An Investigation of Pseudonymization Techniques in Decentralized Transactions

Sandi Rahmadika[1], Muhammad Firdaus[2], Yong-Hwan Lee[1], and Kyung-Hyune Rhee[2*]
[1]Wonkwang University, Jeonbuk, Iksan City 54538, Republic of Korea
ndiikaa@gmail.com, hwany1458@empas.com
[2]Pukyong National University, Busan 48513, Republic of Korea
mfirdaus@pukyong.ac.kr, khrhee@pknu.ac.kr

Received: September 3, 2021; Accepted: November 2, 2021; Published: November 30, 2021

Abstract

Decentralized learning (DL) enables several devices to assemble deep learning models while keeping their private training data on the device. Rather than uploading the training data and model to the server, cross-silo DL only sends the local gradients gradually to the aggregation server back and forth. Hence, DL can provide privacy training of machine learning. Nevertheless, cross-silo DL lacks the proper incentive mechanism for the clients. Thanks to the blockchain, smart contracts (SCs) can address the concerns by providing immutable data records which are self-executing and tamper-proof to failures. Yet, the records of blockchain transactions are publicly visible, which can leak valuable clients' information as analytical systems become more sophisticated. We leverage the Monero (XMR) protocols to be adjusted into cross-silo DL transactions over wireless networks to address the issues. Concurrently, we investigate the performance of constructed protocols embedded into blockchain smart contracts. This paper also reports and analyzes an empirical investigation of several privacy preservation techniques in decentralized transactions. Overall, the performance results satisfy the design goals. Our observations fill the current literature gap concerning an up-to-date systematic mapping study, not to mention extensive techniques in preserving privacy for cross-silo DL combined with blockchain.

Keywords: blockchain-based incentive, decentralized learning, pseudonymization protocols, smart contract

1 Introduction

The vigorous utilization of internet-based information systems that rely on a decentralized approach has been broadly researched by academia, developers, and industries. The foremost objective of the decentralized approach is to address the communication bottleneck issues and memory usage of the conventional centralized system [22]. The paradigm of a robust concentrated approach (relies on a single node) via wireless networks has been gradually shifting toward decentralized practices, such as financial services, healthcare records, any forms of digital rights, and intellectual property. Blockchain technology through Bitcoin cryptocurrency and decentralized learning are the most prominent practical adoptions of decentralized approaches. Concurrently, decentralized ledger and artificial intelligence (AI) are expeditiously converging to address many critical challenges. Blockchain technology arguably is an ingenious invention, the brainchild of a user or group known by Satoshi Nakamoto's pseudonym [11] that make

Journal of Internet Services and Information Security (JISIS), volume: 11, number: 4 (November), pp. 1-18
DOI:*10.22667/JISIS.2021.11.30.001*

*Corresponding author: Department of IT Convergence and Application Engineering, Pukyong National University, Yongso-ro 45, Nam-gu, Busan (48513), Republic of Korea. Telp: +82-(0)51-6296247, Fax: +82-(0)51-6264887

smart contracts are all the rage in the blockchain world these days. Many pundits claim SCs will convey an entirely new paradigm that forever changes how the parties write contracts and conduct business. The successfully conducted blockchain transactions are candidly available in the network and can be accessed through the user interface by blockchain entities. Thus, it has been practiced in various disciplines of science.

Linked to the Bitcoin and Ethereum blockchain, distributed learning also relies on the decentralized approach to collectively building the deep learning model from multiple devices. In contrast to conventional machine learning, where the clients process the training model centrally, FL allows the clients to build the artificial intelligence (AI) model by sending the updated gradient values to the aggregation server without revealing the dataset [7]. Accordingly, private data remain confidential (the DL preserves privacy for clients by design). The DL-based schemes, such as federated learning, lack the proper incentive mechanism to motivate clients to improve AI models. Several applications do not even provide a reward for clients. Blockchain with SC features can be a solution to tackle the incentive mechanism issues. Nevertheless, directly employing SCs threatens the clients' privacy since SCs are transparent and readily available in the blockchain user interface. The existing smart contracts-based solution, such as Ethereum smart contract and Hyperledger chain-code, only perform a simple computation that cannot satisfy the application of real-world AI. The developers can program the codes as a self-execute program without third-party involvement. AI with smart contract integration may render a more resilient and efficient path for a decentralized interactive system for the parties. In short, SCs in DL must be thoroughly investigated when these technologies are implemented in a system with profoundly confidential information, such as federated identity (federated electronic identity) [5] and digital forensics [32]. Precisely, complementary protocols need to be supplemented.

Privacy-awareness in the smart contract and decentralized learning is part of a flaw that must be considered in wireless network environments. Melis et al. [30] surprisingly claimed that the observer could infer the presence of exact data points of clients' datasets with particular assumptions. While SCs present client's transactions visible to the observer. The record of transactions can be accessed anytime, and the value of data is noticeable. Transparency is one of the concrete features of the SC blockchain. However, this feature is not desirable for various cases, especially in decentralized learning with highly confidential data such as medical records, biometric data, employee data, sexual orientation, philosophical beliefs, and so forth. For these reasons, the relationship between the data used in training and the owner needs to be obscured. This research explores the techniques of privacy preservation for cross-silo decentralized learning in untrusted wireless networks. We investigate the application of decentralized transactions such as decentralized learning by referring to the federated learning principles (developed by the Google AI team) [16] as a case study for an incentive scheme running on blockchain SCs. We also analyze the extant protocols in enabling reliable and intelligent system orchestration for 5G networks and beyond running on a mobile edge computing architecture, AI, and blockchain technology. An untraceable incentive mechanism based on the data used by leveraging Ethereum smart contracts is also detailed. A proportional incentive scheme can trigger the entities to contribute to maintaining cross-silo DL transactions continuously. All processes within decentralized transactions are unlinkable to the parties. In short, the transactions occur without revealing parties' information values, which is a part of privacy-awareness in decentralized approaches.

The rest of this paper is arranged as follows. Section 2 presents an in-depth overview of distributed ledger technology with a public ledger embodying transaction records along with blockchain application in the cross-silo decentralized learning (the new technology of trust). The motivation of the paper is outlined in Section 3. Whilst Section 4 elaborates on the privacy awareness in decentralized approaches and the linkability concerns in the smart contracts as a revenue mechanism. We also describe the potential vulnerabilities and defenses in this section. A concrete scheme of secure decentralized transactions is detailed in Section 5. Finally, we draw some concluding remarks in Section 6.

2 Blockchain Insights and Cross-Silo Decentralized Learning

This section provides the essential background related to the decentralized ledger, blockchain tables, and decentralized learning. These approaches are pioneers of the new technology of trust where decentralization is a fundamental component. The implementation of blockchain in decentralized collaborative learning is also detailed.

2.1 The New Technology of Trust

Blockchain smart contract removes trade or service agreement from the realm of static documents that require human management. Smart contracts transform into automation tools that manage complex transactions in the decentralized system. In 2015, Ethereum appeared to the public which adding Turing-Complete smart contracts to the blockchain. Ethereum performs more complex computations, and it manages more responses compared to Bitcoin. However, it is not a self-evolving code. Ethereum is a collection of purely rule-based and recursive programs. A recent study conducted by Kiffer et al. [20] showed that the smart contract's diversity is direct copies of other contracts. Moreover, the smart contracts ecosystem has a considerable lack of diversity since the code is used extensively. Nevertheless, the concern is gradually tackled since various approaches have risen, including in blending quantum and cloud computing [8], artificial intelligence (AI), and blockchain smart contracts.

```
SQL> CREATE BLOCKCHAIN TABLE auditor.ledger_test (id NUMBER, label VARCHAR2(2
))
       NO DROP UNTIL 1 DAYS IDLE
       NO DELETE UNTIL 5 DAYS AFTER INSERT
       HASHING USING "SHA2_512" VERSION "v1";
 2    3    4  CREATE BLOCKCHAIN TABLE auditor.ledger_test (id NUMBER, label V
ARCHAR2(2))
*
ERROR at line 1:
ORA-05741: minimum retention time too low, should be at least 16 days

SQL> CREATE BLOCKCHAIN TABLE auditor.ledger_test (id NUMBER, label VARCHAR2(2
))
       NO DROP UNTIL 16 DAYS IDLE
       NO DELETE UNTIL 16 DAYS AFTER INSERT
       HASHING USING "SHA2_512" VERSION "v1";

Table created.

SQL>
```

Figure 1: Append-only blockchain tables in general (tamper-resistant property) [1].

Blockchain tables are used to deploy centralized blockchain applications with a central authority, namely the Oracle database. This centralized form provides an organization with more customizability and authority to decide who can participate in the network. The authorized users are different database users who trust the Oracle database to manage tamper-proof blockchain transactions. Compared to fully decentralized blockchains, this approach is helpful in situations where a higher throughput and lower latency are favoured over consensus selection. Blockchain tables, in general, provide application transparency security from frauds by other users in the blockchain peer-to-peer network. The frauds can be recognized by verifying rows in the Oracle table. In this sense, this process re-performs the hash value and confirms it with the corresponding value stored.

2.2 Blockchain in Cross-Silo DL

Decentralized learning combines decentralized computation and AI learning models to enhance model training and minimize the risk of privacy breaches in conventional AI techniques. In this context, the

conventional AI technique, such as machine learning, suffers from severe privacy leakage risk by centralizing and aggregating the client's training data that contain private information on a centralized server. Thus, DL as a decentralized machine learning paradigm allows the clients to collaboratively perform AI training without giving raw data containing the client's private information to the central aggregator [18]. The DL approach empowers the clients to perform a local training model that never leaves their own devices. In this sense, the client's raw data is only used to train and update a current global model and send an updated model to the central aggregator in each iteration. Then, the central aggregator generates a new global model by aggregating these updated and trained models gathered from the participated clients to be used in the next iteration. This process is repeated in multiple iterations until the global model achieves a particular accuracy [2].

Based on application characteristics and client setting, DL is divided into two types, i.e., cross-device DL and cross-silo DL. Cross-device DL admits a massive number of clients to participate in model training. Currently, this setting has been widely deployed in consumer digital products, such as Gboard mobile keyboards [15], Android Message [45], and Pixel phones. Contrary, in the context of client setting, cross-silo DL relatively provides only a small number of clients by promising better client reliability and data availability than cross-device DL. In this sense, cross-silo DL ensures that all clients are almost always available and relatively few failures. Further, cross-silo DL can be relevant for sharing incentives among clients who train the model using their data for system improvement.

Although cross-silo DL brings several advantages, the existing framework still potentially experiences various adversarial attacks and concerns, including malicious clients and false data, a single point of failure (SPoF) issue, and the lack of incentives. In order to address these challenges, several works have been proposed the notion of merging the merits of blockchain and DL for the next-generation wireless network [41]. The blockchain-assisted DL approach is used to enhance client privacy and security by recording transactions in immutable distributed ledger networks as well as improving model training in a decentralized manner. Blockchain also might replace the aggregation server in cross-silo DL, which means blockchain nodes can perform the global model aggregation task. Furthermore, blockchain with SC features can be deployed as an incentive mechanism to motivate clients to collaborate on the global model improvement using their local data.

3 Motivation

Most of the prior studies have been particularly focused on utilizing public blockchain platforms for various use cases with transparency property. Some researches have either emphasized the merits of blockchain smart contract [3, 29, 4], or utilized blockchain security as presented in [13, 25]. Blockchain technology as an incentive mechanism with different platforms are also deployed in [35, 19, 42]. Nevertheless, the additional security protocols that can be adjusted to the system requirements are rare to be discussed, especially for sensitive data. The selected studies of previous researches in chronological order can be seen in Table 1 that also summarizes the existing methods and highlights the importance of our study in the matter of cross-silo distributed learning environment.

This research aims to bridge the gap by utilizing Monero (XMR) protocols in any decentralized transactions activities. This research emphasizes the present state of the Ethereum blockchain smart contract as a backbone technology in conducting decentralized transactions over peer-to-peer networks. Smart contracts are self-executing contracts with irreversible data records that can be a plausible solution in propagating incentives within distributed learning schemes. However, the two-phase commit designed between users and model providers through the blockchain environment could violate users' privacy contrary to the principal objective of DL. Hence, by adopting inference attacks, the observer can infer and link specific data features of the private dataset. This research investigates several pseudonymiza-

Table 1: Summary of previous researches in chronological order.

Research Paper	Year	Objective	Demerit	Significance of the Study
Rahmadika and Rhee [37]	2021	Untraceable transactions in the cross-silo federated learning	Centralized aggregation server (irreplacable)	Empirical benchmark
Rahmadika, Firdaus et al. [34]	2021	An intelligent cross-silo FL based on distributed ledger	For general use cases only (specific assumption)	Empirical and performance benchmark
Shuaicheng et al. [28]	2021	Privacy model for federated learning running on BC	Impractical for several use cases	Security and privacy trade-offs
Ayaz et al. [3]	2021	BC and FL for message dissemination in VANETs	Linkability concerns in communication exchanges	Unlinkable techniques
Mahmood and Jusas [29]	2021	BC and FL adoption for classification problems	BC platform's performances are not defined	Performance benchmark
Qu and Wang et al. [33]	2021	FL adoption as a PoW consensus in blockchain	BC hardfork / radical changes are required	Separation practices benchmark studies
Feng et al. [13]	2021	Blockchain to provide security in the MEC system	Transparency and untraceability issues	Untraceable incentive mechanism
Rahmadika and Rhee [35]	2020	Reliable FL with blockchain-based incentive	Impractical to be adopted in the smart contract	Performance benchmark and evaluation
Zhang et al. [46]	2020	BC with FL for device failure detection in industrial IoT	Encryption techniques are not elaborated	Privacy-preserving protocols
Rehman et al. [43]	2020	Blockchain-based reputation-aware fine-grained FL (trustworthy FL in MEC system)	Impractical to be adopted in the smart contract	Benchmark problems - centralized aggregation server/data centers
Khan et al. [19]	2020	Resource optimization and distributed reward	Linkable-reward mechanism	Untraceability features are required
Kumar et al. [25]	2020	Enhancing privacy in BC enabled FL	BC's performances were not elaborated	Performance benchmark and evaluation
Bao and Su et al. [4]	2019	A healthy marketplace with BC and collaborative training	Undefined privacy-preserving protocol and training auditing	Privacy preservation protocols
Kang et al. [17]	2019	Reliable FL reliable with multiweight suebjective logic	Privacy-awareness is beyond the research	Empirical analysis
Toyoda et al. [42]	2019	An incentive-aware BC with FL (incentive compatibility)	Key idea and design (theoretically)	Performance benchmark & evaluation

tion techniques in obscuring decentralized transactions on the peer-to-peer network. This research also outlines several points related to the concept, design and linkability issues of implemented schemes.

4 Privacy Awareness in Decentralized Approaches

First, this section delivers privacy awareness in decentralized approaches by stating the current state of the blockchain-based decentralized learning environment. The existing contemporary works are also presented. At the end of this section, the linkability concerns of decentralized learning with blockchain-based incentive mechanisms are also discussed. We also highlight some research questions.

4.1 Blockchain-based Decentralized Learning

The conventional DL approach still faces several challenges that need to be addressed, especially in privacy problems, such as membership inference attacks, data poisoning attacks, malicious clients, dishonest central aggregator servers, and the possibility of SPoF occurring. In the membership inference

attack, attackers might perform reverse engineering to gather the client's private data by leveraging the updated model training. In contrast, a poisoning attack affects the global model by sending the malicious updated models during the collaborative training phase. Furthermore, the central aggregator responsible for managing whole system orchestration has trouble addressing crucial challenges associated with the SPoF issue, which may cause the risk of client information being possibly exposed. As a result, the clients could be reluctant to participate in improving the cross-silo DL system. On the other hand, blockchain is an open database that guarantees data security by supporting trustworthy and anonymous transactions without requiring any intermediaries. Moreover, those transactions are recorded on the distributed and immutable ledger [14]. Therefore, blockchain has also been exploited to tackle the several flaws of conventional DL, as mentioned above. We investigate that the blockchain-based DL approach has at least the following advantages:

(i) Blockchain can avoid SPoF and achieve decentralization by replacing the aggregation server (centralized approach) and allowing more than one blockchain node to execute the model aggregation to be the global model in the cross-silo DL system.

(ii) The verification mechanism in the blockchain system can filter unreliable data from malicious clients or other attacks (e.g., inference membership attack and data poisoning attack) before it is stored and aggregated to be a global model. In this sense, only valid data will be aggregated to the global model, while the unreliable data of local model updates will be detected in the verification mechanism.

(iii) Blockchain can provide decentralized transactions, which means all transactions are marked with a timestamp, and then a particular consensus mechanism validates and stores the verified transaction on the distributed database network. Hence, the system allows all involved participants in the blockchain network to obtain an updated ledger automatically.

(iv) Blockchain with SC can be deployed to address the lack of incentives as one of the flaws in conventional cross-silo DL. Here, blockchain can be utilized to distribute incentives (i.e., rewards) to clients. Thus, the incentive mechanism can encourage the clients to honestly contribute to training their data using their computational resources to improve the global model.

The blockchain-based DL has been mentioned in various current existing works. In [27], the BLADE-FL framework is proposed, which aims to deploy the fully decentralized model aggregation. Moreover, this framework provides a reliable learning environment by encouraging a self-motivated approach for clients. The smart contract is designed to integrate clients' mining and training tasks to calculate and update a global model. BlockFL architecture is proposed by Kim et al. [21] that focuses on providing a decentralized manner by removing the aggregation server. BlockFL, as the combination of blockchain and FL, is used to verify and exchange the local model updates generated by clients without coordination from a centralized server. Figure 2 illustrates the framework of conventional DL and blockchain-based DL. In general, as shown in Figure 2(b), blockchain-based DL allows the client to perform local model training and verify the updated models to generate the global model in a decentralized manner. Weng et al. [44] proposed DeepChain as a fair, secure, and distributed protocol by providing an incentive mechanism based on blockchain to motivate clients to behave correctly in the system. DeepChain protocol requires every user to state their asset to access the system and perform their task to train the DL model collaboratively. In this regard, users UEx_n send the asset transaction $Tx\_UEx_{n(Asset)}$ using their pseudonymous address public key PK_UEx_n to prove their asset UEx_n_Asset ownership (see formula

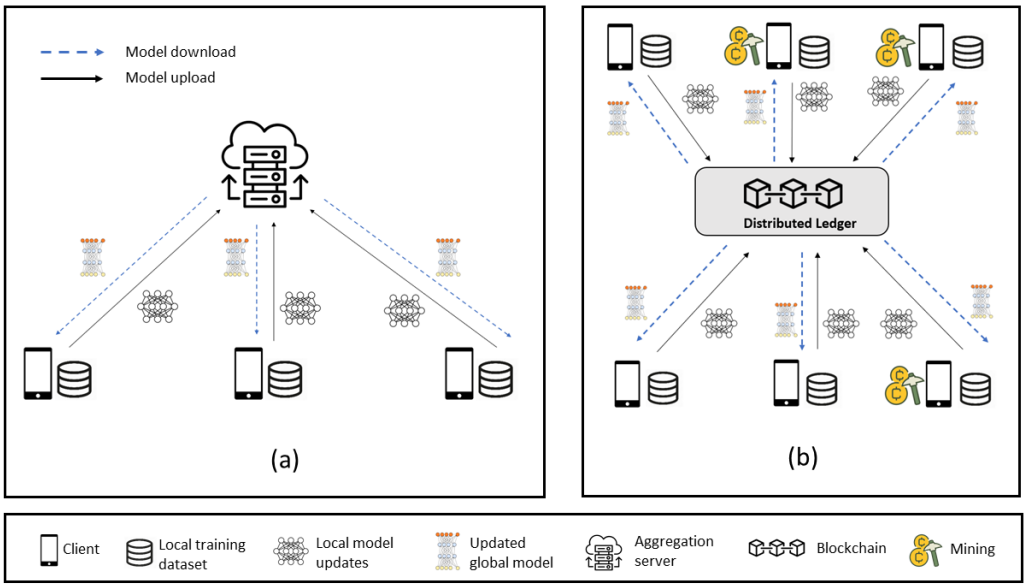


Figure 2: (a) Conventional decentralized learning; (b) Blockchain-based decentralized learning.

[44]).

$$
\begin{aligned}
Tx\_UEx_{n(Asset)} = PK\_UEx_n \rightarrow \Big\{ \Big(& PK_{UEx_n\_Asset} = PubKey^{Hash(UEx_n\_Asset)}, \\
\alpha_j\_UEx_n = & \left(Hash_{(j)}.PubKey^{Hash(UEx_n\_Asset)}\right)^{Hash(UEx_n\_Asset)} \Big), \text{"}Asset\_desc\text{"} \Big\}, \\
where \;\; PK\_UEx_n \in & \left\{ {PubKey_1}^{Sec\_UEx}, {PubKey_2}^{Sec\_UEx}, ..., {PubKey_n}^{Sec\_UEx} \right\}
\end{aligned}
\tag{1}
$$

Formula 1 describes the asset statement transaction of UEx_n. It consists of the client's asset pseudonym public key $PK_{UEx_n\_Asset}$, the proof of UEx's asset ownership $\alpha_j\_UEx_n$ and the UEx_n's asset descriptions "$Asset\_desc$" (e.g., data topic, data format, and data size). Here, $PK_{UEx_n\_Asset}$ and $\alpha_j\_UEx_n$ are essential components that can be utilized to prove the client's asset ownership and ensure that UEx_n_Asset can not be revealed. These asset statement components are composed of the collision-resistant hash function for mapping UEx's assets $Hash(UEx_n\_Asset)$ and the unique-generated public keys ${PubKey_1}^{Sec\_UEx}$ with corresponding private keys to maintain pseudonymity. In order to form a fair and secure collaborative transaction, DeepChain proposed a collaborative information commitment including the number of clients UEx_n, index of the current iteration t, the parameter of threshold using Threshold Paillier algorithm $Thres_{MDL}$, client's commitment ${Commit.}^{Sec}$, collaborative global model $\phi glb_{(x)}$, the initial weights $W_{0,j}$ and the amount of client's deposit UEx_{coin}. In short, all those information are recorded in a collaborative transaction and associated with being a collective address PK_{collab} before uploading it to DeepChain [44].

$$
Tx_{(collab)} = PK_{collab} \rightarrow \left\{ UEx_n, t, Thres_{(MDL)}, {Commit.}^{Sec}, \phi glb_{(x)}, W_{0,j}, UEx_{coin} \right\} \tag{2}
$$

The fundamental DL model with blockchain-based revenue is shown in Algorithm 1. The high-level model is divided into three procedures: server update mechanism, user update, and blockchain revenue. In the server update mechanism, the model provider estimates the number of potential users $Poss.\_UEx_n$ to be included to build the model. Several requirements need to be met, such as device type,

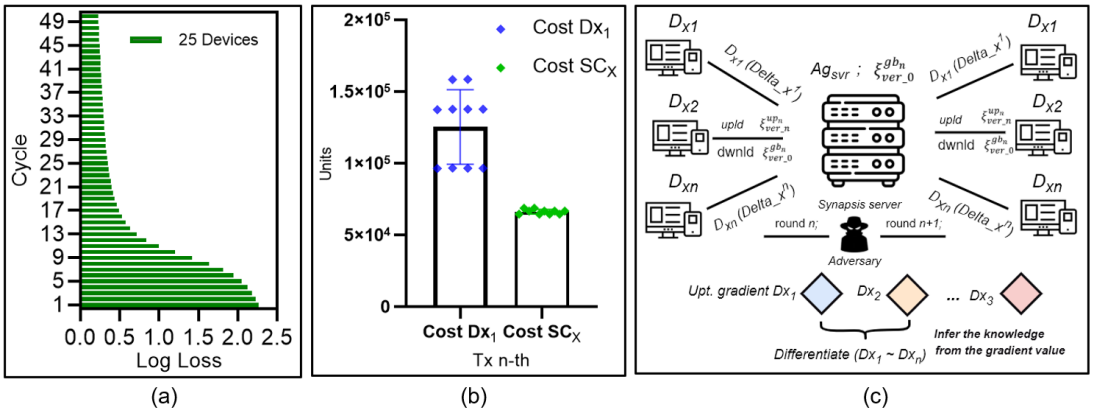
Algorithm 1 Fundamental DL technique with BC-based revenue. BSz is the global batch size; UEx_n are users indexed by i; where Min_BSz is local minibatch size, $Lrate$ is learning rate, inspired by [23].

procedure SERVERUPDATE_MECHANISM:
  Agg_{svr} roughly mapping $Poss.\_UEx_n$ **aggregation server initialized the potential users*
  for each round t = 1,2, ..., n **do**
    $Poss.\_UEx_nglb \leftarrow maximum(BSz \cdot UEx_n\ Poss._{(i,j)})$
    $Poss.\_UEx_n \leftarrow$ (subset $Poss.\_UEx_nglb$ users) **the users are a subset of total global UEx_n available*
    for each user $i \in Poss.\_UEx_n$ **in parallel do**
      $Poss.\_UExn_t^i \leftarrow$ UserUpdate$(i, Poss._UExn_t)$ **$\forall$ updated model is based on $Poss.\_UExn_{glb}$*
    end for
    $Poss.\_UExn_u \leftarrow \frac{1}{nt}\sum_{i \in Poss.\_UExn_n} Poss.\_UExn_u^i$
    $Agg_{svr} \leftarrow Poss.\_UExn_{t+1} = Poss.\_UExn + LratePoss.\_UExn_u$ **gathering updated gradient*
  end for
end procedure
procedure USERUPDATE:$(i, Poss.\_UEx_n)$ **the updated model is broadcasted to the users*
  //Executes on user i
  $Min\_BSz \leftarrow$ (split data i into batches of size Min_BSz)
  for every local epoch nt from 1 to E **do**
    for batch $Min\_BSz_1 \in Min\_BSz$ total **do**
      $Poss.\_UExn_{i(M_{DL})} \leftarrow Poss.\_UExn_t^i - Poss.\_UExn_t$; **$\forall$ UEx_n receive the updated model*
    end for
  end for
  return $Poss.\_UExn$ list to aggregation server Agg_{svr}
end procedure
procedure ETHER_REVENUE(Rv_UEx_n, Rv_{mg}) **incentivized using Ethereum platform*
  Agg_{svr} collects the list of sender (users) $UEx_{n1}, UEx_{n2}, ..., UEx_n$
  Active miners $mg_1, mg_2, ..., mg_n$
  for $mg_1, mg_1, ..., mg_n \in Miner_{tot}$; Agg_{svr} **do**
    $Agg_{svr} \leftarrow$ ConfirmTransaction $H(Poss.\_UExn_t^1, Poss.\_UExn_t^2, ..., Poss.\_UExn_t^{nt})$
    **Agg_{svr} has the list of users*
    *Other miners validate the result till it gets confirmed
    Rv_UEx_n are given to $UEx_{n1}, UEx_{n2}, ..., UEx_n$ **the rewards are distributed to the users*
    Rv_{mg} are distruted to $mg_1, mg_1, ..., mg_n \in Miner_{tot}$ **mining reward for the miners*
  end for
end procedure

the network latency, the minimum number of the dataset, and to name a few. The users send the gradient model back and forth to the model provider after finishing the private training using their dataset. The users are rewarded through the Ethereum blockchain whenever the UEx's transactions are satisfied the requirements. The revenue Rv_UEx_n belongs to the data owner, and the revenue Rv_{mg} is mining revenue for the miners (automatically distributed). We detail this point in Section 5.

4.2 Linkability Concerns at a First Glance

The merits of Ethereum smart contracts can be utilized as a rewarding platform in the DL scheme. It is irreversible and tamper-proof that can solve the dispute between parties. This section illustrates an incentive mechanism in distributed learning by leveraging the Ethereum smart contract to tackle the intermediaries issues in the centralized incentive schemes. Nevertheless, concerns arise if this scheme is implemented for sensitive data such as health-related data since the gradient values [24] from users are exposed publicly through the smart contract. The observer can adopt the active and passive inference attack with certain assumptions to impose training data through gradients value that breaks privacy. We



(a) (b) (c)

Figure 3: (a) Accuracy performance of DL learning; (b) The average amount of gas used by devices and smart contract manager; (c) Inferring the users' information.

suggest the reader refer to our prior work in [36] to comprehend the DL model setting details. Other similar attacks and concerns are also described in [40, 10] .

The users are data owners who possess a large amount of private training data, while the model provider preserves the deep learning model that can be downloaded publicly. The gradient values are gathered by the provider gradually, which later to be used to compute the aggregation value. The provider provides Ag_{svr} revenue through smart contracts SC_x for those proven to contribute to improving the models (using users' resources). The communication between providers and users is carried out via *two-phase commit* transaction. The performance results can be seen in Figure 3 (a) and (b). The overall results positively recommend that the schemes can be applied to real-world implementation (for non-sensitive data). However, when the users Dx_n deploy the transaction that consists of a cipher to encrypt the information to the provider, the observer can impose the dataset knowledge by adopting active and passive inference attacks [30] as shown in Figure 3 (c). Yet, the performance of the adversary decreases with an increasing number of users. Blockchain can be utilized even further as a decentralized gradient counter. In this sense, the role of the centralized aggregation server can be replaced with distributed nodes scattered in the network.

5 Secure Decentralized Transactions

This section presents the techniques in securing decentralized learning transactions where the entities can securely conduct several activities. The entities' information remains secret. We also investigate the pseudonymous rewarding mechanism by utilizing the XMR protocols. In the final section, comparative analysis are discussed.

5.1 Secure Decentralized Learning Transactions

Privacy preservation and anonymity are the most paramount aspects of decentralized transaction activities. Peer-to-peer transactions seek to be concealed from the observer's view, and a distinct difference resembled the centralized approach. In particular, CryptoNote protocol stated two properties that a fully anonymous cryptocurrency standard must meet to comply with the requirements presented by Okamoto and Ohta [31] as follows: (i) **Untraceable transactions**. For the respectively incoming transactions, the probability of all potential senders is equiprobable. (ii) **Unlinkable transactions**. It is extremely hard to prove whether the same entity sent the two or more outgoing transactions.

Perversely, most decentralized transactions activities from many different platforms do not satisfy both requirements, especially the untraceability property. Every transaction can be unambiguously traced to a unique origin and final recipient since the transactions are conducted on the public network. Even when entities indirectly exchange the data or funds, adequately engineered path-finding algorithms will expose the source and final recipient. In regards to the cryptocurrency as a reward in DL activities, Bitcoin does not meet the unlikability property. A sophisticated blockchain analysis can unveil a relationship between the users of the Bitcoin network and their associated transactions. Therefore, several techniques have emerged to answer these challenges, such as Monero (XMR), which applies the CryptoNote protocol as the backbone of the cryptocurrency. In this research, we utilize some properties of the CryptoNote protocol to support the pseudonymous of decentralized learning activities.

The first feature begins with modifying the ring signature algorithm, where BPx be the $ed25519$ basepoint as part of Edwards-curve digital signature algorithms using secure hash algorithms 512 (SHA-512) and a Curve25519 ($q = 2^{255} - 19$) with a twisted Edwards curve shown in formula (3). For detailed information, we recommend readers refer to fast explicit formulas techniques for group operations on an Edwards curve, elaborated in [6].

$$\begin{aligned} -CorX^2 + CorY^2 &= 1 - \frac{121665}{121666} CorX^2 \cdot CorY^2, \\ Where\ l &= 2^{252} + 2774231777372353 \ldots and\ c = 3 \\ CorX &= \frac{ValU}{ValV}\sqrt{-4886664},\ with\ CorY = \frac{ValU - 1}{ValU + 1} \\ &(equivalent\ to\ the\ Montgomery\ curve) \end{aligned} \tag{3}$$

We note that every hash value produces a point in accumulating the base point BPx ($Hash = \psi BPx$ for any undefined ψ). Contrary to what occurs in $secp256k1$ (the curve is leveraged in the Bitcoin cryptocurrency). Suppose $Commit.(a, CorX) = CorX \cdot BPx + a \cdot Hash$, the commitment to the value a with mask $CorX$. We realize that as long as $log_{BPx}\ Hash$ is defined, while $a \neq 0$, then $log_{BPx}\ Commit.(a, CorX)$ is remain unspecified. Contrary, with the value of $a = 0$, then $log_{BPx}\ Commit.(a, CorX) = CorX$. In this sense, it is possible to sign with the secret of sender's private key. Eventually, the networks can check whether the input commitments and output commitments are $\sum Inputs = \sum Outputs$. Yet, these properties are not sufficient in XMR because the given transactions TXs consists of multiple potential inputs $Poss._i, i = 1, 2, 3, \ldots, n$, where only one of which corresponds to the sender. This concern is not expected because it eliminates the anonymity produced by the ring signatures protocol. Hence, the commitments are constructed in (4) as follows:

$$\begin{aligned} Commit._{inputs} &= CorX_{commit.} \cdot BPx + a \cdot Hash \\ Commit._{(output-1)} &= CorY_1 \cdot BPx + b_1 \cdot Hash \\ Commit._{(output-2)} &= CorY_2 \cdot BPx + b_2 \cdot Hash \\ Commit._{(output-n)} &= CorY_n \cdot BPx + b_n \cdot Hash \end{aligned} \tag{4}$$

The constructed ring signatures protocol consists of all $Commit._i, i = 1,\ sec, \ldots, n$ with sec is the secret key index of the commitment of the sender, defining the corresponding public key. In this sense, the commitments and public keys are paired $(Commit._i, Poss._i)$; while the subtracting $\sum Commit._{out}$ is generated in advance. In this case, the commitments are designed in formula (5). This formula is a ring signature that can be signed since the senders have knowledge of the private keys. Precisely, due to the knowledge of private key for $Poss._i$ and the private key for $Poss._i + Commit._{i,in} - \sum_j Commit._{j,out}$, the sender can conduct a signature for any desired transactions. The signer can use the formula (5) to sign a

transaction, as we defined in our previous work in [37].

$$\Bigg\{ Poss._{(1)} + Commit._{(1,in)} - \sum_{j} Commit._{(j,out)}, ..., \; Poss._{(sec)} + Commit._{(sec,in)} - \sum_{j} Commit._{(j,out)}, ..., \; Poss._{(}n) + Commit._{(n,in)} - \sum_{j} Commit._{(j,out)} \Bigg\}. \tag{5}$$

$$DL\_Tx\phi_1\_req. \begin{cases} \bullet \; RNGsgn_{(i,j)} \in RNGtot \equiv v \; \textbf{"AND"} \\ "Tx\phi_n\_req." = True \rightarrow then "Approve"; \; Otherwise: \\ \bullet \; RNGsgn_{(i,j)} \notin RNGtotv \; \textbf{"OR"} \\ "Tx\phi_n\_req" = False \rightarrow then "Decline" \end{cases} \tag{6}$$

For ease of understanding, we construct a scenario where the users desire to use a DL model $\phi glb_{(x)}$ uploaded onto cloud services. The users are required to deploy a transaction request that states the desired global model along with relevant information (the model owner governs the necessary inputs). The users select the number of signatures to be applied to sign a request transaction $DL\_Tx\phi_1\_req.$; where this is can be understood as a pseudonymous request transaction with an anonymity feature. The model owner can frequently change the format of transactions, yet the protocols behind the transactions remain the same. When the model owner receives the $DL\_Tx\phi_1\_req.$ sent by users via a secure channel, then the owner checks the condition of the transaction to confirm the completeness of requirements as defined in (6). In the first place, the model owner checks the correctness of the ring signatures that he received beforehand (it belongs to the group or not). The owner repudiates the transactions if one of the conditions is not satisfied. The models are sent to the users if only conditions in (6) are met.

5.2 Pseudonymous Rewarding Mechanism and Comparative Analysis

In practice, most incentive schemes are centralized where the third party has a vital role in conducting transactions. The centralized nature is inherent to the single point of failure (SPoF) [39] and bottleneck issues that jeopardize the system's root. On the other hand, with its merits, blockchain can be a plausible solution to tackle the problems since it runs on top of the peer-to-peer network where no single intermediary has complete control over transactions. Accordingly, blockchain-based reward mechanisms have been heavily adopted in many various use cases. For instance, Lin et al. [26] utilized blockchain to deliver rewards in the energy-knowledge trading environment. In comparison, Chakrabarti et al. [9] proposed a blockchain-based reward scheme for opportunistic disaster communication over a delay tolerant network. Nevertheless, straightforwardly applying blockchain-based incentives for sensitive information is not desirable. This section relies on several XMR protocols' features to support pseudonymous rewarding in decentralized learning.

The users are rewarded since they build the DL model collaboratively by broadcasting the updated gradient values to the model provider. To be incentivized, the sender generates a Diffie-Hellman key exchange protocol to obtain a shared secret key from his data and half of the recipient's address. The users $UEx_{(i,j)}$ are also needed to tender a new transaction via a private Ethereum smart contract. This transaction is denoted as $DL\_Tx\phi_n\_Rwd.$ that proves the user's contribution in building a DL model using their corresponding valuable datasets. The $DL\_Tx\phi_n\_Rwd.$ transaction has a distinguish feature with $DL\_Tx\phi_1\_req.$ transaction, where within $DL\_Tx\phi_n\_Rwd.$ there are a pair of public keys (*PubA* and *PubB*). The first public key *PubA* is performed by the UEx_n using UEx_n's private key *SecA* blended with a base point BPx_A; $PubA \rightarrow SecA \cdot BPx_A$. Similarly, the other public key *PubB* is generated from a private key *SecB* with their respective generator BPx_B; $PubB \rightarrow SecB \cdot BPx_B$. The produced public key must be

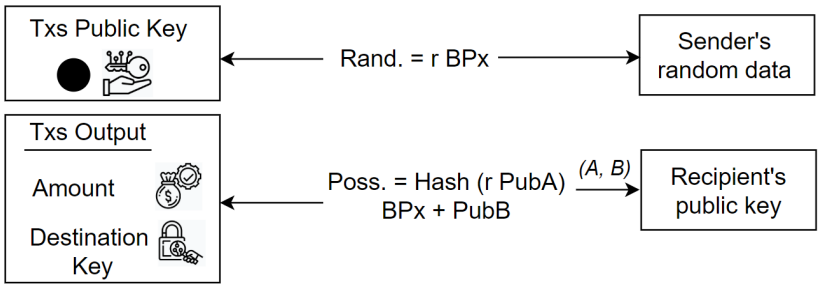


Figure 4: Structure of standard transactions.

unique since the private keys are different from others since the generators are also unique $SecA \neq SecB$ "AND" $BPx_A \neq BPx_B$.

$$DL\_Tx\phi_n\_Rwd. = \frac{\phi glb_{(info)} || \phi up^{\delta_1} || \delta_1 knowledge}{\frac{UEx_n{}'s\ PubKey\ \rightarrow PubA, PubB\ (Generated)}{\{RNGsgn_{(i,j)} \in RNGtot \geq 1 || SecA\} \rightarrow Signed}} \tag{7}$$

$$Dest.\psi = BPx_n + UEX_n\_PubB \cdot Hash(random\_UEX_n\_SecA) \tag{8}$$

$$Spend\,\psi = SecB + Hash(SecA \cdot random \cdot BPX_n) \tag{9}$$

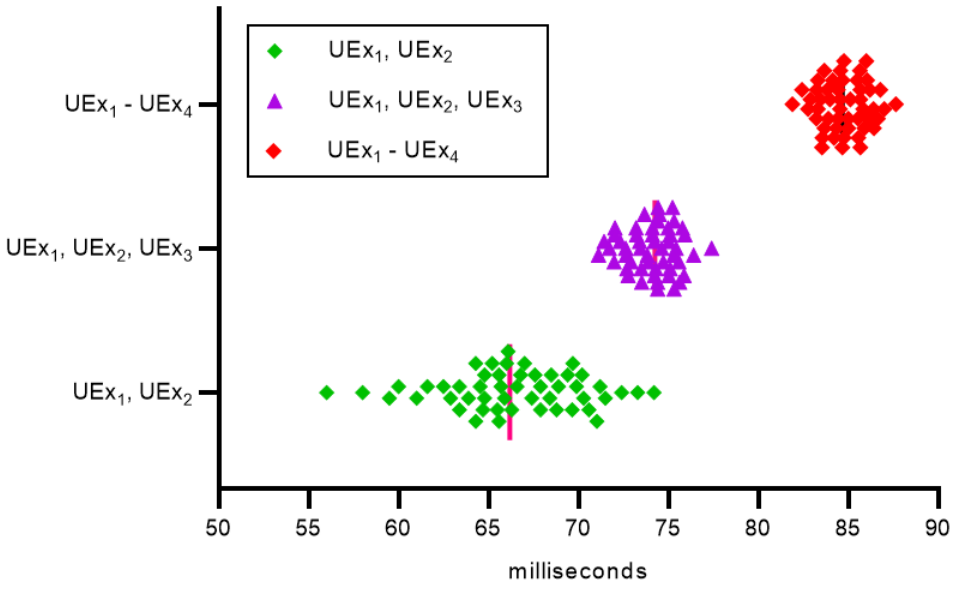


Figure 5: The illustration of distribution points of time spent by UEx_n to generate a shared public key. This process is carried out for 50 repetitions for each number of RNG_{sgn}.

Formula (7) represents the UEx_n's reward transaction. It consists of the global model of DL's information $\phi glb_{(info)}$, the updated gradient values ϕup^{δ_1}, and the knowledge of dataset $\delta_1 knowledge$. The parameters of the transaction can be adjusted accordingly. The user UEx_n also attached a pair of public keys ($PubA$ and $PubB$) within transactions which are derived from $SecA \neq SecB$ "AND" $BPx_A \neq BPx_B$.

Suppose $random = Owner's\ random\ data \rightarrow R = random \cdot BPx_n$ is part of Diffie Hellman exchange principle. If $DL\_Tx\phi_n\_Rwd.$ condition is satisfied, the DL model provider can unpack the public key attached to the transaction. The provider then generates another random base point $rbp \in [1, l-1]$ and manages the one-time destination key $Dest.\psi$ as depicted in 8 to be sent back to the UEx_n via Ethereum smart contract. The UEx_n as a recipient checks every passing blockchain transaction using his private key *SecA* and *SecB*. UEx_n will be able to recover the respective one-time private key to spend the Ether since only UEx_n knows about *SecA* and *SecB* as shown in formula (9). In short, UEx_n is the only legitimate user.

Table 2: Ethereum daily gas used information (historical total daily).

No.	Date (UTC)	POSIX time	Total Value
1	Jan. 1st, 2021	1609459200	80034402241
2	Feb. 1st, 2021	1612137600	79242528840
3	Mar. 1st, 2021	1614556800	79006491494
4	Apr. 1st, 2021	1617235200	79787949899
5	May. 1st, 2021	1619827200	94431457868
...	...	...	...
9	Sep. 1st, 2021	1630454400	99646996226
10	Oct. 1st, 2021	1633046400	99940537596

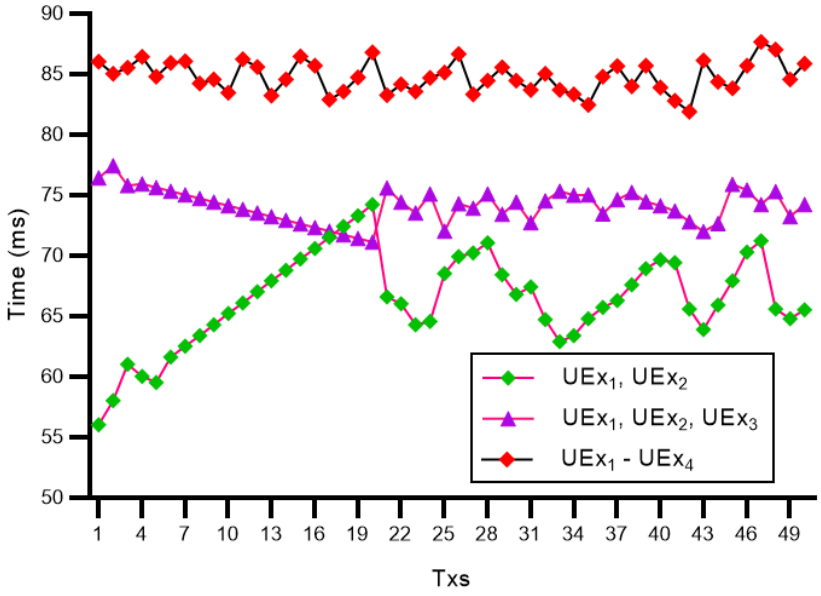


Figure 6: The sequence number of 50 shared key generations for each different RNGsgn.

In terms of time spent in generating a shared key for entities is depicted in Figure 5 (XMR-based protocols). This illustration figure is representative of distribution points of time spent by UEx_n in performing a shared public key to be used to conduct a transaction. For another perspective view, the transactions in sequential order can be seen in Figure 6. This process is carried out for 50 repetitions with a varying number of UEx_n. Suppose user 1 UEx_1 gathers another public key to create a group ring of signature. First, UEx_1 selects one public key of UEx_2. He also chooses the other two public keys (UEx_1, UEx_2, UEx_3). Finally, for the last transaction, UEx_1 uses four public keys including himself

Table 3: Performance benchmark with several existing approaches.

Study	Year	BC Platform	Concerns Addressed	Unlink. TXs
Our approach	2021	Ethereum	Unlinkability TXs and privacy preserv.	Yes (Prot.)
Feng et al. [13]	2021	Hyperledger	Security and trust issues over MENs	Yes (Part.)
Ma et al. [28]	2021	SCx-based	Transparency and contribution evaluation	N/D
Ayaz et al. [3]	2021	SCx-based	Message dissemination in VANETs	N/D
Fan et al. [12]	2020	FISCO-BCOS	Auditable rational reverse auction	N/D
Sandi et al. [35]	2020	Ethereum	Commensurate decentralized incentive	N/D
Khan et al. [19]	2020	N/D	Incentive-based entities interaction	N/D
Weng et al. [44]	2019	Corda	Decentralized incentive mechanism	Yes (Part.)

Prototype (Prot.); Partial (Part.); Not defined (N/D);

(UEx_1, UEx_2, UEx_3,and UEx_4).

The average generating time for the first creation ($UEx_1 and UEx_1$) was 66.415ms, with the fastest and longest time were 55.876ms and 76.263ms, respectively. The same procedures are applied to the second creation, where UEx_1 selects UEx_1, UEx_2, UEx_3 keys. The average generating time was recorded at 74.103ms, with the fastest time was 71.133ms, and the longest time was recorded at 78.274ms. Eventually, the last recorded generation time of the shared key is carried out by selecting four keys of the users. The average time was 84.768ms, with the fastest and the longest time were 82.776ms and 86.055ms. The simulation results show that the more keys involved, the more time it takes to create a shared key. This concern becomes essential to be taken into account since it directly affects the gas usage in the Ethereum smart contract transactions. The amount of gas usage continues to increase over time, affecting the gas limit and gas price. In the end, the uncertainty of the Ethereum gas affects the cost fee per transaction. Therefore, the effective design of smart contracts is essential. The increasing number of Ethereum gas usage can be seen in Table 2 that describes the Unix time-stamps and the values. The data was recorded from January 1st, 2021, up to October 1st, 2021 [38]. We also recommend readers refer to our previous works in [34] and [37].

Finally, we emphasize the unlinkability feature in decentralized learning transactions with several existing approaches, as highlighted in Table 3. We limit the transactions into two categories, namely, a request transaction and distributed reward transaction. Our proposed scheme covers both transactions where also can be adjusted by following the design requirements. However, we only embed the protocols into smart contract transactions in a prototype form due to the hard fork concerns that require radical changes to the entire Ethereum networks. The previous work in [13] and [44] also provided some parts of privacy techniques in decentralized transactions activity. However, the precise methods of unlinkability features were not described in the paper. In contrast, other works listed in Table 3 directly adopted the blockchain and decentralized learning without addressing the linkability issues. This research fills the current literature gap concerning a recent systematic mapping study to preserve privacy in decentralized transactions.

6 Conclusion

We have presented and investigated the pseudonymization techniques in decentralized transactions. Cross-silo distributed learning and blockchain smart contracts emerged as backbone technologies to tackle several issues in the centralized system. This investigation becomes essential to be discussed since many existing schemes only provide privacy protocols, yet the linkability concerns are beyond the topics.

Thus, we utilized the XMR protocols to be adopted into the DL system with a blockchain-based incentive mechanism that can cover many centralized issues. Simulation results and performance benchmarks indicate that the XMR protocols can be a plausible solution to address the linkability concerns inherent to the public blockchain in general. In order to be fully implemented into the Ethereum network, a radical change (hard-fork) is required that makes predecessor invalid blocks and transactions become valid and vice versa. Therefore, we limit this research by implanting the designed protocols into smart contract transactions, where the protocols can be adjusted proportionately. Apart from the benefits of the proposed scheme, several points need to be explored further, such as the aggregation server involvement (irreplaceable), the amount of gas usage in the Ethereum, the devices availability, and to name a few. These points are part of our research interest for the long run.

Acknowledgments

This research was supported in part by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2021-2020-0-01797) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation), and in part by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (2021R1I1A304659011).

References

[1] Blockchain table. `https://docs.oracle.com/en/database/oracle/oracle-database/` [Online; accessed on September 20, 2021].

[2] T. T. Anh, N. C. Luong, D. Niyato, D. I. Kim, and L.-C. Wang. Efficient training management for mobile crowd-machine learning: A deep reinforcement learning approach. *IEEE Wireless Communications Letters*, 8(5):1345–1348, October 2019.

[3] F. Ayaz, Z. Sheng, D. Tian, and Y. L. Guan. A blockchain based federated learning for message dissemination in vehicular networks. *arXiv preprint arXiv:2109.06667*, September 2021.

[4] X. Bao, C. Su, Y. Xiong, W. Huang, and Y. Hu. Flchain: A blockchain for auditable federated learning with trust and incentive. In *Proc. of the 5th International Conference on Big Data Computing and Communications (BIGCOM'19), Shandong Province, China*, pages 151–159. IEEE, August 2019.

[5] D. Berbecaru, A. Lioy, and C. Cameroni. Supporting authorize-then-authenticate for wi-fi access based on an electronic identity infrastructure. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, 11(2):34–54, June 2020.

[6] D. J. Bernstein and T. Lange. Faster addition and doubling on elliptic curves. In *Proc. of the 2007 International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT'07), Kuching, Malaysia*, volume 4833 of *Lecture Notes in Computer Science*, pages 29–50. Springer, Berlin, Heidelberg, December 2007.

[7] A. N. Bhagoji, S. Chakraborty, P. Mittal, and S. Calo. Analyzing federated learning through an adversarial lens. In *Proc. of the 36th International Conference on Machine Learning (ICML'19), California, USA*, volume 97, pages 634–643. PMLR, June 2019.

[8] D. Caputo, L. Verderame, A. Ranieri, A. Merlo, and L. Caviglione. Fine-hearing google home: why silence will not protect your privacy. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, 11(1):35–53, March 2020.

[9] C. Chakrabarti and S. Basu. A blockchain based incentive scheme for post disaster opportunistic communication over dtn. In *Proc. of the 20th International Conference on Distributed Computing and Networking (ICDCN'19), Bangalore, India*, pages 385–388. ACM, January 2019.

[10] R. Chen, J. Guo, D.-C. Wang, J. J. Tsai, H. Al-Hamadi, and I. You. Trust-based service management for mobile cloud iot systems. *IEEE transactions on network and service management*, 16(1):246–263, December 2018.

[11] A. De Villiers and P. Cuffe. A three-tier framework for understanding disruption trajectories for blockchain in the electricity industry. *IEEE Access*, 8:65670–65682, March 2020.

[12] S. Fan, H. Zhang, Y. Zeng, and W. Cai. Hybrid blockchain-based resource trading system for federated learning in edge computing. *IEEE Internet of Things Journal*, 8(4):2252–2264, October 2020.

[13] L. Feng, Z. Yang, S. Guo, X. Qiu, W. Li, and P. Yu. Two-layered blockchain architecture for federated learning over mobile edge network. *IEEE Network*, Early Access:1–14, January 2021.

[14] M. Firdaus, S. Rahmadika, and K.-H. Rhee. Decentralized trusted data sharing management on internet of vehicle edge computing (iovec) networks using consortium blockchain. *Sensors*, 21(7):2410, March 2021.

[15] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*, November 2018.

[16] R. Kanagavelu, Z. Li, J. Samsudin, Y. Yang, F. Yang, R. S. M. Goh, M. Cheah, P. Wiwatphonthana, K. Akkarajitsakul, and S. Wang. Two-phase multi-party computation enabled privacy-preserving federated learning. In *Proc. of the 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGRID'20), Melbourne, Victoria, Australia*, pages 410–419. IEEE, May 2020.

[17] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang. Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory. *IEEE Internet of Things Journal*, 6(6):10700–10714, September 2019.

[18] J. Kang, Z. Xiong, D. Niyato, Y. Zou, Y. Zhang, and M. Guizani. Reliable federated learning for mobile networks. *IEEE Wireless Communications*, 27(2):72–80, February 2020.

[19] L. U. Khan, S. R. Pandey, N. H. Tran, W. Saad, Z. Han, M. N. Nguyen, and C. S. Hong. Federated learning for edge networks: Resource optimization and incentive mechanism. *IEEE Communications Magazine*, 58(10):88–93, November 2020.

[20] L. Kiffer, D. Levin, and A. Mislove. Analyzing ethereum's contract topology. In *Proc. of the 2018 Internet Measurement Conference 2018 (IMC'18), Boston, Massachusetts, USA*, pages 494–499. ACM, October 2018.

[21] H. Kim, J. Park, M. Bennis, and S.-L. Kim. Blockchained on-device federated learning. *IEEE Communications Letters*, 24(6):1279–1283, June 2019.

[22] A. Koloskova, S. U. Stich, and M. Jaggi. Decentralized stochastic optimization and gossip algorithms with compressed communication. *arXiv preprint arXiv:1902.00340*, February 2019.

[23] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon. Federated learning: Strategies for improving communication efficiency. *https://arxiv.org/abs/1610.05492*, October 2016.

[24] P. Kumar, S. Garg, A. Singh, S. Batra, N. Kumar, and I. You. Mvo-based 2-d path planning scheme for providing quality of service in uav environment. *IEEE Internet of Things Journal*, 5(3):1698–1707, January 2018.

[25] S. Kumar, S. Dutta, S. Chatturvedi, and M. Bhatia. Strategies for enhancing training and privacy in blockchain enabled federated learning. In *Proc. of the 6th IEEE International Conference on Multimedia Big Data (BigMM'20), New Delhi, India*, pages 333–340. IEEE, September 2020.

[26] X. Lin, J. Li, J. Wu, H. Liang, and W. Yang. Making knowledge tradable in edge-ai enabled iot: A consortium blockchain-based efficient and incentive approach. *IEEE Transactions on Industrial Informatics*, 15(12):6367–6378, May 2019.

[27] C. Ma, J. Li, M. Ding, L. Shi, T. Wang, Z. Han, and H. V. Poor. When federated learning meets blockchain: A new distributed learning paradigm. *arXiv preprint arXiv:2009.09338*, September 2020.

[28] S. Ma, Y. Cao, and L. Xiong. Transparent contribution evaluation for secure federated learning on blockchain. In *Proc. of the 37th IEEE International Conference on Data Engineering Workshops (ICDEW'21), Chania, Greece*, pages 88–91. IEEE, April 2021.

[29] Z. Mahmood and V. Jusas. Implementation framework for a blockchain-based federated learning model for classification problems. *Symmetry*, 13(7):1116, June 2021.

[30] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov. Exploiting unintended feature leakage in collaborative learning. In *Proc. of the 2019 IEEE Symposium on Security and Privacy (SP'19), San Francisco, California, USA*, pages 691–706. IEEE, May 2019.

[31] T. Okamoto and K. Ohta. Universal electronic cash. In *Proc. of the 1991 Annual international cryptology conference (CRYPTO'91), Santa Barbara, California, USA*, volume 576 of *Lecture Notes in Computer Science*, pages 324–337. Springer, Berlin, Heidelberg, May 1991.

[32] M. Park, S. Kim, and J. Kim. Research on note-taking apps with security features. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, 11(4):63–76, December 2020.

[33] X. Qu, S. Wang, Q. Hu, and X. Cheng. Proof of federated learning: A novel energy-recycling consensus algorithm. *IEEE Transactions on Parallel and Distributed Systems*, 32(8):2074–2085, August 2021.

[34] S. Rahmadika, M. Firdaus, S. Jang, and K.-H. Rhee. Blockchain-enabled 5g edge networks and beyond: An intelligent cross-silo federated learning approach. *Security and Communication Networks*, 2021:1–14, March 2021.

[35] S. Rahmadika and K.-H. Rhee. Reliable collaborative learning with commensurate incentive schemes. In *Proc. of the 2020 IEEE International Conference on Blockchain (Blockchain'20), Rhodes, Greece*, pages 496–502. IEEE, November 2020.

[36] S. Rahmadika and K.-H. Rhee. Enhancing data privacy through a decentralised predictive model with blockchain-based revenue. *International Journal of Ad Hoc and Ubiquitous Computing*, 37(1):1–15, May 2021.

[37] S. Rahmadika and K.-H. Rhee. Unlinkable collaborative learning transactions: Privacy-awareness in decentralized approaches. *IEEE Access*, 9:65293–65307, April 2021.

[38] E. Scan. Ethereum daily gas used chart. `https://etherscan.io/chart/gasused` [Online; accessed on October 15, 2021], October 2021.

[39] V. Sharma, I. You, and G. Kul. Socializing drones for inter-service operability in ultra-dense wireless networks using blockchain. In *Proc. of the 2017 international workshop on managing insider security threats (MIST'17), Dallas, Texas, USA*, pages 81–84. ACM, October 2017.

[40] V. Sharma, I. You, K. Yim, R. Chen, and J.-H. Cho. Briot: Behavior rule specification-based misbehavior detection for iot-embedded cyber-physical systems. *IEEE Access*, 7:118556–118580, May 2019.

[41] R. Shokri, M. Stronati, C. Song, and V. Shmatikov. Membership inference attacks against machine learning models. In *Proc. of the 2017 IEEE Symposium on Security and Privacy (SP'17), San Jose, California, USA*, pages 3–18. IEEE, May 2017.

[42] K. Toyoda and A. N. Zhang. Mechanism design for an incentive-aware blockchain-enabled federated learning platform. In *Proc. of the 2019 IEEE International Conference on Big Data (Big Data'19), Los Angeles, California, USA*, pages 395–403. IEEE, December 2019.

[43] M. H. ur Rehman, K. Salah, E. Damiani, and D. Svetinovic. Towards blockchain-based reputation-aware federated learning. In *Proc. of the 2020 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS'20), Virtual Conference*, pages 183–188. IEEE, July 2020.

[44] J. Weng, J. Weng, J. Zhang, M. Li, Y. Zhang, and W. Luo. Deepchain: Auditable and privacy-preserving deep learning with blockchain-based incentive. *IEEE Transactions on Dependable and Secure Computing*, 14(8):1–18, November 2019.

[45] T. Yang, G. Andrew, H. Eichner, H. Sun, W. Li, N. Kong, D. Ramage, and F. Beaufays. Applied federated learning: Improving google keyboard query suggestions. *arXiv preprint arXiv:1812.02903*, December 2018.

[46] W. Zhang, Q. Lu, Q. Yu, Z. Li, Y. Liu, S. K. Lo, S. Chen, X. Xu, and L. Zhu. Blockchain-based federated learning for device failure detection in industrial iot. *IEEE Internet of Things Journal*, 8(7):5926–5937, October 2020.

Author Biography

Sandi Rahmadika received the Ph.D. degree from Pukyong National University, South Korea. He is currently a Postdoctoral Researcher with the Department of Digital Contents Engineering, Wonkwang University, South Korea. He has authored/co-authored more than 25 papers in academic conferences and journals. He was the Track Chair of the International Symposium and Mobile Internet Security (MobiSec) 2021. His research interests include applied cryptography, privacy preservation in the decentralized system, edge computing, and AI with blockchain integration.

Muhammad Firdaus received his Master of Engineering degree in Telematics and Telecommunication Networks from Institut Teknologi Bandung (ITB), Indonesia. He is currently a Ph.D. student and a member of the Laboratory of Information Security and Internet Applications (LISIA), Pukyong National University. His research interests include applied cryptography, blockchain with AI integration, and communication security.

Yong-Hwan Lee received the MS degree in computer science and PhD in electronics and computer engineering from Dankook University, Korea, in 1995 and 2007, respectively. He is an active member of International Standard committees of ISO/IEC JTC1 SC29 responsible for Image Retrieval and Coding issues. Currently, he is a Professor at the Department of Digital Contents, Wonkwang University, Korea. His research areas include Image Retrieval, Image Coding, Computer Vision and Pattern Recognition, Augmented Reality, Mobile Programming and Multimedia Communication.

Kyung-Hyune Rhee received his M.S. and Ph.D. degrees from the Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea in 1985 and 1992, respectively. He worked as a senior researcher at the Electronic and Telecommunications Research Institute (ETRI), Republic of Korea, from 1985 to 1993. He also worked as a visiting scholar at the University of Adelaide, University of Tokyo, and the University of California, Irvine. He has served as a Chairman of the Division of Information and Communication Technology, Colombo Plan Staff College for Technician Education in Manila, the Philippines. He is currently a professor in the Department of IT Convergence and Application Engineering, Pukyong National University, Republic of Korea. His research interests center on security and the evaluation of blockchain technology, key management and its applications, and AI-enabled security evaluation of cryptographic algorithms.