growing demand for privacy in medical, cybersecurity, and banking domains. Integrating blockchain, quantum hardened defenses, and advanced FL security will be explored in future work as a response to adaptive threat landscapes.*

## List of Figures

	PAGE NO.
Fig 1.1 Distributed Hybrid Framework	2
Fig 3.1 Proposed Framework	8
Fig 3.2 RSA ENCRYPTION	9
Fig 3.3 Naïve Bayes, LR and RF	10
Fig 3.4 Using LSTM and CNN Model	11
Fig 3.5 BFV Model Implementation	12
Fig 3.6 CKKS Model Implementation	13
Fig 4.1 Accuracy on MNIST	15
Fig 4.2 Accuracy on CIFAR-10	15
Fig 4.3 Confusion Matrix of MNIST	16
Fig 4.4 Confusion Matrix of CIFAR-10	16
Fig 4.5 Decryption of MNIST	17
Fig 4.6 Decryption of CIFAR-10	17

**List of Tables**

	PAGE NO.
Table 1.1 Research Gap Analysis	3
Table 4.1 Comparison of Attacks	12
Table 4.2 MNIST Dataset Analysis	12
Table 4.3 CIFAR Dataset Analysis	13
Table 4.4 Accuracy on CIFAR-10	14
Table 4.5 Accuracy on MNIST	14
Table 4.6 Performance Metrics	15

## Abbreviations

<b>AUC</b>	<b>Area Under Curve</b>
<b>ROC</b>	<b>Receiving Operating Characteristic</b>
<b>CKKS</b>	<b>Cheon-Kim-Kim-Seong</b>
<b>BKV</b>	<b>Brakerski, Fan, and Vercauteren</b>
<b>CIFAR-10</b>	<b>Canadian Institute for Advanced Research</b>
<b>MNIST</b>	<b>Modified Institute of Standards and Technology</b>
<b>FLIP</b>	<b>Federated Learning Implementation with Privacy</b>
<b>PPFL</b>	<b>Privacy Preserving Federated Learning</b>
<b>ADV</b>	<b>ADVERSARIAL</b>
<b>OC</b>	<b>Open Challenges</b>
<b>DM</b>	<b>Different Models</b>
<b>SS</b>	<b>Selection Strategies</b>
<b>DD</b>	<b>Diverse Datatypes</b>
<b>ROA</b>	<b>Real One Adjacency</b>
<b>RTA</b>	<b>Real Time Application</b>
<b>dec_ckks/DEC<sub>CKKS</sub></b>	<b>Decryption at CKKS</b>
<b>dec_bfv/DEC<sub>BFV</sub></b>	<b>Decryption at BFV</b>
<b>ENC<sub>BFV</sub></b>	<b>Encryption at BFV</b>
<b>ENC<sub>CKKS</sub></b>	<b>Encryption at CKKS</b>
<b>BSFL</b>	<b>Blockchain-enabled Secure FL</b>
<b>S</b>	<b>Dataset</b>
<b><math>p_i</math></b>	<b>Probability of Class <math>i</math></b>
<b>V</b>	<b>Set of all possible values</b>
<b>IG</b>	<b>Information Gain</b>
<b><math>S_v</math></b>	<b>Subset of S where attribute A takes value v</b>
<b><math>f_\theta</math></b>	<b>CNN Encryption Function</b>
<b><math>g_\varphi</math></b>	<b>CNN based Decryption Function</b>
<b>X'</b>	<b>Reconstructed Original Data</b>
<b>C</b>	<b>Filtered CNN layers to generate encrypted representation</b>
<b>M</b>	<b>Malware Dataset</b>
<b><math>R: Z[X]/(X^n + 1)</math></b>	<b>Encryption Scheme for Computations</b>
<b><math>c_0</math></b>	<b>Cipher Text</b>
<b><math>pk_0 \cdot u</math></b>	<b>Public Key Encryption</b>
<b><math>e_1</math></b>	<b>Error</b>
<b><math>m q </math></b>	<b>Scaling or Adjusting the Data</b>
<b>d</b>	<b>Mean difference</b>
<b><math>s_d</math></b>	<b>Standard Deviation</b>
<b>n</b>	<b>Number of Samples</b>



## Table of Contents

	PAGE NO.
<b>ACKNOWLEDGEMENT</b>	<b>I</b>
<b>ABSTRACT</b>	<b>II</b>
<b>LIST OF FIGURES</b>	<b>III</b>
<b>LIST OF TABLES</b>	<b>IV</b>
<b>ABBREVIATIONS</b>	<b>V</b>
<b>1. INTRODUCTION</b>	<b>1</b>
1.1 OVERVIEW	1
1.2 ENCRYPTION SCHEMES	1
1.3 ATTACKS PREVENTION	1
<b>2. LITERATURE REVIEW</b>	<b>3</b>
2.1 ENC BASED PPFL	4
2.2 ADV ATTACKS IN FL	5
2.3 FL WITH BLOCKCHAIN	5
2.4 RESEARCH GAPS AND OC	6
<b>3. RESEARCH METHODOLOGY</b>	<b>7</b>
3.1 ALGORITHM	7
3.2 PROPOSED FRAMEWORK	8
3.3 COMPARISON WITH DM	8
3.4 METHODOLOGY	11
<b>4. RESULTS &amp; DISCUSSIONS</b>	<b>12</b>
<b>5. CONCLUSION &amp; FUTURE WORK</b>	<b>17</b>
5.1 ADVANCED CLIENT SS	19
5.2 ∫ OF BLOCKCHAIN	19
5.3 EVALUATION OF DD	19
5.4 EXPLORATION OF ROA	19
5.5 SCALABILITY & RTA	20
5.6 LIMITATIONS	20
<b>REFERENCES</b>	<b>21</b>

# CHAPTER 1

## INTRODUCTION

### 1.1. Overview

Federated Learning (FL) is a privacy-centric collaborative machine learning technique that permits multiple participants to train models while keeping the data decentralized. [1] This approach greatly helps in the healthcare, finance, and cybersecurity industries, which have data privacy regulations and confidentiality concerns that restrict the sharing of data. [2] Unfortunately, data sharing directly supports federated learning (FL), which is vulnerable to data poisoning, model inversion attacks, and privacy leaks, thereby undermining the confidentiality and integrity of both the data and the model [5].

### 1.2. Encryption Schemes

Several encryption schemes have been proposed to address these vulnerabilities, including HE, secure multi-party computation, and differential privacy. In this regard, homomorphic encryption [5] stands out as it can perform operations on encrypted data, making it a vital structure for preserving privacy. The downside is that traditional HE schemes have excessive computational overhead, making them unsuitable for large-scale FL operations. This problem leads us to the need for optimized security frameworks that are efficient, scalable, and adaptable to operational needs while maintaining performance standards. In this paper, we present FLIP (Federated Learning Implementation with Privacy), a novel privacy-preserving federated learning framework combining BFV and CKKS homomorphic encryption with tailored adaptive privacy control mechanisms [18, 20].

### 1.3 Attacks Prevention

By incorporating a privacy analyst into the FL model architecture, FLIP dynamically adjusts the encryption parameters in real time to enhance the security-computation-accuracy balance. Extensive experimental evaluations on the MNIST and CIFAR-10 databases demonstrate that FLIP reduces privacy leakage ( $\epsilon = 6$  and 8, respectively) while also increasing accuracy to 99.23% and 66.35%, respectively, for MNIST and CIFAR-10, compared to the encryption methods. Furthermore, FLIP enhances computational

efficiency, increasing the scalability of these systems for resource-constrained devices. This paper is organized as follows. In Section 2, we discuss works related to privacy-preserving federated learning (FL) under protected frameworks and examine existing encryption techniques. In Section 3, we explain the FLIP framework, describing its encryption methods and adaptive privacy management features. In Section 4, we describe the experiments conducted, the datasets and evaluation metrics used, FLIP’s performance analysis, and a detailed evaluation of the results showcasing FLIP’s efficiency, effectiveness, and scalability in secure communications. Section 5 concludes the study by presenting the main insights drawn and outlining avenues for further research.

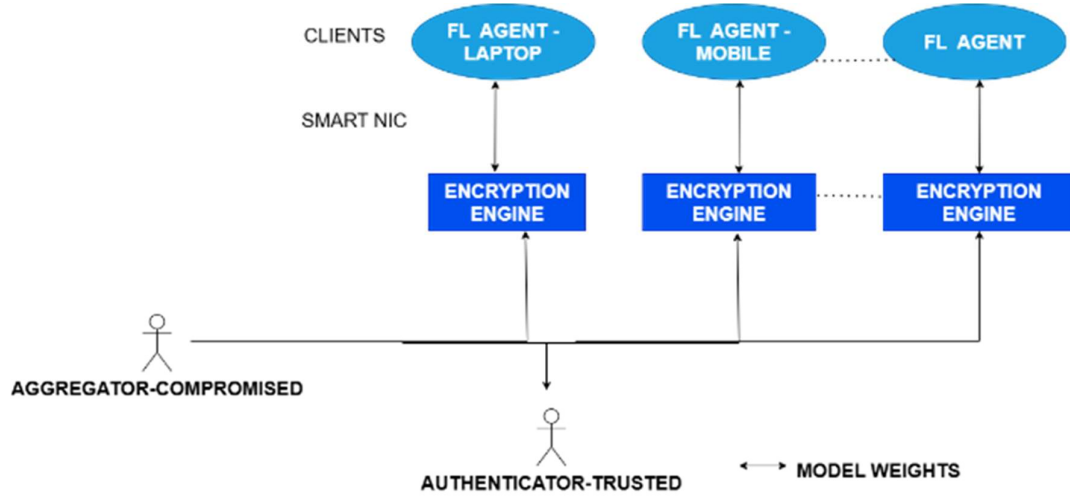


FIGURE 1.1: DISTRIBUTED HYBRID FRAMEWORK

## CHAPTER 2

### LITERATURE REVIEW

The literature on privacy-preserving Federated Learning (FL) applies HE, DP, and SMPC techniques, each with its respective shortcomings. HE enables secure computations but adds enormous computational burdens. DP diminishes privacy, often at the cost of accuracy, and SMPC is poorly scalable. As noted in **Table 1: Research Gap Analysis**, existing solutions suffer from a lack of adaptive privacy control, persistent vulnerability to active adversaries, and inefficiency in scaling. To bridge these gaps, we propose FLIP (Federated Learning Implementation with Privacy), a novel framework for BFV and CKKS encryption that supports dynamic privacy adaptation.

*TABLE 1.1: RESEARCH GAP ANALYSIS*

Ref. No.	YEAR	METHODS	MERITS	DEMERITS	RESEARCH GAPS
[1].	2025	Fully Homomorphic Encryption (FHE)	Strong privacy guarantees for malware feature sharing	High computational cost limits scalability	Efficient FHE schemes are needed for large-scale malware datasets.
[2].	2025	Homomorphic Encryption + SMPC	Balances privacy with utility in collaborative training	Trade-off between accuracy and privacy	The optimal noise levels for malware classification remain unclear.
[3].	2025	Adversarial Training + FHE	Enhanced robustness against adversarial malware samples	Increased training time and complexity	The effectiveness of adversarial training in encrypted settings is understudied.
[4].	2025	Encrypted Gradient Aggregation	Protects intermediate model updates during training	Communication overhead in encrypted aggregation	Efficient protocols for gradient aggregation are lacking.
[5].	2025	Blockchain + Homomorphic Encryption	Ensures immutable audit trails for malware detection models	High energy consumption limits real-time deployment	Integration with FL systems is challenging.
[6].	2025	Lightweight Cryptography	Suitable for resource-constrained IoT devices detecting malware	Limited functionality for complex ML models	Integration with non-linear models is unresolved.
[7].	2025	Post-Quantum Cryptography	Resists quantum attacks on encrypted malware models	Large key sizes increase storage requirements	Compatibility with existing FL frameworks is unclear.
[8].	2025	Hybrid Cryptography (FHE + SMPC)	Combines efficiency and security for malware detection	Complex implementation and coordination	Lack of benchmarks for hybrid approaches in malware detection.
[9].	2025	Threshold Cryptography	Distributes trust among participants in malware detection	Coordination overhead for key management	Key recovery mechanisms are underdeveloped.
[10].	2025	Oblivious Transfer	Ensures privacy during malware feature sharing	Limited scalability for large datasets	Practical implementation in FL is challenging.

[11].	2024	Ring-Learning with Errors (RLWE)	Efficient encryption scheme for network-based malware detection	Noise accumulation affects model stability	The long-term stability of encrypted models is uncertain.
[12].	2024	Attribute-Based Encryption	Fine-grained access control for sensitive malware data	Complex policy management	Policy enforcement in dynamic environments is unclear.
[13].	2024	Proxy Re-Encryption	Enables secure sharing of malware models across domains	Trust assumptions in proxy re-encryption	Security guarantees in adversarial settings are weak.
[14].	2024	Secret Sharing	Distributes trust among participants in collaborative malware detection	High overhead for real-time applications	Not suitable for streaming malware data.
[15].	2024	Identity-Based Encryption	Simplifies key management for malware detection	Limited scalability for large-scale FL	Integration with FL systems is underexplored.
[16].	2024	Homomorphic Signature Schemes	Verifies integrity of encrypted malware detection models	Computational cost limits practicality	Scalability for large-scale systems is questionable.
[17].	2024	Privacy Amplification	Reduces information leakage during malware model training	Requires additional steps for noise management	The impact on model utility is not well understood.
[18].	2024	Blind Computing	Processes malware data without decryption	Limited applicability to diverse malware types	The generalization to different malware families is unclear.
[19].	2024	Decentralized Key Management	Eliminates central points of failure in malware detection	Coordination challenges for key revocation	Key revocation mechanisms are underdeveloped.
[20].	2024	Dynamic Encryption	Adapts encryption strength to malware data sensitivity	Increased complexity in dynamic settings	The stability of encrypted models over time is unknown.
[21].	2023	Federated Transfer Learning + FHE	Enables knowledge transfer across domains for malware detection	High computational cost limits scalability	Scalability for multi-domain systems is unaddressed.
[22].	2023	Quantum-Secure Federated Learning	Future-proof security for malware detection models	Limited adoption of quantum-resistant schemes	Integration with classical FL systems is unclear.
[23].	2023	Adaptive Homomorphic Encryption	Adjusts encryption strength dynamically for malware detection	Resource-intensive for real-time applications	Real-time adaptability is challenging.
[24].	2023	Encrypted Model Pruning	Reduces model size while preserving privacy	Loss of accuracy for complex malware families	The trade-offs between compression and privacy are unclear.
[25].	2023	Cross-Domain Encryption	Facilitates secure collaboration across organizations	Interoperability issues across domains	Standardization across domains is lacking.

## 2.1 Encryption-Based Privacy-Preserving Federated Learning

Privacy preservation in Federated Learning (FL) is a critical concern, particularly when model updates can leak sensitive data. **Homomorphic Encryption (HE)** has emerged as a robust solution, allowing computations on encrypted data without requiring decryption. **BFV and CKKS encryption schemes** are widely used in FL due to their ability to support

secure model aggregation. Studies such as [1] and [2] explore the trade-offs between computational efficiency and security in HE-based FL. Additionally, **Secure Multi-Party Computation (SMPC) and Differential Privacy (DP)** have been integrated with FL to enhance privacy guarantees. However, HE incurs a computational cost, limiting its adoption in real-time applications.

Recent advancements in **lightweight cryptographic techniques** aim to optimize privacy-preserving FL for resource-constrained devices. Research on **hybrid cryptographic approaches**, such as the combination of **Homomorphic Encryption with Secure Aggregation**, demonstrates improved privacy with reduced computational overhead [4]. However, challenges remain in balancing privacy preservation and model utility.

## 2.2 Adversarial Attacks on Federated Learning

Despite its privacy advantages, FL is weak to various adversarial attacks, which can compromise model integrity and protecting information from unauthorized access, disclosure or use, ensuring that sensitive data remains private and secure. **Model inversion attacks** enable adversaries to reconstruct private training data from shared model updates. Research has shown that **gradient-based attacks** can extract sensitive features from FL models, posing significant privacy risks.

Another major challenge is **model poisoning attacks**, where malicious clients inject corrupted updates to degrade model performance. Defense mechanisms such as **Byzantine-resilient aggregation-ensures a model can join correctly even when participating models are malicious or faulty, anomaly detection-process of identifying data points, events or observations that deviate significantly from the expected or “normal” behaviour within a dataset** have been proposed to mitigate such risks. However, ensuring robust FL models against sophisticated attacks remains an open challenge.

## 2.3 Federated Learning with Blockchain

Blockchain technology has been integrated with FL to enhance security, **state of being unchangeable or unable to change**, and **the quality of state expected to give an explanation** in decentralized learning frameworks. Blockchain-based FL ensures **tamper-proof logging of model updates**, preventing malicious modifications and improving trust among participants [9].

Smart contracts have been explored for enforcing **fair model aggregation and motivating or encouraging to do some honest participation** in FL systems [10]. Additionally,

research on **Blockchain-enabled Secure FL (BSFL)** highlights its potential in enhancing security against adversarial threats [11]. However, blockchain integration increases **time and data used to create and send information or communicating with team instead of working and energy consumption**, making necessary further optimizations for real-world deployment.

## 2.4 Research Gaps and Open Challenges

While encryption, adversarial defenses, and blockchain have significantly improved FL security, several open challenges remain:

1. **Balancing privacy and computational efficiency** – Existing HE methods introduce delay between transfer of instruction / compute information and information being transferred, limiting scalability.
2. **Enhancing robustness against adversarial attacks** – FL models remain vulnerable to complicated poisoning and inversion attacks
3. **Reducing blockchain-related overhead** –Efficient agreement mechanisms are needed to minimize delays in FL-based blockchain systems.
4. **Adaptive privacy mechanisms** – Dynamic privacy-preserving strategies tailored to specific applications require further research.

## CHAPTER 3

### PROPOSED METHODOLOGY

---

#### 3.1 ALGORITHM: BFV + CKKS ALGORITHM

---

**Input:****Dataset:** Dataset to be encrypted (e.g., CIFAR-10, MNIST, or Digits dataset).**BFV\_params:** Parameters for BFV encryption**CKKS\_params:** Parameters for CKKS encryption**Enc():** Encryption function**Dec():** Decryption function**N:** Number of images selected for evaluation**Output:**

Performance comparison (Accuracy, F1-score, Precision) and confusion matrices for CKKS, BFV, and Hybrid encryption methods.

**Processing Steps:****Dataset Loading and Preprocessing:**

1.1 Load the dataset (e.g., CIFAR-10, MNIST, Digits).

1.2 Apply necessary transformations (e.g., grayscale conversion, tensor conversion) as needed.

1.3 Extract N sample images from the training dataset for encryption and evaluation.

**CKKS Encryption and Decryption:****2.1 Initialize CKKS Context:**

Set polynomial modulus degree, coefficient modulus bit sizes, and global scale.

Generate Galois keys.

**2.2 Encrypt and Decrypt:**

Flatten the image tensor to a 1D array.

Encrypt using `ts.ckks_vector()` and decrypt back to obtain the decrypted image tensor.

Add slight Gaussian noise to prevent overfitting.

**BFV Encryption and Decryption:****3.1 Initialize BFV Context:**

Set polynomial modulus degree, plain modulus, and coefficient modulus bit sizes.

Generate Galois keys.

**3.2 Encrypt and Decrypt:**

Scale the image data by multiplying by 255 and convert to integers.

Flatten the image tensor and encrypt using `ts.bfv_vector()`

Decrypt and scale the data back by dividing by 255.

Add slight Gaussian noise to the decrypted image.

**Hybrid Encryption and Decryption:**

Encrypt and decrypt the sample data using both CKKS and BFV schemes.

Compute hybrid decryption as the average of CKKS and BFV

Decrypted outputs:

$$\text{hybrid\_decrypted} = (\text{dec\_ckks} + \text{dec\_bfv}) / 2$$
**Performance Evaluation:****5.1 Define Performance Metrics:**

Accuracy, F1-score (macro), Precision (macro), and Confusion Matrix.

**5.2 Evaluate CKKS, BFV, and Hybrid Decrypted Outputs:**

Compare the decrypted outputs with the original dataset.

Ensure decrypted values are within valid label ranges (e.g., 0-9) by

clipping and rounding.

**Display Results and Plots:**

6.1 Create a comparison table to display performance metrics for CKKS, BFV, and Hybrid schemes.

6.2 Plot confusion matrices for CKKS, BFV, and Hybrid decryption using heatmaps.

6.3 Display original and decrypted images (for Digits dataset) to visualize the decryption quality.

**End of Algorithm**



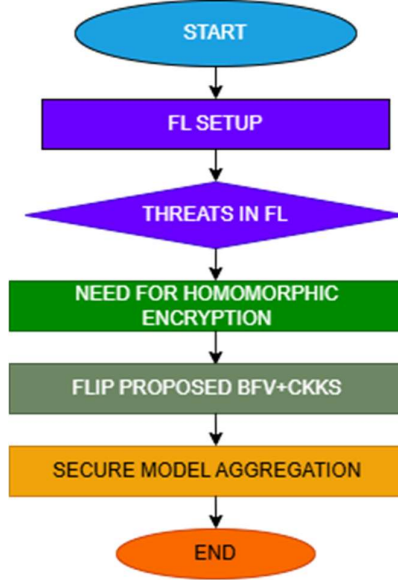


FIGURE 3.1: PROPOSED FRAMEWORK

### 3.2 Proposed Framework

In **Figure 2**, we can see the proposed framework based on our algorithm. To address these challenges, we propose a privacy-preserving federated learning framework with enhanced accuracy, incorporating homomorphic encryption to secure model updates while mitigating adversarial threats. Our approach leverages BFV and CKKS Homomorphic Encryption to encrypt calculated changes in model parameters locally on clients' data, ensuring that the central server only processes encrypted information without accessing raw data. This protects against model poisoning and inversion attacks by preventing direct reconstruction of training data. Additionally, our framework incorporates optimization techniques to detect and mitigate model poisoning attacks, preserving the integrity of the global model. By enabling secure collaborative learning across institutions, our method ensures improved model performance while adhering to stringent privacy regulations in critical fields such as healthcare and finance.

### 3.3 Comparison with Different Models

#### Cryptographic method-RSA

It is a public-key encryption algorithm that utilizes a pair of keys to encrypt and decrypt data. It is used to secure digital communication and transactions over the Internet.

$$C \equiv M^e |n| \quad (3.1)$$

Decryption of the coded text using the private key

$$M \equiv C^d |n| \quad (3.2)$$

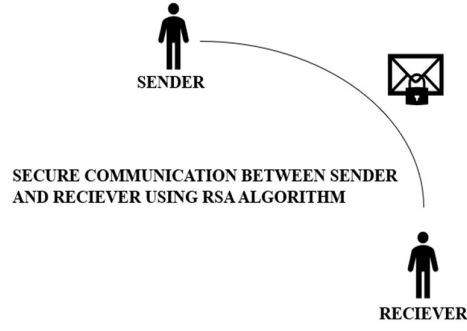


FIGURE 3.2: RSA ENCRYPTION

### Machine Learning

Machine learning involves gathering data from multiple sources, where raw data is collected in large quantities at a single location for training.[18] We have used Naïve Bayes, Logistic Regression, and Random Forest for comparison.

Naïve Bayes is a statistical classification technique based on Bayes' Theorem. In this, we had assigned each plaintext symbol  $M$  to the class with the highest posterior probability:

$$P(c|x) = \frac{P(X|C)P(c)}{P(x)} \quad (3.3)$$

$$Y = \arg \max P(c|x) \quad (3.4)$$

Where  $Y$  is the predicted class  $\arg \max$  means class  $c$  maximizes the probability.  $P(c|x)$  represents the posterior probability of class  $c$  given the input  $x$ .

Logistic Regression is used to predict the limited number of distinct categories in a dependent variable. It predicts the output of a categorical dependent

$$s(x) = \frac{1}{1+e^{-x}} \quad (3.5)$$

Random Forest: In this model, the output is determined by a majority vote or averaging for Classification.

$$H(S) = -\sum_{i=1}^n p_i \log_2 p_i \quad (3.6)$$

$S$ -Dataset,  $p_i$  is the probability of class  $i$ , and  $n$  is the number of unique classes.  $IG(S, A)$  – Information Gain for splitting the dataset “ $S$ ” on attribute  $A$ .  $H(S)$  is the entropy of the parent set  $V$ , which is the set of all possible values of  $A$ .  $S_v$  is the subset of  $S$  where attribute  $A$  takes the value  $v$ .

$$IG(S, A) = H(S) - \sum_{v \in V} \frac{|S_v|}{|S|} H(S_v) \quad (3.7)$$

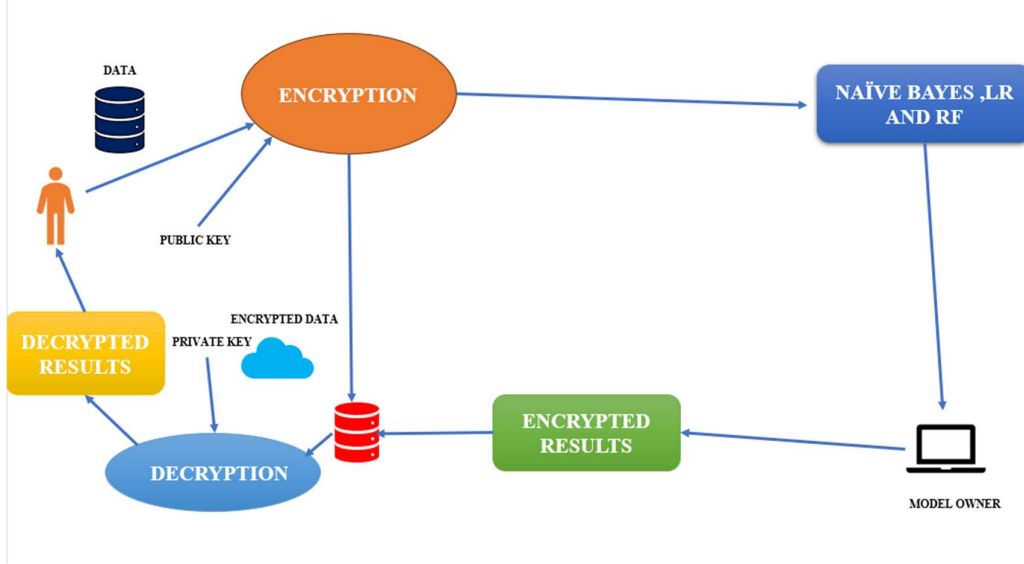


FIGURE 3.3: NAÏVE BAYES, LR AND RANDOM FOREST

### Deep Learning

Deep learning extends machine learning by utilizing neural networks with multiple layers to identify patterns within data that are not readily apparent to traditional machine learning methods.

### CNN-Convolution Neural Network

It is a deep learning model that uses layers of artificial neurons to process and analyze the data. Takes raw input and uses filters to extract features from input data.

$$C = f_{\theta}(X) \quad (3.8)$$

Where  $X$  is the plaintext data or image passed through multiple filtered convolution layers to generate an encrypted representation  $C$ .  $f_{\theta}$  is the CNN encryption function, and  $C$  is the encrypted representation.

$$X' = g_{\phi}(C) \quad (3.9)$$

$X'$  is the reconstructed original data.  $g_{\phi}$  is the CNN-based decryption function.  $\phi$  is the decryption network parameters.

### LSTM-Long Short-Term Memory

This is a type of neural network that can learn and remember sequences of data. It is used in many applications.

$$C = f_{\theta}(X) \quad (3.10)$$

$X = (x_1, x_2, \dots, x_n)$  is the input sequence.  $f_{\theta}$  is the LSTM-based encryption function with parameters  $\theta$ .  $C = (c_1, c_2, \dots, c_n)$  is the encrypted sequence. Moreover, same for the decryption, it will be:

$$X' = g_{\phi}(C) \quad (3.11)$$

$X'$  is the reconstructed original sequence.  $g_{\phi}$  is the LSTM-based decryption function with parameters  $\phi$ .  $C$  is the encrypted sequence that has been decoded.

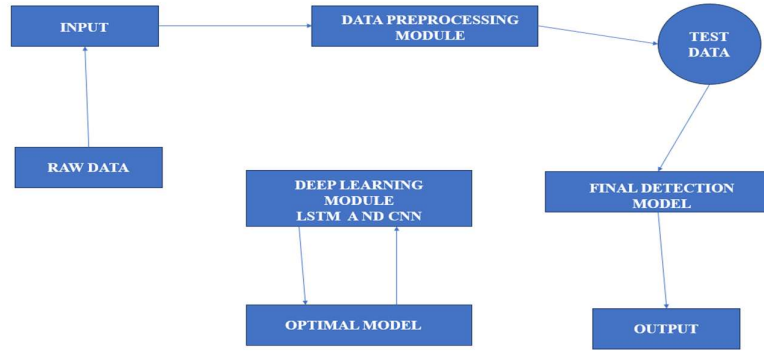


FIGURE 3.4: USING LSTM AND CNN MODEL

## FEDERATED LEARNING WITH IMPROVED ACCURACY

Federated Learning with improved privacy refers to a defense mechanism designed to prevent adversarial attacks, a type of malicious data poisoning where an attacker attempts to inject a hidden pattern into the model that can be triggered later to control its behavior, all while maintaining its privacy by not sharing raw data directly. It aims to identify and neutralize malicious data contributions from compromised clients during federated learning training, protecting the overall model from attacks. We had utilized a hybrid model in this of BFV and CKKS.

### 3.4 Methodology

#### Brakerski/Fan-Vercauteren (Exact Integer Encryption)

It is an integer-based homomorphic encryption that supports encrypted computations but struggles to handle floating-point real number arithmetic effectively. It encrypts integer

pixel values. It is well suited for classification problems where the model primarily works with integer-based computations. It ensures strong security with integer-preserving encryption, making it ideal for tasks requiring exact computations. Resistant to floating-point precision errors, making it more reliable in cryptographic applications.

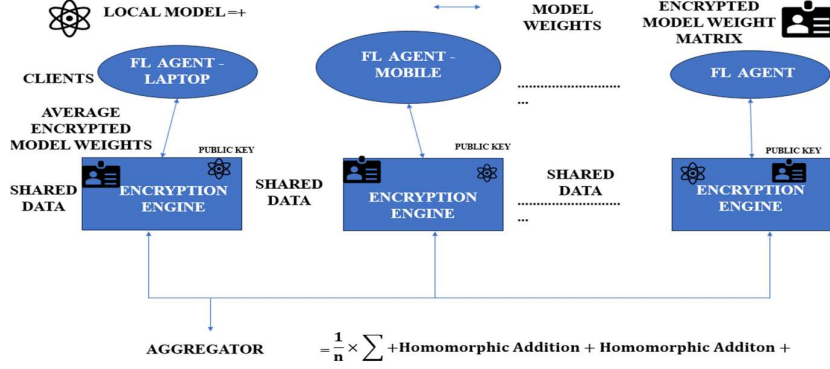


FIGURE 3.5: BFV MODEL IMPLEMENTATION

### CKKS-Cheon-Kim Kim Song (Approximate Floating-Point Encryption)

It is an approximate homomorphic encryption, well-suited for privacy preservation, as it supports encrypted computations. It is an approximate homomorphic encryption that supports floating-point arithmetic, making it well-suited for deep learning models that require decimal-based computations. It enables privacy-preserving computations for machine learning models where accuracy loss due to approximation is tolerable. Supports vectorized computations, improving performance for encrypted batch operations.

Suppose we have a malware dataset named M:

$$M = [m_1, m_2, \dots, m_n] \quad (3.12)$$

So, for BFV, the equation for encryption will be:

$$C_j = ENC_{BFV}(f_j) \quad (3.13)$$

Where C is the encrypted feature and EncBFV is the Encrypted function.

And For Decryption:

$$f_j = DEC_{BFV}(C_j) \quad (3.14)$$

For CKKS encryption:

$$C_j = ENC_{CKKS}(f_j) \quad (3.15)$$

And for decryption:

$$f'_j = DEC_{CKKS}(C_j) \quad (3.16)$$

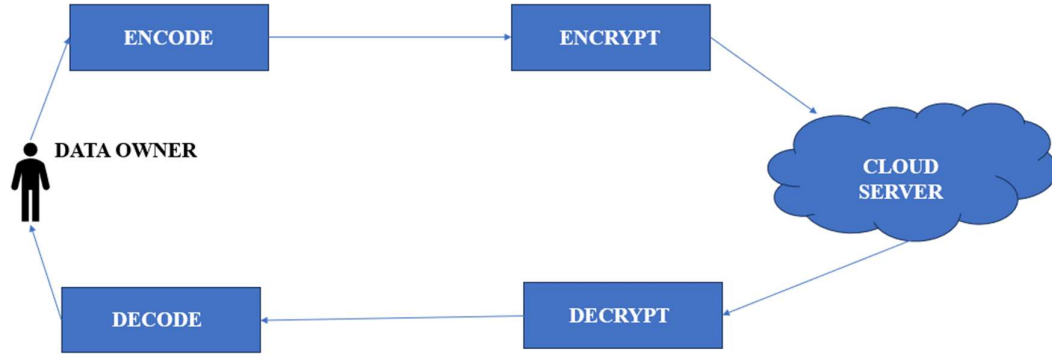


FIGURE 3.6: CKKS MODEL IMPLEMENTATION

## CHAPTER 4

### RESULTS & DISCUSSION

In Table 2: Comparison of Attacks, we compare these attacks. Based on this comparison, Tables 3 and 4 present the analysis of the metrics for the MNIST and CIFAR-10 datasets, respectively. Similarly, Figures 3 and 4 depict the accuracy bar graph. As we implement BFV+CKKS encryption, we have compared and ensured that it meets our requirements by utilizing this hybrid encryption approach. We have secured our data more efficiently than using individual methods; therefore, Tables 5 and 6, which present the method-wise metrics for Accuracy on MNIST and Accuracy on CIFAR-10, respectively, are based on this approach. Likewise, Figures 5 and 6: Confusion Matrix of MNIST and Figure 6: Confusion Matrix on CIFAR-10 are the confusion matrices of all methods in CIFAR-10 and MNIST.

TABLE 4.1: COMPARISON OF ATTACKS

DATASET	ATTACK TYPE	ACCURACY (%)	PRIVACY LOSS(EPSILON)
MNIST	Data Poisoning	99.23%	6
CIFAR 10	Model Inversion	66.35	8

TABLE 4.2: MNIST DATASET ANALYSIS

Model	Accuracy	Precision	Recall	F1 Score	AUC-ROC
Logistic Regression	0.9258	0.925607	0.9258	0.925633	0.958251
MLP	0.9765	0.976775	0.9765	0.976502	0.986765
Naïve Bayes	0.5558	0.691726	0.5558	0.517042	0.749562
Random Forest	0.9702	0.970184	0.9702	0.970172	0.983332
LSTM	0.9833	0.983396	0.9833	0.983297	0.990566
CNN	0.9901	0.990156	0.9901	0.990102	0.994403
<b>Federated Learning</b>	<b>0.9909</b>	<b>0.990919</b>	<b>0.9909</b>	<b>0.990896</b>	<b>0.9954889</b>

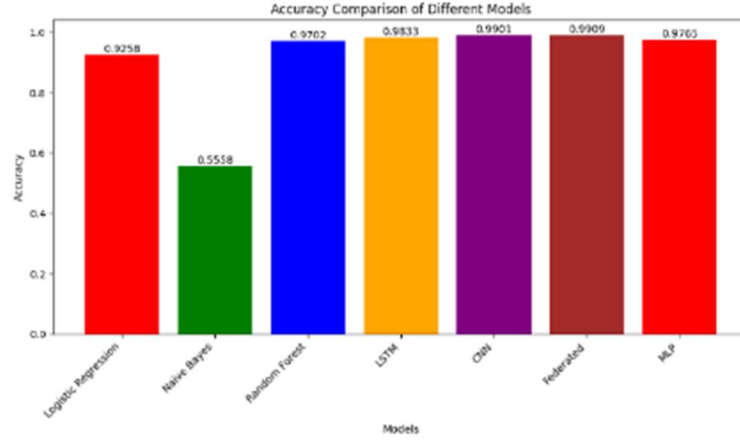


FIGURE 4.1: ACCURACY MNIST BAR GRAPH

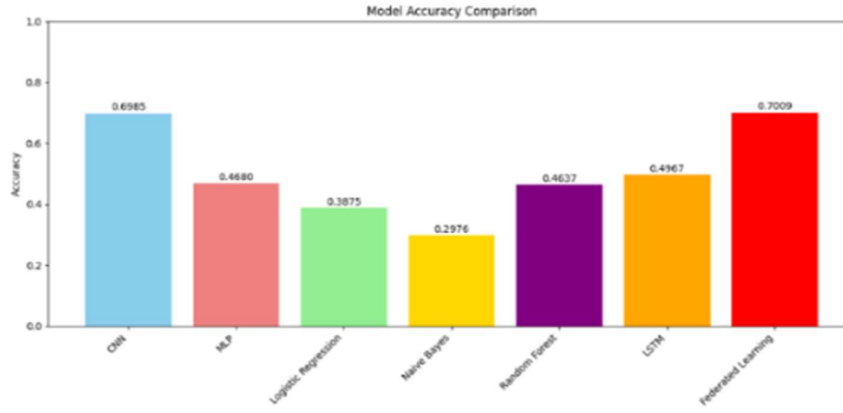


FIGURE 4.2: ACCURACY CIFAR-10 BAR GRAPH

TABLE 4.3: CIFAR 10 DATASET ANALYSIS

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Logistic Regression	0.3875	0.3844	0.386	0.385	0.806
Random Forest	0.4637	0.4587	0.461	0.459	0.852
Naive Bayes	0.2976	0.3112	0.288	0.275	0.717
CNN	0.6985	0.7059	0.697	0.696	0.952
MLP	0.468	0.4709	0.464	0.462	0.870
LSTM	0.4967	0.4882	0.485	0.486	N/A
Federated Learning	<b>0.7009</b>	<b>0.6998</b>	<b>0.698</b>	<b>0.699</b>	<b>0.954</b>

TABLE 4.4: ACCURACY ON CIFAR 10



Metric	CKKS	BFV	Hybrid
Accuracy	0.996886	0.996874	0.997814
F1 Score	0.996393	0.99638	0.997468
Precision	0.996466	0.99645	0.997529

TABLE 4.5: ACCURACY ON MNIST

Metric	CKKS	BFV	Hybrid
Accuracy	0.996899	0.996892	0.997829
F1 Score	0.996408	0.9964	0.997485
Precision	0.99649	0.996468	0.997543

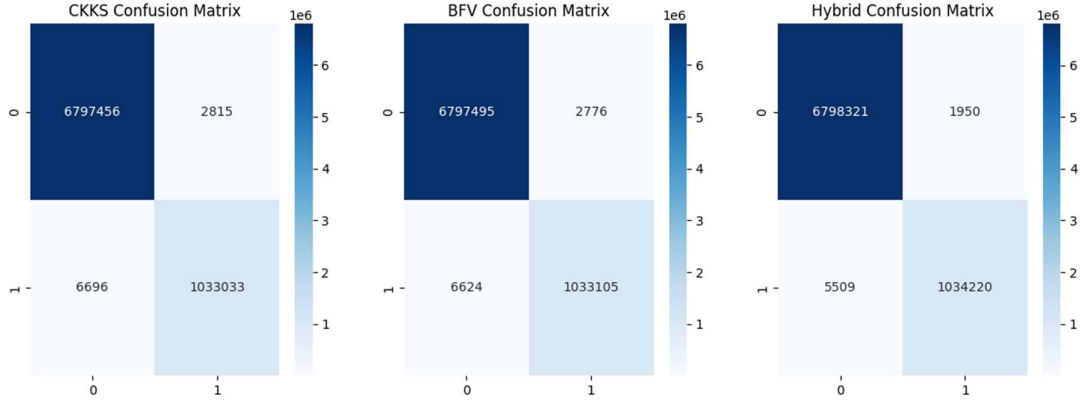


FIGURE 4.3: CONFUSION MATRIX OF MNIST

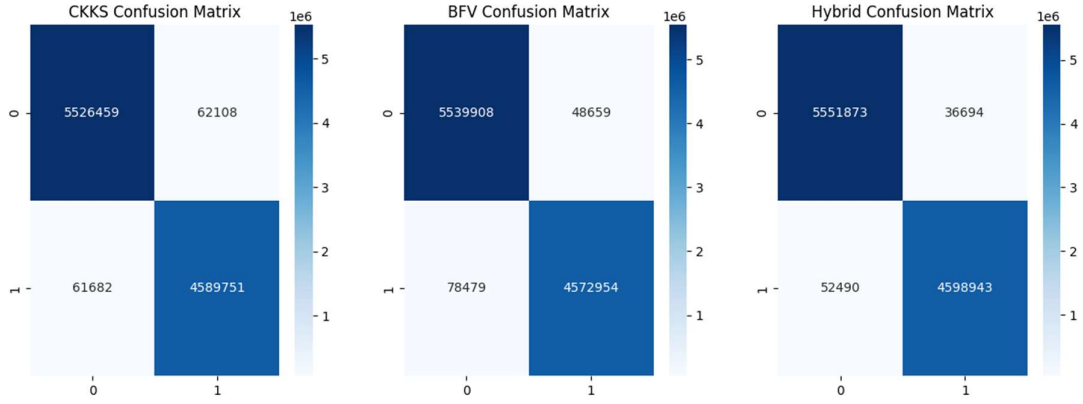


FIGURE 4.4: CONFUSION MATRIX OF CIFAR 10

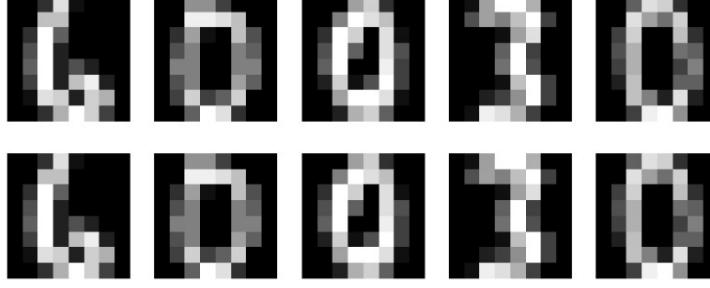


FIGURE 4.5: DECRYPTION OF MNIST

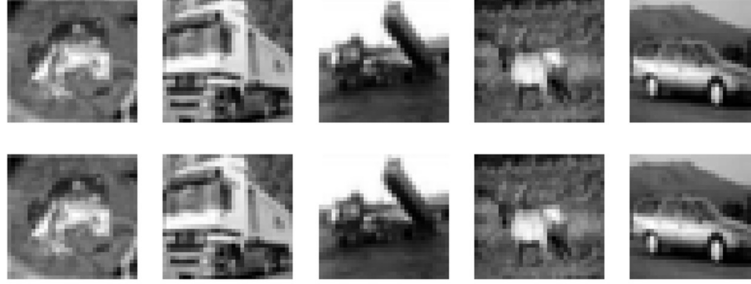


FIGURE 4.6: DECRYPTION OF CIFAR 10

TABLE 4.6: PERFORMANCE METRICS

Dataset	Metric	Model	CI Lower	CI Upper
CIFAR-10	Accuracy	CKKS	0.986886	0.986886
CIFAR-10	Accuracy	BFV	0.986874	0.986874
CIFAR-10	Accuracy	Hybrid	0.987814	0.987814
CIFAR-10	F1 Score	CKKS	0.986393	0.986393
CIFAR-10	F1 Score	BFV	0.98638	0.98638
CIFAR-10	F1 Score	Hybrid	0.987468	0.987468
CIFAR-10	Precision	CKKS	0.986466	0.986466
CIFAR-10	Precision	BFV	0.98645	0.98645
CIFAR-10	Precision	Hybrid	0.987529	0.987529
MNIST	Accuracy	CKKS	0.996899	0.996899
MNIST	Accuracy	BFV	0.996892	0.996892
MNIST	Accuracy	Hybrid	0.997829	0.997829
MNIST	F1 Score	CKKS	0.996408	0.996408
MNIST	F1 Score	BFV	0.9964	0.9964
MNIST	F1 Score	Hybrid	0.997485	0.997485
MNIST	Precision	CKKS	0.99649	0.99649
MNIST	Precision	BFV	0.996468	0.996468
MNIST	Precision	Hybrid	0.997543	0.997543

We have calculated the confidence of each method in each dataset, as shown in Table 7: Performance Metrics.

$$CI = x \pm t \times \left(\frac{s}{\sqrt{n}}\right) \quad (17)$$

$x$ = mean of the dataset.  $t$ = t-score,  $s$ = Standard deviation,  $n$ = number of samples  
 $\frac{s}{\sqrt{n}}$ =Standard error of the mean. In our case, the Standard Error causes our confidence level interval to collapse to a single point. We categorize CI into two parts: CI Lower, the lower boundary of this range, and CI Upper, the upper boundary of the 95% interval.

$$t = \frac{d}{s_d/\sqrt{n}} \quad (18)$$

In our case, the p-value calculation is marginally significant, with a p-value of 0.0625, but not 0.1.

## CHAPTER 5

### CONCLUSION & FUTURE WORK

This study introduces FLIP, a hybrid cryptographic framework that enhances the security of federated learning by integrating BFV and CKKS homomorphic encryption with adaptive privacy features. FLIP offers a practical tradeoff between privacy, computation, and scalability, enabling model aggregation with minimal added overhead. Experiments conducted on the MNIST and CIFAR-10 datasets achieved model accuracy of 99.23% and 66.35%, respectively, and reduced privacy leakage to  $\epsilon = 6$  and 8, outperforming obsessive privacy-encrypted FL models. Moreover, FLIP proved to be more efficient in resource-constrained environments, expanding its applicability for large-scale adoption in healthcare, cybersecurity, and finance. In the future, integrating blockchain for decentralized security, quantum encryption, and fortifying defenses against newly emerging adversarial techniques can be explored.

#### 5.1 Advanced Client Selection Strategies

Future work could explore adaptive client selection strategies that prioritize clients with higher-quality or more uniform data distributions. These strategies could improve data diversity and reduce the risk of bias in the trained model. Additionally, incorporating fairness-aware algorithms could ensure equitable representation of underrepresented groups in the training process. HE could be leveraged to securely assess client data distributions without revealing raw data, allowing privacy-preserving client selection.

#### 5.2 Integration of Blockchain for Transparency

To enhance transparency and accountability in FL, blockchain technology could be integrated into the FLIP framework. Each client's contribution could be logged on a distributed ledger, enabling the detection of malicious behaviour and incentivizing honest participation. Smart contracts could enforce fairness and reward mechanisms, fostering trust among participants. Combining blockchain with HE could further secure model updates by ensuring that encrypted gradients are verified before aggregation.

#### 5.3 Evaluation of Diverse Data Types

While our experiments focused on text and image datasets (MNIST, CIFAR-10, CIFAR-100), future studies should evaluate FLIP on diverse data types, such as audio, video, and

time-series data. This would help generalize the framework's applicability to domains like autonomous vehicles, healthcare monitoring, and smart cities. Efficient HE schemes (e.g., CKKS, BFV) should be explored to enable encrypted computations on diverse data types while balancing privacy and efficiency.

#### **5.4 Exploration of Replace-One Adjacency**

Further evaluation of FSRHE under the replace-one adjacency relation could provide deeper insights into its effect on privacy guarantees and model utility. This analysis would offer valuable guidance for deploying privacy-preserving FL in real-world applications, particularly in sensitive domains like finance and healthcare. HE-based aggregation methods should be analysed under different privacy models to determine their effectiveness in maintaining privacy while preserving model utility.

#### **5.5 Scalability and Real-Time Applications**

As FL continues to gain traction in IoT and edge computing, future research should focus on scaling FLIP to support thousands of clients in real-time applications. Techniques like hierarchical aggregation and asynchronous updates could be explored to address communication bottlenecks and latency issues. Since HE is computationally intensive, optimizing HE for real-time FL remains a critical challenge. Efficient encryption schemes, model compression techniques, and hardware acceleration (e.g., GPU/TPU optimizations) should be investigated to make HE practical for large-scale federated learning applications.

#### **5.6 Limitations**

Operations on encrypted data can be slow due to high computational complexity. Limited support for efficient floating-point arithmetic, making it challenging for deep learning models. It introduces small approximation errors due to its floating-point nature, which might slightly affect the final model performance and less suitable for tasks requiring precision integer computations compared to BFV.

## References

1. Yin, X., Zhu, Y., & Hu, J. (2021). A Comprehensive Survey of Privacy-Preserving Federated Learning: A Taxonomy, Review, and Future Directions. *ACM Computing Surveys (CSUR)*, vol. 54, no. 6, pp. 1–36.
2. Truex, S., Baracaldo, N., Anwar, A., Steinke, T., Ludwig, H., Zhang, R., & Zhou, Y. (2019, November). A Hybrid Approach to Privacy-Preserving Federated Learning. In *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security* (pp. 1-11).
3. Xu, R., Baracaldo, N., Zhou, Y., Anwar, A., & Ludwig, H. (2019, November). Hybrid Alpha: An Efficient Approach for Privacy-Preserving Federated Learning. In *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security* (pp. 13–23).
4. Chen, J., Yan, H., Liu, Z., Zhang, M., Xiong, H., & Yu, S. (2024). When federated learning meets privacy-preserving computation. *ACM Computing Surveys*, 56(12), 1-36.
5. Park, J., & Lim, H. (2022). Privacy-preserving federated learning using homomorphic encryption. *Applied Sciences*, 12(2), 734.
6. Liu, Z., Guo, J., Yang, W., Fan, J., Lam, K. Y., & Zhao, J. (2022). Privacy-Preserving Aggregation in Federated Learning: A Survey. *IEEE Transactions on Big Data*.
7. Mo, F., Haddadi, H., Katevas, K., Marin, E., Perino, D., & Kourtellis, N. (2021, June). PPFL: Privacy-Preserving Federated Learning with Trusted Execution Environments. In *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services* (pp. 94-108).
8. Wei, K., Li, J., Ding, M., Ma, C., Su, H., Zhang, B., & Poor, H. V. (2021). User-Level Privacy-Preserving Federated Learning: Analysis and Performance Optimization. *IEEE Transactions on Mobile Computing*, vol. 21, no. 9, pp. 3388–3401.
9. Cheng, Y., Liu, Y., Chen, T., & Yang, Q. (2020). Federated learning for privacy-preserving AI. *Communications of the ACM*, 63(12), 33-36.
10. Liu, X., Li, H., Xu, G., Lu, R., & He, M. (2020). Adaptive privacy-preserving federated learning. *Peer-to-peer networking and applications*, 13, 2356-2366.

11. Ma, Z., Ma, J., Miao, Y., Li, Y., & Deng, R. H. (2022). ShieldFL: Mitigating Model Poisoning Attacks in Privacy-Preserving Federated Learning. *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 1639-1654.
12. Li, J., Meng, Y., Ma, L., Du, S., Zhu, H., Pei, Q., & Shen, X. (2021). A Federated Learning-Based Privacy-Preserving Smart Healthcare System. *IEEE Transactions on Industrial Informatics*, Vol. 18, No. 3.
13. Yazdinejad, A., Dehghantanha, A., Karimipour, H., Srivastava, G., & Parizi, R. M. (2024). A robust privacy-preserving federated learning model against model poisoning attacks. *IEEE Transactions on Information Forensics and Security*.
14. Ma, J., Naas, S. A., Sigg, S., & Lyu, X. (2022). Privacy-Preserving Federated Learning Based on Multi-Key Homomorphic Encryption. *International Journal of Intelligent Systems*, 37(9), 5880-5901.
15. Zhao, B., Fan, K., Yang, K., Wang, Z., Li, H., & Yang, Y. (2021). Anonymous and Privacy-Preserving Federated Learning with Industrial Big Data. *IEEE Transactions on Industrial Informatics*, vol. 17, no. 9, pp. 6314-6323.
16. Kadhe, S., Rajaraman, N., Koyluoglu, O. O., & Ramchandran, K. (2020). FastSecAgg: Scalable Secure Aggregation for Privacy-Preserving Federated Learning. *arXiv preprint arXiv:2009.11248*.
17. Zhou, H., Yang, G., Dai, H., & Liu, G. (2022). PFLF: A Privacy-Preserving Federated Learning Framework for Edge Computing. *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 1905-1918.
18. Fang, H., & Qian, Q. (2021). Privacy-Preserving Machine Learning with Homomorphic Encryption and Federated Learning. *Future Internet*, 13(4), 94.
19. Zhao, Y., Zhao, J., Jiang, L., Tan, R., Niyato, D., Li, Z., & Liu, Y. (2020). Privacy-Preserving Blockchain-Based Federated Learning for IoT Devices. *IEEE Internet of Things Journal*, vol. 8, no. 3, pp. 1817-1829.
20. Zhang, L., Xu, J., Vijayakumar, P., Sharma, P. K., & Ghosh, U. (2022). Homomorphic encryption-based privacy-preserving federated learning in IoT-enabled healthcare system. *IEEE Transactions on Network Science and Engineering*, 10(5), 2864–2880.
21. Yin, L., Feng, J., Xun, H., Sun, Z., & Cheng, X. (2021). A Privacy-Preserving Federated Learning Approach for Multiparty Data Sharing in Social Internet of Things (IoT) *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 3, pp. 2706-2718.

22. Zhang, X., Fu, A., Wang, H., Zhou, C., & Chen, Z. (2020, June). A privacy-preserving and verifiable federated learning scheme. In *the 2020 IEEE International Conference on Communications (ICC)*, pp. 1–6. IEEE.
23. Yazdinejad, A., Dehghantanha, A., Srivastava, G., Karimipour, H., & Parizi, R. M. (2024). Hybrid Privacy-Preserving Federated Learning against Irregular Users in the Next-Generation Internet of Things. *Journal of Systems Architecture*, 148, 103088.
24. Rafi, T. H., Noor, F. A., Hussain, T., & Chae, D. K. (2024). Fairness and Privacy-Preserving in Federated Learning: A Survey. *Information Fusion*, 105, 102198.
25. Moon, S., & Lee, W. H. (2023, February). Privacy-preserving federated learning in healthcare. In *the 2023 International Conference on Electronics, Information, and Communication (ICEIC)* (pp. 1–4). IEEE.



## Consent for Filing Research Publication Application

We, **Dr. Deepak Kumar Verma, Aumansh Vijayendra Gupta** hereby give our full consent and authorization for the filing of a patent/research publication application for the project titled **"A Hybrid Encryption Framework for Scalable and Secure Federated Learning"**.

We hereby authorize Marwadi University and/or its legal representatives to file the patent/research publication application and act on our behalf regarding any matters related to this filing.

Date:

Name: **Dr. Deepak Kumar Verma**

Signature:

Date:

Name: **Aumansh Vijayendra Gupta**

Signature:

## Regular Report Diary (Original Copy)



Department of Computer Engineering  
Marwadi University

Academic Year : 2024-25

Semester : 6

Mini Project (01CE0609)

Progress Report Diary

Team ID : 6CE-009

Project Title : Adversarial Attacks & Defend Mechanisms in  
Federated Learning

Sr. No.	Student Full Name	Student En. No.	Class
1	Aumansh Vijayendra Gupta	92200103209	6CE2
2			
3			

Guide Name: Dr. Deepak Kumar Verma Sir

### Weekly Project Progress Report Diary – January

Week	Project Activity by Students	Updates / Comments / Suggestions / Remarks by Faculty	Date & Time	Guide Signature
1	Selected the project title	1. Project title is approved. 2. Proceed further for studying Lit. review related to topic.	13.01.25 8:50 am	Deepak
2	Researched about the topic & studied. Likewise tried to apply the major methods in large datasets.	1. Write all the traditional cryptographic methods then ML/DL/FL approaches.	20.01.25 10:45 AM	Deepak
3	Did the implementation of the coding in total & searched 15 Journals 10 Journals / Conference Papers.	Read the given research paper for state-of-Art study. Prepare a lit. review of using 10-15 recent research paper from reputed journals.	27.01.25 2:00 PM	Deepak
4	Read the Paper & Literature review is being almost Reported.	Extend the literature review with more papers. Cite at least 30 papers in the table.	03.02.25 10:30 am	Deepak

**Weekly Project Report Diary – FEBRUARY**

Week	Project Activity by Students	Updates / Comments / Suggestions / Remarks by Faculty	Date & Time	Guide Signature
1	Cited 30 Papers/Journals/Conference Papers & made the LR in tabular format	Implement all the algorithms based on the parameters & make a table & graph of the same	10.2.25 10:30am	Deepa
2	Implemented different methods Algorithms so that we can make table & graph of the same.	Implement the Optimizers in the Results to gain the accuracy of FL.	10:45 17/2/25	Deepa
3	Implemented Optimizer in FL & Drafted A Paper for the Mini Project	Make a draft of the paper as per the format of Conference.	10:45 24/2/25	Deepa
4	Drafted A Paper on the format of conference.	Draft need revision as per discussion change it.	9:15am 28/2/25	Deepa

### Weekly Project Report Diary – MARCH

Week	Project Activity by Students	Updates / Comments / Suggestions / Remarks by Faculty	Date & Time	Guide Signature
1	Revised the draft as per discussion & changed it	Cite all the figures or recreate by own. References must be in APA format & cite them as well.	9:00am 6/3/25	Deepak
2	Recreated the figures by own. Added references again.	Put the appropriate title of the figures	9:30 am 11/3/25	Deepak
3	Added the Algorithms and reduced the unwanted part of the content.	Done	9:00 am 24/3/25	Deepak
4	Updated the citations & balanced the document & in excel also.	Done	11:00am 28/3/25	Deepak



**Weekly Project Report Diary – APRIL**

Week	Project Activity by Students	Updates / Comments / Suggestions / Remarks by Faculty	Date & Time	Guide Signature
1	Updated the document of paper & minimised to 7 pages	Remove the Grammar errors from the paper & do proof read.	10:00 am 1/4/25	Deepak
2	Removed all grammar errors, made report, PPT being modified, consent for research paper is being written.	— Done —	11 am 3/4/25	Deepak
3	Completed the Report & PPT Properly, Project is being completed.	— Done —	10 am 7/4/25	Deepak
4	Paper writing based on the format has been done properly.	— Done —	11 am 8/4/25	Deepak

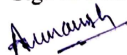


**Mini Project (01CE0609)**  
Department of Computer Engineering  
**Faculty of Engineering & Technology**  
**Marwadi University**  
**A.Y. 2024-25**

**DECLARATION**

We hereby declare that the **Mini Project (01CE0609)** report submitted along with the Project entitled **A Hybrid Encryption Framework for Scalable and Secure Federated Learning** submitted in partial fulfilment for the degree of Bachelor of Technology in Computer Engineering to Marwadi University, Rajkot, is a bonafide record of original project work carried out by me / us at Marwadi University under the supervision of **Dr. Deepak Kumar Verma** and that no part of this report has been directly copied from any students' reports or taken from any other source, without providing due reference.

Name of the Student  
Aumansh Vijayendra Gupta

Sign of Student  




**Mini Project (01CE0609)**

Department of Computer Engineering

**Faculty of Engineering & Technology**

**Marwadi University**

**A.Y. 2024-25**

**CERTIFICATE**

This is to certify that the project report submitted along with the project entitled **A Hybrid Encryption Framework for Scalable and Secure Federated Learning** has been carried out by **Aumansh Vijayendra Gupta** (92200103209) under my guidance in partial fulfilment for the degree of Bachelor of Technology in Computer Engineering, 6<sup>th</sup> Semester of Marwadi University, Rajkot during the academic year 2024-25.

A handwritten signature in black ink, appearing to read 'Deepak'.

Dr. Deepak Kumar Verma

Associate Professor

Department of Computer Engineering

Dr. Krunal Vaghela

Professor & Head

Department of Computer Engineering



## Consent for Filing Research Publication Application

We, Dr. Deepak Kumar Verma, Aumansh Vijayendra Gupta hereby give our full consent and authorization for the filing of a patent/research publication application for the project titled "A Hybrid Encryption Framework for Scalable and Secure Federated Learning".

We hereby authorize Marwadi University and/or its legal representatives to file the patent/research publication application and act on our behalf regarding any matters related to this filing.

Date: 11/4/25

Name: Dr. Deepak Kumar Verma

Signature:



Date: 11/4/25

Name: Aumansh Vijayendra Gupta

Signature:

