Digital Communications and Networks 9 (2023) 981–989

Contents lists available at ScienceDirect

Digital Communications and Networks

journal homepage: www.keaipublishing.com/dcan

VPFL: A verifiable privacy-preserving federated learning scheme for edge computing systems

Jiale Zhang [a,*,1], Yue Liu [b,1], Di Wu [c], Shuai Lou [b], Bing Chen [d], Shui Yu [e]

[a] School of Information Engineering, Yangzhou University, Yangzhou, 225009, China
[b] Water Conservancy and Civil Engineering College, Inner Mongolia Agricultural University, Hohhot, 010018, China
[c] Deakin Blockchain Innovation Lab, School of Information Technology, Deakin University, Melbourne, VIC, 3125, Australia
[d] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, 211106, China
[e] School of Computer Science, University of Technology Sydney, Sydney, 2007, Australia

ARTICLE INFO

Keywords:
Federated learning
Edge computing
Privacy-preserving
Verifiable aggregation
Homomorphic cryptosystem

ABSTRACT

Federated learning for edge computing is a promising solution in the data booming era, which leverages the computation ability of each edge device to train local models and only shares the model gradients to the central server. However, the frequently transmitted local gradients could also leak the participants' private data. To protect the privacy of local training data, lots of cryptographic-based Privacy-Preserving Federated Learning (PPFL) schemes have been proposed. However, due to the constrained resource nature of mobile devices and complex cryptographic operations, traditional PPFL schemes fail to provide efficient data confidentiality and lightweight integrity verification simultaneously. To tackle this problem, we propose a Verifiable Privacy-preserving Federated Learning scheme (VPFL) for edge computing systems to prevent local gradients from leaking over the transmission stage. Firstly, we combine the Distributed Selective Stochastic Gradient Descent (DSSGD) method with Paillier homomorphic cryptosystem to achieve the distributed encryption functionality, so as to reduce the computation cost of the complex cryptosystem. Secondly, we further present an online/offline signature method to realize the lightweight gradients integrity verification, where the offline part can be securely outsourced to the edge server. Comprehensive security analysis demonstrates the proposed VPFL can achieve data confidentiality, authentication, and integrity. At last, we evaluate both communication overhead and computation cost of the proposed VPFL scheme, the experimental results have shown VPFL has low computation costs and communication overheads while maintaining high training accuracy.

1. Introduction

With the development of Artificial Intelligence (AI) chips and algorithms, Internet-of-Things (IoT) infrastructures are widely deployed in multiple areas including vehicle networks [1], medical devices [2], smart grids [3], and smart cities [4]. However, the traditional cloud computing architecture is hard to meet the requirements during the data-processing for the real-time services due to the limitation of network bandwidth and privacy concerns [5]. To overcome the limitations, edge computing [6,7] has been introduced into the computation paradigm of IoT. It introduces an Edge Server (*ES*) for local processing which processes the raw data with aggregation, mining, or sharing tasks. In edge computing, *ES* plays an important role as a preliminary processing device that provides sufficient local services through the whole cloud service architecture [8,9]. Therefore, the bottleneck of computation and communication for traditional cloud-based architecture can be solved.

Recently, to achieve edge intelligence, Federated Learning (FL) has been envisioned as a novel technology to provide the ability on processing big data and protecting user privacy [10,11]. As shown in Fig. 1, FL only requires the end devices to train a local model on each device and upload the local model updates such as gradients or weights to the central server [12]. Although the FL can provide the basic privacy guarantee, participants' local training data is still under high leaking risks if they upload the model parameters to an untrusted server [13], such as gradient inference attacks [14] or model inversion attacks [15]. Recent research demonstrates that the untrusted server has the ability to recover

* Corresponding author.
E-mail addresses: jialezhang@yzu.edu.cn (J. Zhang), 737396997@qq.com (Y. Liu), di.wu@deakin.edu.au (D. Wu), 865822716@qq.com (S. Lou), cb_china@nuaa.edu.cn (B. Chen), shui.yu@uts.edu.au (S. Yu).
[1] Jiale Zhang and Yue Liu have same contribution.

https://doi.org/10.1016/j.dcan.2022.05.010
Received 8 August 2021; Received in revised form 30 April 2022; Accepted 13 May 2022
Available online 24 May 2022

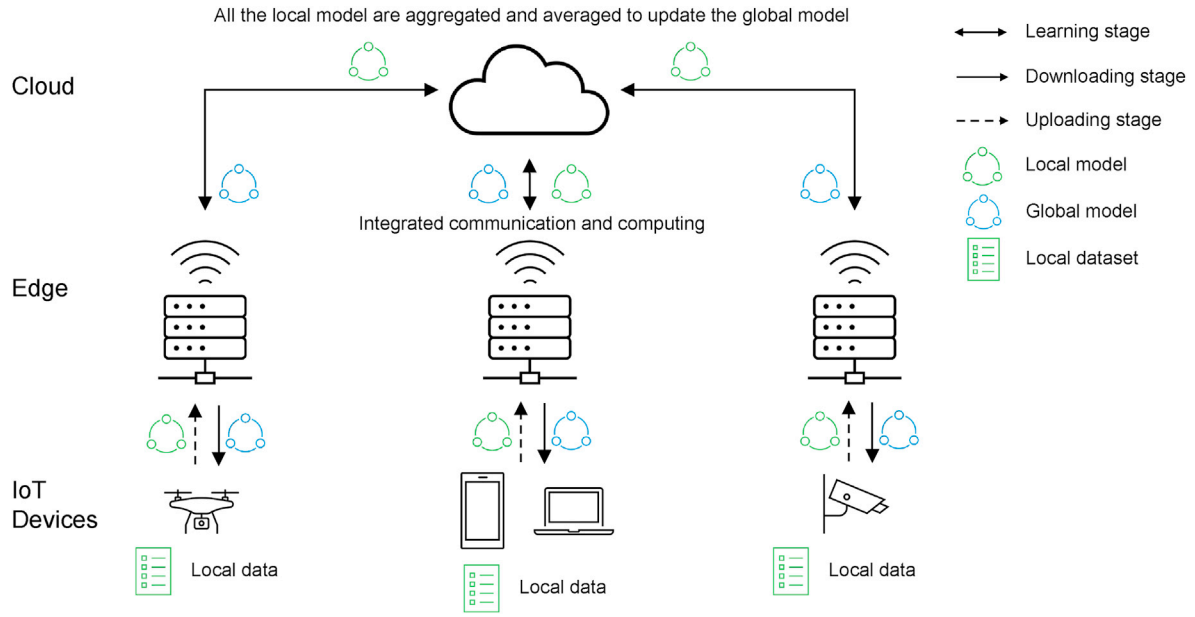


Fig. 1. Federated learning framework with edge computing.

the user's local data from the uploaded model gradients or weights [16, 17]. The untrusted server can observe the model structure, initial parameters, and the changes of the gradients or training labels, so as to reveal private information from participants by using adversarial techniques [18] [–] [21].

Another security challenge in federated learning for edge computing is that the participants cannot be fully trusted either [22]. The untrusted nodes may leak the data from other participants, poisoning the global models and destroying the gradients aggregation protocol [23]. Additionally, the malicious users may also be able to listen to the communication channel among the participants, and try to modify the upload gradients, create fake signatures, or destroy the normal gradients transmission procedure by initialing a relay attack. To protect the user privacy from untrusted nodes, many Privacy-Preserving Data Aggregation (PPDA) approaches have been proposed [24–31]. Most of the approaches apply homomorphic cryptosystem to demonstrate the specific functions, such as sum, max or min to guarantee the data privacy.

However, there are still many practical challenges that need to solve when applying PPDA schemes to federated learning scenarios. The first challenge is to overcome the high communication delay when processing the real-time training tasks, based on the nature of frequent communication for edge computing IoT systems. Secondly, data authentication and verification schemes are crucial to prevent external attacks such as modifying gradients or labels, forging signatures, and replaying attacks. Lastly, to process a large amount of authentication and verification requests, huge computation resources are needed which exhausts the resource-constrained IoT systems. Therefore, a novel lightweight PPDA approach is desirable to reduce the computational cost and fulfill the privacy-preserving requirements.

In this paper, we propose a verifiable privacy-preserving federated learning scheme, named VPFL, for edge computing systems by combining the Paillier homomorphic cryptosystem, DSSGD approach, and online/offline signature method. Firstly, the DSSGD approach is used to realize the distributed encryption property during the local training stage, so as to reduce the computation costs of the cryptosystem. Then, the security of the server-side gradients aggregation can be guaranteed by using the additive homomorphic property of the Paillier cryptosystem. Moreover, to further verify the gradients' integrity in federated learning, a lightweight online/offline signature method is utilized to shift the integrity verification tasks into the edge server. Therefore, the heavy computation costs can be reduced and more suitable for resource-constrained edge computing framework. The main contributions of this paper are as follows.

- We propose a verifiable privacy-preserving federated learning scheme (VPFL) based on the Paillier homomorphic cryptosystem and distributed selective SGD method. Each participant's gradient vector can be divided into different shards and encrypted in a distributed manner.
- We present an online/offline signature method to realize lightweight integrity verification during the gradients transmission stage. The mobile devices are only required to execute simple online processes while most time-consumed offline operations are safely outsourced to the edge servers.
- We provide a comprehensive security analysis to illustrate how the VPFL scheme can achieve data confidentiality, authentication, and verifiability. Exhaustive experimental evaluations demonstrate that the VPFL scheme can realize high performance on accuracy and efficiency.

The rest of our paper is organized as follows. The related works are summarized in Section 2, followed by the preliminaries in Section 3. The proposed VPFL is detailed in Section 4. Section 5 and Section 6 analysis the security and system performance, respectively. Section 7 gives the summary and future work.

2. Related work

For the privacy-preserving data aggregation and integrity verification methods, previous works mainly focused on applying the homomorphic cryptosystem to secure data aggregation in different application scenarios [24–30]. Some researchers applied different homomorphic encryption methods to construct PPDA approaches, such as additive and multiplicative homomorphism [24–26]. To prevent internal attacks, Fan et al. [27] added the blinding factors in the encryption step to enhance the privacy guarantee. Additionally, Ni et al. [28,29] proposed PPDA approaches with random noisy technique and trapdoor hash function, which can verify data integrity during ciphertexts transmission.

To protect the privacy of participant's local training data, many PPFL approaches have been proposed [32–38]. For examples, Xu et al. [32] proposed HybridAlpha which applies functional encryption-based SMC protocol for PPFL. In addition, Chen et al. [33] designed a

training-integrity protocol based on the Trusted Execution Environment (TEE) to prevent attacks. Wei et al. [34] proposed a user-level differential privacy scheme by adding the crafted noises to local gradients during the uploading procedure. Li et al. [35] proposed a chained Secure Multi-party Computing-based federated learning to prevent the data leaking of the shared models. To prevent the data leak in industrial artificial intelligence, Hao et al. [36] proposed privacy-enhanced federated learning which is able to prevent the data leaking via colluded entities.

Besides, to guarantee the integrity of local model gradients, the Verifiable Federated Learning (VFL) was proposed by Fu et al. [37] which applied Lagrange interpolation to verify the aggregated gradients. Guo et al. [38] proposed VERIFL which is specially designed for limited bandwidth and high-dimensional gradients federated learning participants. Zhang et al. [39] presented a lightweight batch encryption method in a federated learning framework, which can encode a batch of gradients into a long integer data type. In Ref. [40], the authors presented a privacy-preserving and verifiable federated learning scheme to process the shared gradients by combining the Chinese Remainder Theorem and Paillier homomorphic encryption. However, the aforementioned approaches only consider user privacy and data integration for federated learning, and the computational cost of the system had been ignored.

Computation costs of cryptography have drawn the attention of researchers recently, where many PPDA methods with low costs of cryptography have been proposed. A smart grid system that can predict the electricity demand for a certain cluster of houses with a lightweight PPDA system has been proposed [41]. This approach can fulfill both privacy-preserving requirements and low communication costs. Xu et al. [42] firstly focus on resisting data link attacks and proposed a data aggregation and classification approach for vehicular sensing systems. Guan et al. [43] proposed work on certifying local IoT devices and fog nodes by multiple authorities for fog-enhanced IoT. In our early work [44], we proposed a Double Trapdoor Chameleon Hash (DTCH) based online/offline signature and verification approach to reduce the computational costs of verifying data integrity.

However, the aforementioned works did not consider the high computational cost of signature and verification operations for data integrity. Therefore, the proposed VPFL includes a novel lightweight PPDA method that can provide a lightweight data integrity verification service with user data privacy protection mechanism for edge computing-enabled federated learning IoT systems.

3. Preliminary

In this section, we briefly introduce several definitions and notations used in our proposed VPFL scheme, including bilinear pairs, Paillier homomorphic cryptosystem, and online/offline signatures.

3.1. Bilinear pairs

With prime order p, G and G_T present two multiplicative cyclic groups, where g represents a generator of group G. According to Boneh et al. [45], a nodegenerated and computable bilinear map $e: \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ should satisfies the following three properties:

1. Bilinear: $e(u^a, v^b) = e(u,v)^{ab}$, where $u, v \in G, a, b \in Z_p^*$.
2. Nondegenerate: $e(g,g) \neq 1_{G_T}$, where $g \in G$.
3. Computable: $e(u, v)$ can be efficiently computed.

To evaluate the security of VPFL, we give the following definition and theorem.

Definition 1. q-SDH: Given two elements $(m, x) \in Z_p^*$, q-Strong Diffie-Hellman can be defined as calculating a pair (m, Σ_x). The q-SDH is a (q, t, ε)-hard problem when Eq.1 holds for any t-time adversary $\mathcal{A}$.

$$Pr\left[\mathcal{A}\left(g, g^x, g^{(x^2)}, \ldots, g^{(x^q)}\right) = (m, \Sigma_x)\right] < \varepsilon \tag{1}$$

3.2. Paillier homomorphic cryptosystem

We utilize the advantage of Paillier homomorphic cryptosystem [46] which is able to achieve additive homomorphism property to guarantee the model confidentiality during the model aggregation on the central server. Paillier homomorphic cryptosystem can be divided into the following steps.

1. *Key Generation*: Initially, selecting two primes (p, q) to compute the Carmichael function and RSA modulus as $\lambda = (p-1)(q-1)$ and $n = pq$, respectively. Further calculating the private element $\mu = \left(R(g^\lambda \bmod n^2)\right)^{-1}$, where $R(u) = \frac{u-1}{n}$.
2. *Encryption*: Given a plaintext $m_i \in \mathbb{Z}_n$, the encryption algorithm outputs the ciphertext as $c_i = g^{m_i} \cdot r^n \bmod n^2$, where $gcd(r, n) = 1$.
3. *Aggregation*: On receiving multiple ciphertexts c_i, the aggregation algorithm calculates the aggregated ciphertext as $c = \prod_{i=1}^{n} c_i \bmod n^2$.
4. *Decryption*: On receiving c, the decryption algorithm computes the aggregated plaintext as $m = \Sigma_{i+1}^{n} m_i = R(c^\lambda \bmod n^2)\mu \bmod n$.

3.3. Online/offline signatures

For achieving online and offline properties, we utilize a useful Double Trapdoor Chameleon Hash (DTCH) function [47] to divide a complete signature mechanism into two phases. Firstly, DTCH selects two elements $y, z \in \mathbb{Z}_{p_1}^*$ and randomly generates $(\mathbf{a}, \mathbf{b}, \mathbf{c})$ from chameleon hashes. Then, it computes $H_{ch}(\mathbf{a}, \mathbf{b}, \mathbf{c}) = g_1^{\mathbf{a}} \cdot g_2^{\mathbf{b}} \cdot g_3^{\mathbf{c}}$, where $g_2 = g_1^y$, $g_3 = g_1^z$. DTCH function satisfies three properties of computable, collision resistance, and trapdoor collision, which have been proved at length in Ref. [47].

In this paper, we applied the "hash-sign-switch" approach with the aforementioned DTCH function to construct the online/offline signature method, which can be represented as the following algorithms.

1. *System Setup*: It takes a random parameter 1^λ as input, and outputs the signature and verification key pair as (Sig_{sk}, Ver_{pk}).
2. *Offline Signature*: It takes (Sig_{sk}, Ver_{pk}) as inputs, and outputs Σ_{off} and SI, which represents the offline signature report and state information.
3. *Offline Verification*: It takes (Ver_{pk}), Σ_{off} as inputs, and outputs the offline verification results (*accept* or *reject*).
4. *Online Signature*: It takes (Sig_{sk}, SI, m) as inputs, and outputs the online signature report Σ_{on}.
5. *Online Verification*: It takes (Ver_{pk}, m, Σ_{on}, Σ_{off}) as inputs, and outputs the online verification results (*accept* or *reject*). In this way, the online/offline signature of a certain message m is defined as $\Sigma = (\Sigma_{off}, \Sigma_{on})$.

Note that, the bilinear pairs can provide the mathematical basics and generate the key materials. Besides, the Paillier homomorphic cryptosystem is used to guarantee the data confidentiality of the local training gradients, while the online/offline signature is a key technology to achieve data integrity during the parameter transmission phases. In the next section, we will show the details about how to apply the Paillier homomorphic cryptosystem and online/offline signature to the federated learning algorithm.

4. Our proposed VPFL scheme

4.1. Overview of VPFL scheme

As shown in Fig. 2, our proposed VPFL scheme consists of four

entities: Mobile Device (*MD*), Edge Server (*ES*), Central Server (*CS*), and Key Materials Generator (*KMG*).

1. *MD*: a set of participants that join in the federated learning protocol by training the local model on the sensed data. During the training procedure, all the participants transmit the local ciphertexts to the edge server and verify the correctness of the aggregated ciphertext.
2. *ES*: a set of relay units that execute the outsourced online/offline signature and verification processes. In this way, the communication overheads of *MD* can be reduced since the local ciphertexts are sent to the nearby *ES* instead of the long-term distance central server.
3. *CS*: a central entity that verifies and aggregates all the received local ciphertext. It also generates the aggregation signature and sends it to the *MD* along with the aggregated ciphertext.
4. *KMG*: a fully trusted entity that bootstraps the VPFL system and generate the key materials, including the Paillier public and private key pair, hash functions, random elements et al. We assume that the key materials can be secretly transmitted among different VPFL entities through a secure communication channel.

In Fig. 2, the overall procedure of the VPFL scheme is divided into the following five phases: *initialization phase*, *registration phase*, *local training phase*, *aggregation and decryption phase*, and *global update phase*. In the next subsections, we will show the details of the above-mentioned different phases.

4.2. Initialization phase

In the proposed VPFL, a trusted entity (*KMG*) exists that is responsible for bootstrapping the federated learning and transferring the cryptographic information to MD and CS. Note that, *KMG* only joins the system initialization and will not involve in the further processes.

For key materials generation (*KMG* execution):

1. Draw a random number p_1 and a bilinear map $e: \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$, where $|p_1| = k_1$ and (k, k_1) are two security parameters.
2. Generate Paillier cryptosystem key pair $(pk, sk) = (n, g)$, (λ, μ), three secure cryptographic one-way hash functions $H_0: \{0,1\}^* \to \mathbb{G}$, $H_1: \{0,1\}^* \to \mathbb{Z}_p^*$, $H_2: \mathbb{G} \to \mathbb{Z}_p^*$, and one Chameleon hash function $H_{ch}: \mathbb{Z}_{p_1}^* \to \mathbb{G}$.
3. Compute $e(g_1, g_1)^{\tilde{\alpha}}$ and $\tilde{Y} = g_1{}^{\tilde{x}}$, where $\tilde{\alpha}, \tilde{x} \in \mathbb{Z}_{p_1}^*$, $Q \in \mathbb{G}$ are three randomly generated elements.
4. Release the initialized key materials to *MD* as

$$km = \left\{ \begin{matrix} \mathbb{G}, \mathbb{G}_T, n, p_1, g, Q, \tilde{Y}, \\ e(g_1, g_1)^{\tilde{\alpha}}, H_0, H_1, H_2, H_{ch} \end{matrix} \right\} \tag{2}$$

5. Assign the master key $mk = (\lambda, \mu, p, q, \tilde{\alpha}, \tilde{x})$ to *CS* through a secured transmission channel.

4.3. Registration phase

When a new user i joins the VPFL system, i needs to register and authenticate its identity first. Once the authentication succeeds, it is required to generate the offline signature and transmit it to *CS*.

For user registration (*MD* execution):

1. Draw $x_i \in Z_p^*$ as the signature key Sig_k and $y_i = g^{x_i} \in G$ as the verification key Ver_k.
2. Compute $e_i = H_1(k_i \| ID_i \| SI_i)$, where $k_i \in Z_p^*$ is a blind factor and ID_i represents participant i's identity.
3. Compute $\alpha_i = g_1^{e_i}$ and $\beta_i = e_i - x_i H_2(\alpha_i)$.
4. Send the registration material $\{y_i, \alpha_i, \beta_i\}$ to *CS*.

For identity authentication (*CS* execution):

1. Check the correctness of equation $\alpha_i = g_1^{\beta_i} y_i^{H_2(\alpha_i)}$.
2. Compute $ak_i = (g_1{}^{\tilde{\alpha}} \cdot y^{t_i}, Q^{t_i}, g_1{}^{t_i})$ as the authenticated key of participant i, where $t_i \in \mathbb{Z}_{p_1}^*$.
3. Broadcast the registration material $\{y_i, \alpha_i, \beta_i\}$.

For offline signature generation (*MD* execution):

1. Draw $(y, z, s_i, u_i) \in \mathbb{Z}_{p_1}^*$.
2. Compute $g_2 = g_1^y$, $g_3 = g_1^z$ and keep state information $SI = (e_i, s_i, u_i)$ in local.
3. Compute the *BLS* signature [48] based on DTCH function as

$$\Sigma_i^{BLS} = (H_0(H_{ch_i}(e_i, s_i, u_i)))^{x_i} \tag{3}$$

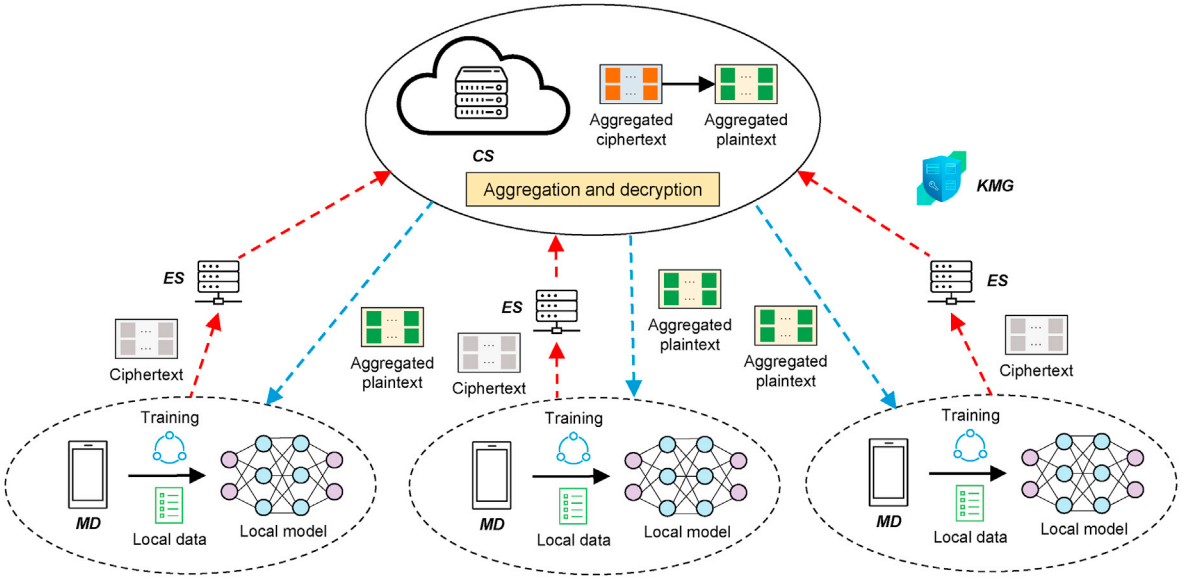


Fig. 2. The architecture and system model of VPFL scheme.

4. Compute the offline signature as

$$\Sigma_i^{off} = \left(\Sigma_i^{BLS} \| H_{ch_i} \| ID_i \| SI\right) \tag{4}$$

which is sent to *ES*.

5. Send the online verification key $Ver_{on} = (g_1, g_2, g_3)$ to *CS*.

4.4. Local training phase

In this phase, *ES* will batch verify all the received offline signatures Σ_i^{off}. Then, *MD* executes the local training processes based on the DSSGD method [49] and encrypt the local model gradients to generate the ciphertexts. Finally, the local report will be sent to *CS* along with the online signature Σ_i^{on}.

For offline signature verification (*ES* execution):

1. Check the correctness of equation $e(g_1, \Sigma_i^{BLS}) = e(y_i, H_0(H_{ch_i}))$ by using Ver_k..
2. Batch verify the offline signature by determining whether the following equation is true

$$\prod_{i=1}^{n} e(y_i, H_0(H_{ch_i})) = e\left(g_1, \prod_{i=1}^{n} \Sigma_i^{BLS}\right) \tag{5}$$

3. If the above equation holds, return *accept*, otherwise return *reject*.

For local ciphertext generation (*MD* execution):

1. Compute gradient vector in a certain communication round of federated learning as

$$g_t^{(i)} = \nabla_{w_t} \mathcal{L}(w_t, d_i); d_i \in \mathcal{D} \tag{6}$$

2. Split the weight w_t and gradient g_t as $w_t = (w_t^1, \ldots, w_t^n)$ and $g_t = (g_t^1, \ldots, g_t^n)$.
3. Compute local parameters: $w_{t+1}^{(i)} = w_t^{(i)} - \eta \cdot g_t^{(i)}$.
4. Generate local model updates: $L_{t+1}^{(i)} = w_{t+1}^{(i)} - w_t^{(i)}$.
5. Compute the local ciphertext as

$$c_{t+1}^{(i)} = g^{\left(L_{t+1}^{(i)}\right)} \cdot \left(v_t^{(i)}\right)^n \bmod n^2; v_t^{(i)} \in \mathbb{Z}_{n^2}^* \tag{7}$$

For online signature generation (*MD* execution):

1. Compute $u_i' = \left(\left(e_i - c_{t+1}^{(i)}\right) + (s_i - s_i')y + u_i z\right) z^{-1}$, where $s_i' \in \mathbb{Z}_{p_1}^*$.
2. Generate online signature as $\Sigma_i^{on} = (s_i', u_i')$.
3. Send the local ciphertext $c_{t+1}^{(i)}$ and online signature Σ_i^{on} to *CS*.

4.5. Aggregation and decryption phase

In the aggregation and decryption phase, *CS* first verify the validity of all the received online signatures Σ_i^{on}. Then, it aggregates all the local ciphertexts and executes the decryption algorithm to extract the aggregated model updates in plaintext form.

For online signature verification (*CS* execution):

1. Verify the correctness of Σ_i^{on} by determining whether the following equation is true

$$H_{ch}(r_i, s_i, u_i) = H_{ch}(c_i, s_i', u_i') \tag{8}$$

2. If the above equation holds, return *accept*, otherwise return *reject*.

For aggregation and decryption (*CS* execution):

1. Compute the aggregated ciphertext as

$$c_{t+1} = \prod_{i=1}^{n_{t+1}} c_{t+1}^{(i)} \bmod n^2 \tag{9}$$

2. Transfer c_{t+1} to $g^{\sum_{i=1}^{n_{t+1}} \left(L_{t+1}^{(i)}\right)} \cdot \prod_{i=1}^{n_{t+1}} \left(v_t^{(i)}\right)^n \bmod n^2$, which is a standard Paillier encryption form.
3. Decrypt the aggregated ciphertext c_{t+1} as

$$L_{t+1} = \frac{R(c_{t+1}{}^{\lambda} \bmod n^2)}{R(g^{\lambda} \bmod n^2)} \bmod n \tag{10}$$

4.6. Global update phase

In this phase, *CS* further average the aggregated plaintext L_{t+1} to generate a new global model as

$$G_{t+1}^{(global)} = G_t^{(global)} + \frac{1}{n_{t+1}} L_{t+1} \tag{11}$$

When a next communication round $t+1$ starts, each participant i updates it local model parameters $w_{(t+1)}$ based on $w_{(t+1)} \leftarrow w_t^{global} - \eta' \cdot G_{t+1}^{(global)}$. The whole VPFL procedure will not end until the global model w^{global} tends to convergence.

5. Security analysis

5.1. Verifiability of signature

Theorem 1. *In the VPFL scheme, if all the entities can execute the algorithm honestly, the online signature verification and offline signature verification phases could also be computed correctly.*

Proof. For offline signature verification phase, ES checks all the received offline signature $e(g_1, \Sigma_i^{BLS})$ by using the batch verification method. We prove Eq. (5) through the following derivation process:

$$\begin{aligned} \prod_{i=1}^{n} e(y_i, H_0(H_{ch_i})) &= \prod_{i=1}^{n} e(g_1^{x_i}, H_0(H_{ch_i})) \\ &= \prod_{i=1}^{n} e\left(g_1, \Sigma_i^{BLS}\right) = e\left(g_1, \prod_{i=1}^{n} \Sigma_i^{BLS}\right) \end{aligned} \tag{12}$$

For online signature verification phase, CS checks all the received online signature $\Sigma_i^{on} = (s_i', u_i')$ by determining whether $H_{ch}(r_i, s_i, u_i) = H_{ch}(c_i, s_i', u_i')$ holds. We prove Eq. (8) through the following derivation process:

$$\begin{aligned} H_{ch}(c_i, s_i', u_i') &= g_1^{c_i'} \cdot g_2^{s_i'} \cdot g_3^{u_i'} \\ &= g_1^{c_i} \cdot (g_1^y)^{s_i'} \cdot g_1^{z((r_i - c_i) + (s_i - s_i')y + u_i z) z^{-1}} \\ &= g_1^{c_i} \cdot (g_1^y)^{s_i'} \cdot g_1^{r_i} \cdot g_1^{-c_i} \cdot g_1^{y \cdot s_i} \cdot (g_1^y)^{-s_i'} \cdot g_1^{z \cdot u_i} \\ &= (g_1)^{r_i} \cdot (g_1^y)^{s_i} \cdot \left(g_1^z\right)^{u_i} = g_1^{r_i} \cdot g_2^{s_i} \cdot g_3^{u_i} \\ &= H_{ch}(r_i, s_i, u_i) \end{aligned} \tag{13}$$

5.2. Participant authentication

Theorem 2. *In the VPFL scheme, each participant's identity ID_i can be efficiently authenticated by CS, while the private identity information will not be leaked to any entity.*

Proof. In the registration phase of the VPFL scheme, we utilize a

simple extended Schnorr's signature method based on discrete logarithm problem [50], to authenticate the identity ID_i of each participant. Specifically, CS authenticates the identity of i based on the received registration material $\{y_i, \alpha_i, \beta_i\}$ as

$$g_1^{\beta_i} Y_i^{H_2(\alpha_i)} = g_1^{(r_i - x_i H_2(\alpha_i))} \cdot g_1^{x_i H_2(\alpha_i)} = g_1^{r_i} = \alpha_i \tag{14}$$

The security of authentications rely on the unforgeable of registration material $\{\alpha_i, \beta_i\}$. In VPFL, we utilize a blind factor k_i and the hash function H_1 to hide the participant i's identity information ID_i and hash function value e_i. Without ID_i and e_i, the adversary has no way to obtain the signature key $x_i \in Z_p^*$, thus guarantee the security of participants' authentication in the proposed VPFL scheme.

5.3. Confidentiality and privacy-preserving

Theorem 3. *In the VPFL scheme, if the Paillier cryptosystem is proved to be secure, the ciphertext $c_{t+1}^{(i)}$ can prevent the gradients information $L_{t+1}^{(i)}$ from being leaked to the internal and external adversaries.*

Proof. In the local training phase of the VPFL scheme, a Paillier encryption method is used to transfer all the local gradients $L_{t+1}^{(i)}$ to the ciphertext $c_{t+1}^{(i)}$. Whenever $c_{t+1}^{(i)}$ is transmitted to CS, it firstly aggregates all the received ciphertexts as c_{t+1}. Then, CS can derive the aggregated plaintext L_{t+1} from c_{t+1}, since $c_{t+1} = g^{\sum_{i=1}^{n_{t+1}} (L_{t+1}^{(i)})} \cdot \prod_{i=1}^{n_{t+1}} (v_t^{(i)})^n \bmod n^2$ is also a valid ciphertext form of Paillier cryptosystem. In this situation, the decryption key (λ, μ) consists of two random variables which are specific in each communication round and invisible to any internal and external adversary. Moreover, each individual's gradient information $L_{t+1}^{(i)}$ can be protected since CS can only obtain the aggregated plaintext L_{t+1}. In summary, our proposed VPFL scheme can offer confidentiality and privacy-preserving for the participants' local gradients.

5.4. Integrity verification

Definition 2. EU-CMA: We say an online/offline signature method is existential unforgeability under chosen message attacks [31,51] if an adversary $\mathcal{A}$ can successfully forge a signature $\hat{\Sigma}$ through multiply queries to the signature oracles $(\Sigma^{on}, \Sigma^{off})$ with a probability of no less than

$$Pr_{\mathcal{A}} = Pr\left[Ver_{on}(pk, \hat{m}, \hat{\Sigma}) = 1 : (pk, sk); (\hat{m}, \hat{\Sigma}) \leftarrow \mathcal{A}^{(\Sigma^{on}, \Sigma^{off})} \right] \tag{15}$$

Theorem 4. *In the VPFL scheme, the embedded online/offline signature scheme is $(\tau, \varepsilon, q_1, q_2)$ secure against EU-CMA if the there exists a challenger $\mathcal{C}$ that can compute the q-SDH problem with a non-negligible probability $\varepsilon' \geq \frac{\varepsilon}{3} - \frac{q_2}{p}$ in polynomial time.*

Proof. According to the Definition 2, EU-CMA represents an adversary $\mathcal{A}$ tries to forge a valid signature $\hat{\Sigma}$ of the target message $\hat{m}$ by multiply querying the online/offline signature oracles $(\Sigma^{on}, \Sigma^{off})$. We turn the above problem into an adversary-challenger game based on the Definition 1. That is, the adversary $\mathcal{A}$ obtains a set of q-SDH instances $(g, g^*, g^{(*2)}, \ldots, g^{(*q)})$ from the challenger $\mathcal{C}$, which can solve the q-SDH problem by constructing a new valid online/offline signature $\hat{\Sigma} = (\hat{\Sigma}^{on}, \hat{\Sigma}^{off})$. The detailed proof process of Theorem 5 can be found in our previous work in [31]. Here, we give a brief description of the proof process as follows. Assuming the adversary $\mathcal{A}$ already queried q_1-*th* offline and q_2-*th* online signatures of the target message $\hat{m}$ (represent as $(\hat{\Sigma}_{q_1}^{off}, \hat{\Sigma}_{q_2}^{on})$) and $g^{\hat{m}} g_2^{\hat{s}} g_3^{\hat{u}} = g^{m_i} g_2^{s_i} g_3^{u_i}$ for some $i \in \{1, \ldots, q_2\}$, or $\hat{s} \neq s_i$ happened. Then, the challenger $\mathcal{C}$ forges one of the double trapdoors y or z, and further generates a new Chameleon hash function value $\hat{H}_{ch}$. So far, we know that the probability of $g^{\hat{m}} g_2^{\hat{s}} g_3^{\hat{u}} = g^{m_i} g_2^{s_i} g_3^{u_i}$ occurred is no less than $\varepsilon/3$ and $\hat{s} = s_i$ happened is $1/p$. Therefore, for online signatures that have been queried q_2 times, the probability of $\hat{s} = s_i$ occurred is no more than q_2/p. We say that if the adversary $\mathcal{A}$ can successfully forge a new signature $\hat{\Sigma} = (\hat{m}, \hat{\Sigma}^{on}(\hat{s}, \hat{u}), \hat{\Sigma}^{off}(\tau, \hat{e}, \hat{s}, \hat{u}))$ that meets the condition of $g^{\hat{m}} g_2^{\hat{s}} g_3^{\hat{u}} = g^{m_i} g_2^{s_i} g_3^{u_i}$, then the challenger $\mathcal{C}$ can also compute $\tau = y = ((\hat{m} - m_i) + (\hat{u} - u_i)z)(s_i - \hat{s})^{-1}$ or $\tau = z = ((\hat{m} - m_i) + (\hat{s} - s_i)z)(u_i - \hat{u})^{-1}$ correctly. That means the challenger $\mathcal{C}$ can solve the q-SDH problem with a minimum probability of $\varepsilon/3 - q_2/p$. In this way, the Theorem 5 is proved since the above conclusion of challenger $\mathcal{C}$ is contrary to the Definition 1.

6. Performance analysis

This section evaluates the performance of the proposed VPFL scheme from the computational complexity, communication overhead, and performance on federated learning, respectively. For the computational complexity evaluations, we compare our proposed VPFL scheme with the other three recently related works [37,39,40]. For the communication overhead evaluations, we choose two traditional data aggregation methods [25,28] to comparably demonstrate the effectiveness of the VPFL scheme.

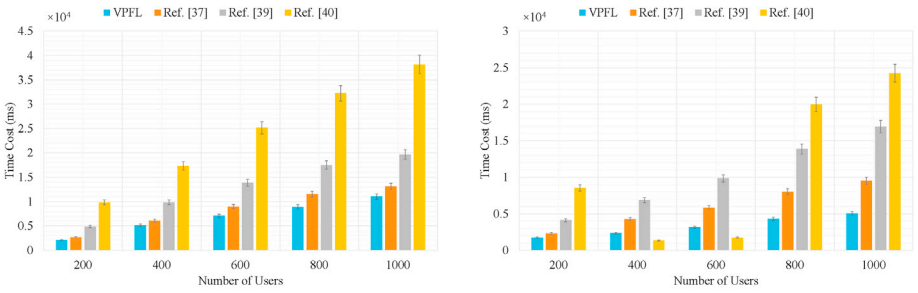
6.1. Dataset and experimental setup

We use a benchmark dataset MNIST for our experiment, which is consist of 70000 instances of handwriting digits, which consists of 60000 training images and 10000 testing instances. For the experimental settings, we apply the pairing-based cryptography (PBC) library to measure the costs of the Paillier cryptosystem. The RSA modulus n and security parameter p_1 are set to 1024 bits and 160 bits, respectively. For comparing the classification performance, we apply the Convolutional Neural Network (CNN) as the image classifier for the MNIST dataset. The CNN structure includes two convolutional layers and two dense layers with ReLU as the activation function, additionally, kernel size is set to 4 × 4. The result will go through a Softmax layer of the total 10 classes of MNIST. All the experiments were conducted on Nvidia Quadro P4000 GPU and 32 GB RAM platform with Linux RHEL7.5 system.

6.2. Computational complexity of cryptosystem

There are several calculations required for the proposed VPFL, which include two exponentiation operations in Z_{n^2} to generate ciphertext c_i, and three multiplication operations in G to calculate the online signature Σ_i^{on}. The online signature will be verified by *ES*, where all the collected ciphertexts will also be aggregated by *ES* by using the aforementioned exponentiation and multiplication operations in G and Z_{n^2}. Furthermore, the aggregation signature Σ_{Agg} will be generated by one one exponentiation operation in G on ES. The generated Σ_{Agg} will be sent to *CS* and verified by *CS*, where it decrypts the aggregated ciphertext c to obtain the sum plaintext of the previous operations.

The complexity analysis above proves that cryptographic operations and signature operations require less time-consuming in terms of *MD*. Fig. 3(a) compares the overall computational cost among the four approaches. The results show that the computational cost of the proposed VPFL is significantly lower than other three approaches [37,39], and [40]. In which the proposed VPFL shift the time-consuming complex operations into the offline phase. Fig. 3(b) compares the time cost tendency of signature and verification parts. It is clear that the time cost of verification and signature operations in the proposed VPFL is almost 50% lower than the other three methods.



(a) Overall computation cost of cryptosystem (b) Signature and verification costs

Fig. 3. Computational complexity of cryptosystem compared with [37,39], and [40].

6.3. Communication overhead of cryptosystem

For the evaluation of communication overhead, we set the RSA modulus n and security parameter p_1 to 1024 bits and 160 bits respectively. Then, we compare the communication overheads of two transmission stages by using the numerical analysis method. Specifically, there are two phases in the proposed VPFL which are *MD* to *ES* communication (*MD*-to-*ES*) and *ES* to *CS* communication (*ES*-to-*CS*). *MD* first generates a data report and sent to a edge server *ES*, which can be represented as $P_i = ID_i||c_i||SI_i||\Sigma_i^{on}$ in the *MD*-to-*ES* phase. The proposed VPFL scheme can reduce the communication cost significantly since traditional cloud computing requires each participant to upload their reports individually. Fig. 4 compares the communication cost on both two phases of the proposed VPFL and the other two approaches [25,28]. The results show that the proposed VPFL work more efficiently than [25, 28] when the number of users increases. Note that, the results in Fig. 4(b) have no changes when the number of users increases, this is because there is no correlation between communication costs and the number of users in the *ES*-to-*CS* phase. Besides, since the received offline signatures in the local training procedure are executed by using the batch verification method, thus significantly reduces the number of communication iterations between the edge server and each mobile device.

6.4. Performance on federated learning

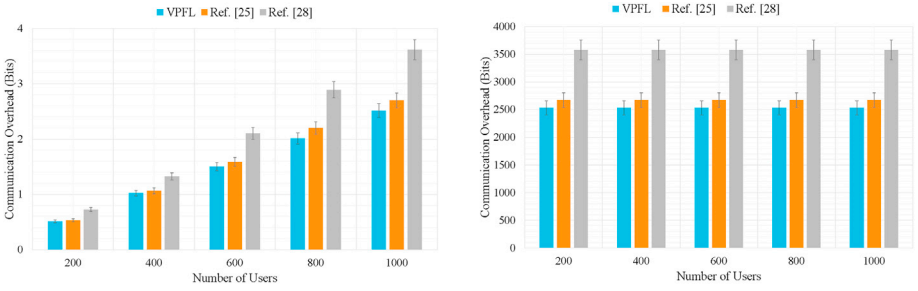
We evaluate the performance of federated learning from two perspectives which are computational costs of cryptosystem and performance of VPFL on MNIST dataset. Firstly, we evaluate the time cost of the Paillier homomorphic encryption method with the change of lengths of the gradient vectors. Fig. 5(a) illustrates that even though the times cost increases when the number of gradients increases, however, it is still a low time cost level for encryption and decryption operations. Moreover, we also train a CNN model on the MNIST dataset to evaluate the computational overheads of VPFL. Fig. 5(b) compares the trend of time cost of the training procedure when the number of participants increases. The results show that the impact on encrypted training is slight, where the result remains constant. In the meantime, the time cost of server-side decryption and aggregation operations is kept low during the training stage.

To further evaluate the performance of federated learning, we perform an image classification task on the proposed VPFL with the MNIST dataset. The classification accuracy will be measured over different communication rounds. We construct the experiment in Pytorch environment and set the number of participants $i = 20$. Fig. 6 shows that the accuracy and loss of the proposed VPFL are very close to the results of original federated learning. Therefore, the cryptographic operation in VPFL does not sacrifice the performance of federated learning.

According to the above performance analysis, VPFL scheme has a low burden on both communication overhead and computation cost, which can be used in lots of smart IoT systems. For example, in a smart grid system, the smart customers' consumption information and electricity bills cannot be directly uploaded to the central server due to privacy concerns. By utilizing VPFL scheme, the private electricity information can be encrypted and aggregated to realize privacy protection. Besides, the signature and verification methods can also provide the integrity of electricity information during the transmission procedure.

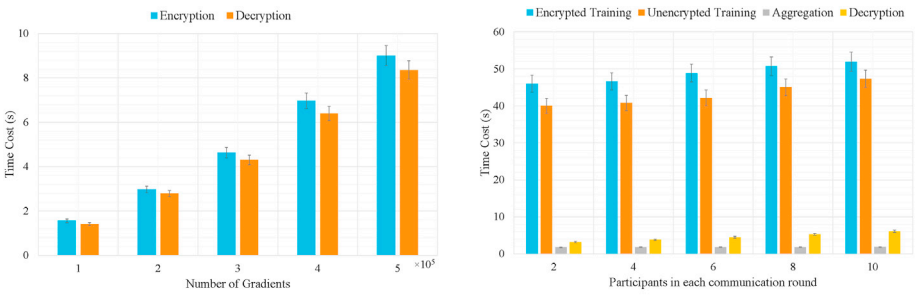
7. Summary and future work

In this paper, we proposed the VPFL scheme, a verifiable privacy-preserving federated learning scheme for edge computing systems to prevent local gradients from leaking over the transmission stage. The VPFL allows each participant to encrypt the local gradients efficiently through the Paillier cryptosystem and the central server can only observe the ciphertexts of local updates. Meanwhile, we embedded the DSSGD method into the VPFL scheme to reduce the computation cost of the cryptosystem during the local training phase. Besides, we further presented an online/offline signature method to achieve lightweight signature verification for local gradients' integrity. At last, the comprehensive security analysis and experimental evaluations demonstrate that



(a) Communication overhead between *MD* and *ES* (b) Communication overhead between *ES* and *CS*

Fig. 4. *Communication overhead of cryptosystem compared with* [25,28].



(a) Computational costs of encryption and decryption (b) Computational costs of cryptosystem on MNIST dataset

Fig. 5. Computation cost of VPFL on MNIST dataset.

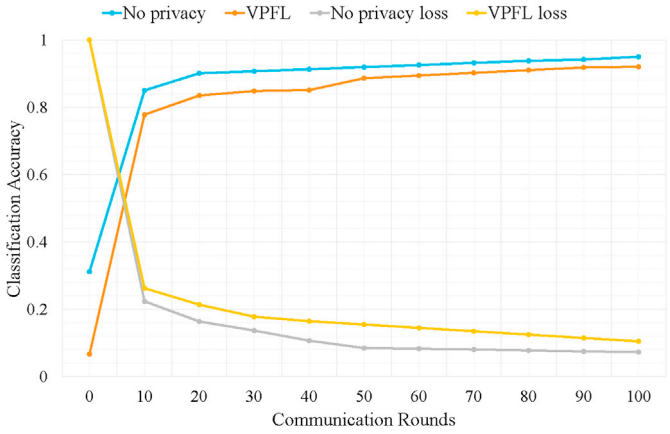


Fig. 6. Accuracy comparison over communication rounds.

the proposed VPFL scheme can realize privacy-preserving, lightweight integrity verification, and high performance on accuracy and efficiency. Since we combine the Paillier cryptosystem and signature method to achieve confidentiality and integrity simultaneously, the tradeoffs between the privacy-preserving and computation costs are nonnegligible. Fortunately, thanks to the mobile edge computing framework, the most complex operations of the cryptosystem are outsourced to the edge server, thus reducing the resource consumption of mobile devices. In future work, we plan to study more complex neural networks and datasets to explore the personalized PPFL mechanisms.

Declaration of competing interest

We declare that we have no conflicts of interest in this work (DCAN 435). We also declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 62206238), the Natural Science Foundation of Jiangsu Province (Grant No. BK20220562), and the Natural Science Research Project of Universities in Jiangsu Province (No. 22KJB520010).

References

[1] K. Zhang, Y. Mao, S. Leng, Y. He, Y. Zhang, Mobile-edge computing for vehicular networks: a promising network paradigm with predictive off-loading, IEEE Veh. Technol. Mag. 12 (2) (2017) 36–44.
[2] L. Catarinucci, D. De Donno, L. Mainetti, L. Palano, L. Patrono, M.L. Stefanizzi, L. Tarricone, An iot-aware architecture for smart healthcare systems, IEEE Internet Things J. 2 (6) (2015) 515–526.
[3] D. He, N. Kumar, S. Zeadally, A. Vinel, L.T. Yang, Efficient and privacy-preserving data aggregation scheme for smart grid against internal adversaries, IEEE Trans. Smart Grid 8 (5) (2017) 2411–2419.
[4] W. Ejaz, M. Naeem, A. Shahid, A. Anpalagan, M. Jo, Efficient energy management for the internet of things in smart cities, IEEE Commun. Mag. 55 (1) (2017) 84–91.
[5] I. Psychoula, D. Singh, L. Chen, F. Chen, A. Holzinger, H. Ning, Users' privacy concerns in iot based applications, in: 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), IEEE, 2018, pp. 1887–1894.
[6] W. Shi, J. Cao, Q. Zhang, Y. Li, L. Xu, Edge computing: vision and challenges, IEEE Internet Things J. 3 (5) (2016) 637–646.
[7] X. Sun, N. Ansari, Edgeiot: mobile edge computing for the internet of things, IEEE Commun. Mag. 54 (12) (2016) 22–29.
[8] Y. Li, H. Ma, L. Wang, S. Mao, G. Wang, Optimized content caching and user association for edge computing in densely deployed heterogeneous networks, IEEE Trans. Mobile Comput. 21 (6) (2022) 2130–2142.
[9] S. Xia, Z. Yao, Y. Li, S. Mao, Online distributed offloading and computing resource management with energy harvesting for heterogeneous mec-enabled iot, IEEE Trans. Wireless Commun. 20 (10) (2021) 6743–6757.
[10] B. McMahan, E. Moore, D. Ramage, S. Hampson, B.A. y Arcas, Communication-efficient learning of deep networks from decentralized data, in: Artificial Intelligence and Statistics, PMLR, 2017, pp. 1273–1282.
[11] Q. Yang, Y. Liu, T. Chen, Y. Tong, Federated machine learning: concept and applications, ACM Trans. Intell.Syst.Technol.(TIST) 10 (2) (2019) 1–19.
[12] R. Shokri, V. Shmatikov, Privacy-preserving deep learning, in: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, 2015, pp. 1310–1321.
[13] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H.B. McMahan, S. Patel, D. Ramage, A. Segal, K. Seth, Practical secure aggregation for privacy-preserving machine learning, in: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, 2017, pp. 1175–1191.
[14] Y. Aono, T. Hayashi, L. Wang, S. Moriai, et al., Privacy-preserving deep learning via additively homomorphic encryption, IEEE Trans. Inf. Forensics Secur. 13 (5) (2017) 1333–1345.
[15] M. Fredrikson, S. Jha, T. Ristenpart, Model inversion attacks that exploit confidence information and basic countermeasures, in: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, 2015, pp. 1322–1333.
[16] Z. Wang, M. Song, Z. Zhang, Y. Song, Q. Wang, H. Qi, Beyond inferring class representatives: user-level privacy leakage from federated learning, in: IEEE INFOCOM 2019-IEEE Conference on Computer Communications, IEEE, 2019, pp. 2512–2520.
[17] X. Yin, Y. Zhu, J. Hu, A comprehensive survey of privacy-preserving federated learning: a taxonomy, review, and future directions, ACM Comput. Surv. 54 (6) (2021) 1–36.
[18] X. Gong, A. Sharma, S. Karanam, Z. Wu, T. Chen, D. Doermann, A. Innanje, Ensemble attention distillation for privacy-preserving federated learning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 15076–15086.
[19] Z. Zhang, C. Guan, H. Chen, X. Yang, W. Gong, A. Yang, Adaptive privacy preserving federated learning for fault diagnosis in internet of ships, IEEE Internet Things J. 9 (9) (2022) 6844–6854.
[20] P. Sun, H. Che, Z. Wang, Y. Wang, T. Wang, L. Wu, H. Shao, Pain-fl: personalized privacy-preserving incentive for federated learning, IEEE J. Sel. Area. Commun. 39 (12) (2021) 3805–3820.
[21] C. Jiang, C. Xu, Y. Zhang, Pflm: privacy-preserving federated learning with membership proof, Inf. Sci. 576 (2021) 288–311.
[22] J. Zhang, B. Chen, S. Yu, H. Deng, Pefl, A privacy-enhanced federated learning scheme for big data analytics, in: 2019 IEEE Global Communications Conference (GLOBECOM), IEEE, 2019, pp. 1–6.
[23] J. He, L. Cai, P. Cheng, J. Pan, L. Shi, Distributed privacy-preserving data aggregation against dishonest nodes in network systems, IEEE Internet Things J. 6 (2) (2018) 1462–1470.

[24] F. Li, B. Luo, P. Liu, Secure information aggregation for smart grids using homomorphic encryption, in: 2010 First IEEE International Conference on Smart Grid Communications, IEEE, 2010, pp. 327–332.
[25] R. Lu, X. Liang, X. Li, X. Lin, X. Shen, Eppa: an efficient and privacy-preserving aggregation scheme for secure smart grid communications, IEEE Trans. Parallel Distr. Syst. 23 (9) (2012) 1621–1631.
[26] H. Li, X. Lin, H. Yang, X. Liang, R. Lu, X. Shen, Eppdr: an efficient privacy-preserving demand response scheme with adaptive key evolution in smart grid, IEEE Trans. Parallel Distr. Syst. 25 (8) (2013) 2053–2064.
[27] C.-I. Fan, S.-Y. Huang, Y.-L. Lai, Privacy-enhanced data aggregation scheme against internal attackers in smart grid, IEEE Trans. Ind. Inf. 10 (1) (2013) 666–675.
[28] J. Ni, K. Alharbi, X. Lin, X. Shen, Security-enhanced data aggregation against malicious gateways in smart grid, in: 2015 IEEE Global Communications Conference (GLOBECOM), IEEE, 2015, pp. 1–6.
[29] J. Ni, K. Zhang, X. Lin, X.S. Shen, Edat: efficient data aggregation without ttp for privacy-assured smart metering, in: 2016 IEEE International Conference on Communications (ICC), IEEE, 2016, pp. 1–6.
[30] Z. Guan, Y. Zhang, L. Zhu, L. Wu, S. Yu, Effect: an efficient flexible privacy-preserving data aggregation scheme with authentication in smart grid, Sci. China Inf. Sci. 62 (3) (2019) 32103.
[31] J. Zhang, Y. Zhao, J. Wu, B. Chen, Lvpda: a lightweight and verifiable privacy-preserving data aggregation scheme for edge-enabled iot, IEEE Internet Things J. 7 (5) (2020) 4016–4027.
[32] R. Xu, N. Baracaldo, Y. Zhou, A. Anwar, H. Ludwig, Hybridalpha: an efficient approach for privacy-preserving federated learning, in: Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security, 2019, pp. 13–23.
[33] Y. Chen, F. Luo, T. Li, T. Xiang, Z. Liu, J. Li, A training-integrity privacy-preserving federated learning scheme with trusted execution environment, Inf. Sci. 522 (2020) 69–79.
[34] K. Wei, J. Li, M. Ding, C. Ma, H. Su, B. Zhang, H. V. Poor, User-level privacy-preserving federated learning: analysis and performance optimization, IEEE Trans. Mobile Comput., https://doi.org/10.1109/TMC.2021.3056991.
[35] Y. Li, Y. Zhou, A. Jolfaei, D. Yu, G. Xu, X. Zheng, Privacy-preserving federated learning framework based on chained secure multiparty computing, IEEE Internet Things J. 8 (8) (2020) 6178–6186.
[36] M. Hao, H. Li, X. Luo, G. Xu, H. Yang, S. Liu, Efficient and privacy-enhanced federated learning for industrial artificial intelligence, IEEE Trans. Ind. Inf. 16 (10) (2019) 6532–6542.
[37] A. Fu, X. Zhang, N. Xiong, Y. Gao, H. Wang, J. Zhang, Vfl: A verifiable federated learning with privacy-preserving for big data in industrial iot, IEEE Trans. Ind. Inf. 18 (5) (2022) 3316–3326.
[38] X. Guo, Z. Liu, J. Li, J. Gao, B. Hou, C. Dong, T. Baker, V. eri fl, Communication-efficient and fast verifiable aggregation for federated learning, IEEE Trans. Inf. Forensics Secur. 16 (2020) 1736–1751.
[39] C. Zhang, S. Li, J. Xia, W. Wang, F. Yan, Y. Liu, Batchcrypt: efficient homomorphic encryption for cross-silo federated learning, in: 2020 \{USENIX\} Annual Technical Conference (\{USENIX\} \{ATC\} 20), 2020, pp. 493–506.
[40] X. Zhang, A. Fu, H. Wang, C. Zhou, Z. Chen, A privacy-preserving and verifiable federated learning scheme, in: ICC 2020-2020 IEEE International Conference on Communications (ICC), IEEE, 2020, pp. 1–6.
[41] A. Abdallah, X. Shen, Lightweight security and privacy preserving scheme for smart grid customer-side networks, IEEE Trans. Smart Grid 8 (3) (2015) 1064–1074.
[42] C. Xu, R. Lu, H. Wang, L. Zhu, C. Huang, Pavs: a new privacy-preserving data aggregation scheme for vehicle sensing systems, Sensors 17 (3) (2017) 500.
[43] Z. Guan, Y. Zhang, L. Wu, J. Wu, J. Li, Y. Ma, J. Hu, Appa: an anonymous and privacy preserving data aggregation scheme for fog-enhanced iot, J. Netw. Comput. Appl. 125 (2019) 82–92.
[44] J. Zhang, Y. Zhao, J. Wu, B. Chen, Lpda-ec: a lightweight privacy-preserving data aggregation scheme for edge computing, in: 2018 IEEE 15th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), IEEE, 2018, pp. 98–106.
[45] D. Boneh, M. Franklin, Identity-based encryption from the weil pairing, in: Annual International Cryptology Conference, Springer, 2001, pp. 213–229.
[46] P. Paillier, Public-key cryptosystems based on composite degree residuosity classes, in: International Conference on the Theory and Applications of Cryptographic Techniques, Springer, 1999, pp. 223–238.
[47] E. Bresson, D. Catalano, R. Gennaro, Improved on-line/off-line threshold signatures, in: International Workshop on Public Key Cryptography, Springer, 2007, pp. 217–232.
[48] D. Boneh, B. Lynn, H. Shacham, Short signatures from the weil pairing, J. Cryptol. 17 (4) (2004) 297–319.
[49] J. Dean, G.S. Corrado, R. Monga, K. Chen, M. Devin, Q.V. Le, M.Z. Mao, M. Ranzato, A. Senior, P. Tucker, et al., Large Scale Distributed Deep, Networks (2011).
[50] E.-J. Goh, S. Jarecki, J. Katz, N. Wang, Efficient signature schemes with tight reductions to the diffie-hellman problems, J. Cryptol. 20 (4) (2007) 493–514.
[51] Y. Zhang, Z. Chen, F. Guo, Online/offline verification of short signatures, in: International Conference on Information Security and Cryptology, Springer, 2010, pp. 350–358.