Research Article

A Secure Incentive Scheme for Vehicular Delay Tolerant Networks Using Cryptocurrency

Youngho Park ¹, Chul Sur,² and Kyung-Hyune Rhee ¹

¹Department of IT Convergence and Application Engineering, Pukyong National University, Busan, Republic of Korea

²Department of Information Security, Busan University of Foreign Studies, Busan, Republic of Korea

Correspondence should be addressed to Kyung-Hyune Rhee; khrrhee@pknu.ac.kr

Received 23 February 2018; Revised 8 May 2018; Accepted 20 May 2018; Published 8 July 2018

Academic Editor: Ilsun You

Copyright © 2018 Youngho Park et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

One remarkable feature of vehicular ad hoc networks is characterized by an opportunistic communications by means of store-carry-forward message relaying which requires the cooperation of vehicles on the networks. However, we cannot be sure that all vehicles willingly contribute their computing resources to the networks for message forwarding with no rewards for their efforts in real-world scenarios. In addition, unfortunately, there may exist some selfish and greedy node which may not help others but tend to take their own gain. To cope with this challenge, incentive mechanisms are generally considered as the promising solution. In this paper, we design a Bitcoin-based secure and reliable incentive scheme for cooperative vehicular delay tolerant networking services. Bitcoin is the well-known worldwide cryptocurrency and digital payment system whose implementation relies on cryptographic techniques, which makes it possible to develop a practical credit-based incentive scheme on the vehicular networks at a low cost. We also implement Bitcoin transaction scripts to handle our proposed incentive scheme.

1. Introduction

It is trend of modern vehicles to equip GPS-based navigation system with digital map and on-board unit (OBU) devices which allow vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications. Due to the advances of modern car technologies incorporating with wireless communication, vehicular communications have been an active research area over the last decade. In particular, we have also witnessed 5G connected vehicular communications test-bed in South Korea recently. As a result, up to date, a variety of vehicular ad hoc network (VANET) applications have been researched to provide not only comfortable transportation services but also location-based infotainment services on the road. Another attractive vehicular network architecture is vehicular delay tolerant networks (VDTNs), focusing on the opportunistic connectivity among vehicles and road side units, in which vehicles contribute to a message relaying service by utilizing store-carry-forward paradigm [1, 2]. As shown in Figure 1, some information collected at a source location (*S*) can be stored, carried, and then forwarded to a destination location (*D*) by a vehicle passing through

the roads. An example of such opportunistic networking applications is to deliver some location-aware information such as gas and parking around *S* to the display located at *D*. Another application is to forward messages to the Internet through the gateway server located at *D* when Internet connection is not possible at *S*.

However, because VANETs and VDTNs are autonomous and self-organized networks with the cooperation among vehicles, we cannot always expect that all vehicles voluntarily contribute their computing resources to the network. Moreover, some selfish vehicles would not help message relaying service for others while they enjoy the services provided by the network. One solution to this challenge is to provide incentives to stimulate vehicles to voluntarily participate in the networks by rewarding for their contribution with an actual money or credit. For example, when a sender asks a vehicle for help, the sender gives some incentive to the vehicle so that the vehicle is willing to store, carry, and forward sender's message to a destination.

Although, on the one hand, an incentive looks like an attractive approach to selfish behavior, another concern is how the sender can be convinced whether the message

FIGURE 1: An example of store-carry-forwarding scenario in VANETs.

is actually delivered to the destination or not [3, 4]. If a source location server first gives credits to a vehicle, a malicious vehicle will not faithfully store-carry-forward the message to the specified destination after receiving the credit. The source location server may be concerned about such malicious behavior of a vehicle so-called dine and dash, such situation is unfair to the source location server. Therefore, it is also a critical challenge how to resolve such unfairness issue of incentive scheme on autonomous vehicular networks.

1.1. Related Work. While vehicular communications have received a great deal of attentions, the researches on secure vehicular communications have also widely carried out, and most of them are focusing on anonymous authentication schemes implemented by digital signature combined with pseudonyms of vehicles so as to guarantee nonrepudiation and anonymity of the communication activities simultaneously [5–12]. However, ordinary digital signature itself cannot deal with the fairness problem. To resolve the fairness problem in store-carry-forward message delivery, Lin et al. proposed a location-release signature [4] in which the credit signed by a source location server becomes valid if the credit reaches to a specific destination location server. However, they did not present how to actual incentives are rewarded to the vehicles.

Incentive schemes for cooperative VANET or VDTN environments can be categorized into reputation-based scheme and credit-based scheme. In reputation-based schemes, network nodes evaluate trust relationship with each other and vote on neighboring node's activity of message forwarding so that uncooperative nodes of bad reputation are excluded from the networks [13, 14]. Li et al. proposed an announcement scheme for VANETs based on a reputation system which allows for vehicles to evaluate message reliability [13]. In their scheme, the reliability of a message is evaluated by the reputation of the vehicle which generates the message, and the reputation score is collected, updated, and certified by a trusted third party.

Credit-based schemes employ some form of virtual currency for regulating message forwarding among different nodes and rewarding nodes for their helps [15–17]. When a source node needs help of other nodes for message forwarding, the source node should pay a certain amount of virtual coins to the helper nodes. To incentivize nodes for DTNs, Zhu et al. proposed a secure multilayer credit-based incentive scheme, named SMART [15], which provides nodes with virtual coins to charge for and reward the provision of data forwarding. They also briefly discussed several security issues in DTNs and countermeasures; however, they did not consider fairness issue. With regard to fairness issues in VDTNs, Lu et al. proposed a secure and practical incentive protocol Pi, which is a hybrid model combining reputation and credit, using verifiably encrypted signature technique.

However, those schemes additionally require implementing an application-dependent reputation management system or a virtual coin management system on VANETs. Furthermore, the existing incentive schemes entirely rely on a central trusted third party to assign some virtual coins to each node and to keep track of issued virtual coins in the system.

1.2. Contribution. In this paper, we present a secure credit-based incentive scheme for cooperative VDTNs integrating with a blockchain-based cryptocurrency system. Bitcoin is the most famous and practical cryptocurrency, whose implementation relies on cryptographic techniques and a distributed electronic payment system in which no trusted third party is required. By taking advantage of the Bitcoin system or Bitcoin overlay network, we can design a secure message delivery service and credit-based incentive scheme for VDTNs at a low cost.

As compared to the existing credit-based scheme, we do not need to be concerned about the reliability of virtual coin rewards on VDTNs. Instead, reliable virtual coin exchange transactions are shifted to the Bitcoin system. Moreover, we do not need to implement public key and pseudonym management system such as vehicular-PKI to authenticate

TABLE 1: Some remarkable cryptocurrencies based on blockchain.

project	launched	anonymity	remarks	ref.
Bitcoin	2009	pseudonymous	(i) first blockchain-based cryptocurrency (ii) uses proof-of-work timestamping	[20]
Namecoin	2011	censorship resistance	(i) first altcoin that implemented Satoshi's BitDNS idea (ii) key-value pair registration and transfer	[21]
Litecoin	2011	pseudonymous	(i) technical detail is similar to Bitcoin (ii) aims to process a block every 2.5 minutes, rather than Bitcoin's 10 minutes (iii) uses scrypt in its proof-of-work algorithm	[22]
Monero	2014	pseudonymous, unlinkable	(i) implemented based on the CryptoNote protocol (ii) uses ring confidential transactions technique	[23]
Ethereum	2015	pseudonymous	(i) provides Turing-complete virtual machine, Ethereum Virtual Machine (EVM) (ii) distributed computing platform featuring smart contract functionality (iii) Ether is a cryptocurrency generated by the Ethereum platform	[24]
Zcash	2016	strong anonymity, shielded transactions	(i) decentralized cryptocurrency that provides strong privacy protections (ii) transactions can be controlled by a zero-knowledge proof called zk-SNARKs	[25]

vehicles participating in store-carry-forward communications because the private and public key pair of Bitcoin account owned by the vehicle/user can be used for vehicular communications on VDTNs as well as handling Bitcoin transactions rewarded as incentives. That is, the Bitcoin public key can be viewed as vehicle's pseudonym.

However, ordinary Bitcoin public keys are not sufficient to incorporate trustworthiness from real-world entities into the system, but vehicular networking is a cyberphysical system implemented in real world. We present authenticated vehicular communication using certified Bitcoin public key technique [18] to validate the legitimacy of the entity taking part in the system. Even though a trusted authority is involved in the proposed system for issuing certified public keys, we can more efficiently implement authentication system than the existing vehicular-PKI.

In our preliminary version of this work [19], we presented how Bitcoin can be used as incentive in VANET but did not show implementation details. In this paper, we show that how the fairness issues of incentive scheme on VDTNs can be resolved by handling Bitcoin transactions which transfer coins to a volunteer vehicle as incentives and implement Bitcoin locking and unlocking scripts to control the redemption conditions to spend the coins specified in the incentive transactions.

The rest of this paper is organized as follows: we first shortly introduce the technical concept of the Bitcoin needed to understand our proposed system in Section 2, then we design the proposed Bitcoin-based secure incentive system on VDTNs in Section 3. We discuss the security features of

the proposed system in Section 4 and finally conclude this paper in Section 5.

2. Background

Cryptocurrency is a digital asset to support a medium of exchange based on cryptography to secure its transactions and to verify the transfer of assets. As opposed to centralized electronic cash and banking systems, cryptocurrencies are maintained by decentralized control through a blockchain functioning as a distributed ledger. Since the first implementation of decentralized cryptocurrency, Bitcoin, numerous alternative coins (altcoins) have been created. Table 1 summarizes some remarkable cryptocurrencies and their technological characteristics. Due to the cost effectiveness in validating transactions and the security of immutable ledgers on a distributed blockchain, the concept of blockchain is evolving to a platform beyond the cryptocurrency to develop decentralized applications and collaborative organizations to remove the need for a trusted third party.

In the followings, to understand blockchain-based cryptocurrency system, we briefly give a general overview of the Bitcoin on which our proposed incentive scheme is built.

2.1. Bitcoin System. Bitcoin is the first cryptocurrency which is a new form of a decentralized electronic cash system introduced by Satoshi Nakamoto [26]. Unlike traditional currency systems relying on a central authority such as a bank, Bitcoin is based on Peer-to-Peer (P2P) network and distributed consensus protocol without a trusted third party.

In Bitcoin system, each user has a private and public key pair to sign the transactions for coin transfers, and the address to uniquely identify a user is represented by a cryptographic hash of the public key for the respective user. The address is associated with user's account and the private key is used to sign transactions for spending coins.

Bitcoin payments are processed by generating transactions which transfer the values of coins from one user's account to another. Transactions are composed by senders and distributed to the Bitcoin P2P network, then the validity of the transactions is verified by Bitcoin network nodes called *miners*. After validating the transactions pending for a given time period, miners collect the transactions into a single unit called *block*. The new block accepted by the miners according to a consensus protocol is then added to the Bitcoin public ledger called *blockchain*.

2.2. Transaction. Bitcoin transaction is the record implying that transfers the value of coins from a sender to a recipient as shown in Figure 2. A transaction (TX) has a unique identifier and consists of a set of inputs and outputs which are key components of the transaction. Each input specifies unspent coins, belonging to a particular user, of the previous transaction identified by its hash code. Each output represents the amount transferred to the specified address of a recipient. To authorize spending the coins in the transaction input, the user should present his/her signature for the transaction and corresponding public key. Bitcoin uses ECDSA as a digital signature scheme.

```
scriptPubKey: OP_DUP OP_HASH160 <PubKeyHash> OP_EQUALVERIFY OP_CHECKSIG
```

To authorize spending the output, the corresponding input specifies an unlocking script of the form:

```
scriptSig: <Signature> <PubKey>.
```

Another interesting transaction to us is MultiSig transaction which requires multiple signatures to unlock the

```
scriptPubKey: 2 <PubKey A> <PubKey B> <PubKey C> 3 OP_CHECKMULTISIG
```

can be redeemable by including any combination of two signatures of the private keys corresponding to the three listed public keys as follows:

```
scriptSig: OP_0 <Signature A> <Signature C>, or
scriptSig: OP_0 <Signature B> <Signature C>
```

2.4. Time-Locked Transaction. Bitcoin supports both transaction-level and script-level time-lock features which restrict the spending of outputs of the time-locked transactions by a certain time in the future. The functions of time-locks are useful for postdating transactions and withholding redemption of funds to a date in the future. We are interested in script-level time-locks.

Unlike traditional bank services in which transactions are verified and maintained by a central bank, transaction verification in Bitcoin system is distributed to P2P network nodes and only the valid transactions are recorded in the distributed ledger by means of a blockchain.

The concept of processing Bitcoin transactions, which spends outputs of previous transactions, is to manage transaction chain which transfers coin ownership from a sender to a recipient [27]. Therefore, sending some Bitcoin is creating an unspent transaction output (UTXO) locked to a specific public key owner, and a new transaction can consume one or more UTXOs as transaction inputs.

2.3. Script. Each UTXO has to be bound to a specific user eligible for spending the coins in it. Bitcoin system introduces a scripting language to describe the essential conditions (encumbrances) to claim the coins. That is, each transaction output can contain a locking script which defines the conditions that must be met to spend the coins associated with the UTXO. One dominant script supported by today's Bitcoin system is Pay-to-Public-Key-Hash (P2PKH) which encumbers the output with a public key hash known as address. An output locked by a P2PKH script can be unlocked by the user who can present a public key and a signature generated by the corresponding private key. A P2PKH transaction output would have the following script of form (we omit detailed operation of each script command):

encumbrance. MultiSig transaction outputs are usually denoted as M-of-N, where N is the total number of public keys and M is the minimum number of signatures required for redeeming the transaction output. For example, 2-of-3 MultiSig transaction of the following script

There are two types of time-locks in the Bitcoin system: one is absolute time-lock and the other is relative time-lock. Absolute time-lock transactions use CHECKLOCKTIMEVERIFY opcode to specify a fixed date in the future when the output of the transaction can be spent, and relative time-lock transactions use CHECKSEQUENCEVERIFY opcode to establish amount of time far from the transaction publishing time.

2.5. Blockchain. Blockchain is a linked-list type data structure which maintains entire transaction history in terms of blocks. Figure 3 shows the blockchain structure used in the Bitcoin. When a transaction is generated and distributed to the Bitcoin network, some node called miner in the network

FIGURE 3: Blockchain structure in the Bitcoin system.

FIGURE 4: System architecture for Bitcoin-based incentives for VDTNs.

receives a confirmation from the destination point, the vehicle can spend the coins.

For such a scenario, in this paper, we focus on how to design Bitcoin-based secure incentive scheme for VDTNs by taking the following security goals into account.

- (i) *Fairness*: actual incentives must be rewarded to the volunteer vehicles only if the vehicle completes to transfer a message bundle to the destination point, which is fair to the source server to prevent a sort of dine and dash.
- (ii) *Authorization*: in cyberphysical vehicular networks, only the entities authorized by SM must be able to take part in the vehicular communications for preventing illegal entities from misusing the system.
- (iii) *Anonymity of vehicles*: the identity of the vehicle participating in store-carry-forwarding should be hidden

during vehicular communications and incentive protocol run for privacy preservation.

The proposed system achieves the security goals by leveraging the functionalities of Bitcoin transactions and adopts certified Bitcoin public key technique to authorize vehicles in the system. Table 2 describes the notations used in the proposed system.

3.2. Certified Bitcoin Public Key. For providing Bitcoin-based incentives, we assume that each roadside server RS_i and each vehicle v_i participated in message dissemination service on VDTNs has certified Bitcoin public key [18] issued by the SM. That is, RS_i and v_i have their Bitcoin private and public key pair $\langle bsk_{S_i}, bpk_{S_i} \rangle$ and $\langle bsk_{v_i}, bpk_{v_i} \rangle$, respectively. More specifically, let $\langle p, q, G, P \in \mathbb{G} \rangle$ be the public parameters where p and q are two large prime numbers, $\mathbb{G}$ is an additive group with the order q consisting of all points on an elliptic

TABLE 2: Notations and descriptions.

notation	description
$\langle x, X \rangle$	master secret and public key pair of SM
$\langle bsk_i, bpk_i \rangle$	private and public key pair of entity i
C_i	certified public key of entity i
TX	Bitcoin transaction
$TX.in$	input of the transaction TX
$TX.out$	output of the transaction TX
$Sig(sk, M)$	signature for a message M under a private key sk
$Vrf(pk, \sigma)$	verification for a signature σ under a public key pk
$\rho : \mathbb{G} \rightarrow \mathbb{Z}_q$	a mapping function
ts	timestamp

curve $E(\mathbb{F}_p)$, and P is a base point of $\mathbb{G}$. We can choose the parameters in accordance with secp256k1 ECDSA curve specification used in Bitcoin.

Let $\langle x, X \rangle$ be the secret and public key pair of SM where $x \in \mathbb{Z}_q$ and $X = x \cdot P$. Certified public keys for each entity U_i authorized by SM are generated as follows:

- (1) U_i chooses a random $k_i \in \mathbb{Z}_q$ and computes $K_i = k_i \cdot P$, then requests its certified public key by sending K_i to SM.
- (2) Assuming that the legitimacy of U_i was validated, SM chooses a random $r_i \in \mathbb{Z}_q$, computes $C_i = K_i + r_i \cdot P$ and $y'_i = r_i + \rho(C_i) \cdot x \pmod{q}$ where ρ is a function encoding an element of $\mathbb{G}$ as a positive integer. Then SM provides $\langle C_i, y'_i \rangle$ to U_i .
- (3) U_i computes $y_i = y'_i + k_i \pmod{q}$, $Y_i = y_i \cdot P$ and checks if $Y_i \stackrel{?}{=} C_i + \rho(C_i) \cdot X$. If it holds, U_i sets $\langle bsk_i \leftarrow y_i, bpk_i \leftarrow Y_i \rangle$ as its ECDSA key pair and C_i as certified public key.

When U_i 's certified public key C_i is given, we can derive the public key Y_i from C_i by using SM's public key X as $Y_i = C_i + \rho(C_i) \cdot X$ so as to verify the signature generated under y_i . In [18], the certified public key itself is encoded in the Bitcoin transaction output in the form of scriptPubKey script to designate the recipient. On the other hand, in our proposed system, the certified public key C_i is only used in authentication between a vehicle and a roadside server, and then the public key Y_i derived from C_i is encoded in the Bitcoin transaction rewarded as incentives.

3.3. Bitcoin-Based Incentive. In this section, we describe Bitcoin-based incentive scheme to reward a vehicle for the effort of message forwarding. Suppose that a source roadside server RS_s wants to send a message M to a destination point RS_d by means of store-carry-forward with the help of a vehicle v_i . When RS_s requests v_i to deliver a message to the destination point RS_d , the incentive transaction TX of RS_s is published to the Bitcoin network under the condition that the output of TX can be redeemed by v_i if v_i completes the message forwarding to RS_d by using MultiSig transaction. However, if v_i does not forward the message to the destination, RS_s is likely to lose its coins without taking advantage

of message delivery service from v_i because the RS_s 's input of TX is treated as spent in the Bitcoin system once TX is published. To cope with this situation, we put time-locked condition together with MultiSig so as for RS_s to withdraw the coins from TX if v_i does not forward the message nor redeem the output before the time-lock expires.

The details of the proposed scheme are described in the following:

- (1) A source roadside server RS_s broadcasts a request message including the identity of the destination point RS_d and the location information loc_d to ask for a volunteer vehicle which will help carrying a message to RS_d .
- (2) A vehicle v_i , which will pass by RS_d 's location and be willing to help message forwarding, responds to RS_s by giving $\sigma = Sig(bsk_{v_i}, RS_d || loc_d)$ with its certified public key C_{v_i} .
- (3) RS_s verifies the signature σ as $Vrf(bpk_{v_i}, \sigma)$ by deriving v_i 's public key bpk_{v_i} from C_{v_i} as described in Section 3.2. If the signature is valid, RS_s prepares a Bitcoin transaction TX_1 and composes a message bundle $msg_1 := \{M || ts || Sig(bsk_{RS_s}, M || ts), C_{RS_s}, TX_1\}$. Figure 5 shows the transactions for transferring Bitcoin as incentives. $TX_1.in$ specifies unspent coins retrieved from RS_s 's UTXO pool and it includes RS_s 's signature for the transaction. The amount of coins given to v_i as incentives is recorded in $TX_1.out$ and the redemption condition for $TX_1.out$ is written by using locking script consisting of 2-of-2 MultiSig for v_i and time-lock constraint for RS_d . Implementation details of the locking and unlocking scripts are presented in the next sections. RS_s publishes the TX_1 to the Bitcoin network and provides msg_1 to v_i . Note, at this phase, that TX_1 does not mean a prompt coin transfer but functions as a deposit which will be spent by someone who satisfies the unlocking condition.
- (4) Upon receiving msg_1 , v_i derives RS_s 's public key from C_{RS_s} and verifies RS_s 's signature. Then, v_i stores and carries the message bundle to the destination point RS_d . In addition, v_i checks the validity of TX_1 through the Bitcoin network and partially prepares a transaction TX_2 to redeem the coins specified in $TX_1.out$ while moving to the destination. At this phase, v_i fills in all other forms of TX_2 except MultiSig unlocking script for $TX_2.in$.
- (5) If v_i is an honest volunteer vehicle, v_i will faithfully store, carry, and forward the message bundle. Hence, when v_i arrives at the destination location and recognizes RS_d , v_i composes the message $msg_2 := \{M || ts || Sig(bsk_{RS_s}, M || ts), C_{RS_s}, TX_2\}$ and sends to RS_d .
- (6) RS_d parses the msg_2 and verifies the signature of $Sig(bsk_{RS_s}, M || ts)$ by using C_{RS_s} , then accepts the message M if the signature is valid. Then, as a witness to the effort of message delivery of v_i , RS_d generates a partial signature for TX_2 to unlock 2-of-2 MultiSig

```

scriptPubKey:  OP_IF
                2 < bpkvi > < bpkRSd > 2 OP_CHECKMULTISIG
                OP_ELSE
                < time-lock > OP_CHECKSEQUENCEVERIFY OP_DROP
                < bpkRSs > OP_CHECKSIGVERIFY
                OP_ENDIF

```

ALGORITHM 1: Locking script of $TX_1.out$ for giving incentives.

FIGURE 5: Transactions to give incentive or withdraw.

script which is needed for v_i to spend the coins specified in the previous transaction $TX_1.out$. RS_d provides $\{TX_2, Sig(bsk_{RS_d}, TX_2), C_{RS_d}\}$ to v_i .

- (7) v_i derives RS_d 's public key from C_{RS_d} and verifies the signature. If it holds, v_i completes 2-of-2 MultiSig unlocking script by adding v_i signature for TX_2 and finally publish TX_2 to the Bitcoin network to transfer the incentive given by TX_1 to v_i 's another Bitcoin account.

If all the above steps are correctly processed, the transactions TX_1 and TX_2 will be validated over the Bitcoin network and successfully appended to the blockchain, then v_i can gain Bitcoin incentives as a reward for its contribution to message delivery on VDTNs. In other words, v_i will not be rewarded if it ceases from forwarding the message even though TX_1 is published to the Bitcoin network in step 3 because v_i alone cannot fulfill 2-of-2 MultiSig locking script. Therefore, when RS_s finds that TX_1 is not redeemed by v_i after the time-lock expired, RS_s withdraws the coins by publishing the transaction TX'_2 as regarding that v_i did not forward the message faithfully.

3.4. Implementing Transaction Scripts. Validating a Bitcoin transaction relies on two types of scripts, a locking script and an unlocking script. For guaranteeing the fairness to the source server RS_s in our incentive scheme, we make

use of MultiSig script and time-lock script in the Bitcoin transactions. Hence, when the source server RS_s publishes the transaction TX_1 , RS_s can specify the proper recipient for $TX_1.out$ (i.e., v_i for incentive or RS_s for withdraw) by using MultiSig and time-lock script as shown in Algorithm 1. In our implementation, we consider relative time-lock which implies that $TX_1.out$ can be spent after the specified time has elapsed starting from the TX_1 publishing time. For example, RS_s can specify one day amount of time in $< time - lock >$ if RS_s allows for v_i to deliver the message within a day so that RS_s does not withdraw the coins for the day.

Once RS_s published the TX_1 to the Bitcoin network for providing incentives to v_i , the redemption condition for v_i to spend the coins of $TX_1.out$ is constrained under 2-of-2 MultiSig script fulfilled by both v_i 's and destination point RS_d 's signatures. Hence, if v_i correctly delivers the message requested by RS_s to the destination point RS_d and receives a partial signature for TX_2 from RS_d , v_i can spend the coins of TX_1 by completing the unlocking script of $TX_2.in$ as shown in Algorithm 2.

The Bitcoin transaction script is a stack-based execution language which uses a stack data structure to store input parameters and a return value of each operation. To execute an operation, the arguments for the operation are first pushed onto a stack and its calculation is performed by reading these arguments directly from the stack. The unlocking script contained in the output and the locking script in the

FIGURE 6: Combining unlocking script and locking script to evaluate a transaction script.

scriptSig: $OP_0 < Sig_{v_i} > < Sig_{RS_d} > TRUE$

ALGORITHM 2: Unlocking script in $TX_2.in$ for v_i to redeem the incentive.

scriptSig: $< Sig_{RS_s} > FALSE$

ALGORITHM 3: Unlocking script in $TX'_2.in$ for RS_s to withdraw the coins.

referenced input are combined as shown in Figure 6, and the combined script is executed from left to right in sequence by every Bitcoin validating node.

By combining those scripts of $TX_2.in$ and $TX_1.out$, the execution path of the first IF clause is selected by putting TRUE, then it would be evaluated in the script execution stack as the following form:

$OP_0 < Sig_{v_i} > < Sig_{RS_d} > 2 < bpk_{v_i} > < bpk_{RS_d} > 2OP_CHECKMULTISIG$

Figure 7 shows the stack-based script execution to validate v_i 's redemption condition by using MultiSig operation.

On the other hand, in order to withdraw the coins of TX_1 after the time-lock expired, RS_s publishes the transaction TX_2 containing the unlocking script as shown in Algorithm 3.

Like the preceding, by putting FALSE at the end of unlocking script, the execution path of ELSE clause is selected, but this execution is only used after the amount of $< time - lock >$ has elapsed from the creation of TX_1 . If so, RS_d 's script would be evaluated in the script execution stack as the following form:

$< Sig_{RS_s} > < bpk_{RS_s} > OP_CHECKSIGVERIFY$

Figure 8 shows the stack-based script execution to validate RS_s 's redemption condition by using time-lock restriction.

4. Discussion

4.1. Security of the Proposed System. As presented so far, our incentive scheme for VDTNs is designed by making use of Bitcoin system which is a cryptographically secure and practical decentralized virtual currency system. In this section, we discuss the security properties of the proposed system in terms of fairness, authorization, and anonymity of vehicular communications.

4.1.1. Fairness. When we design an incentive scheme based on virtual currency for VDTN environments in this paper, one of the important issues is fairness to the source server because a malicious vehicle might not follow the protocol run if the source server provides incentives first.

In the proposed system, providing incentives to a vehicle contributed to message forwarding is processed by the Bitcoin transaction TX_1 which conceptually transfers coins from the source server RS_s 's Bitcoin account ($TX_1.in$) to the forwarding vehicle v_i 's account ($TX_1.out$). Since the $TX_1.out$ for v_i is locked by 2-of-2 MultiSig script when RS_s publishes TX_1 to the Bitcoin network, the coin amount specified in $TX_1.out$ is ineffective for v_i to redeem it by $TX_2.in$ at this moment unless the destination point RS_d confirms the message receiving by giving its signature for TX_2 to unlock 2-of-2 MultiSig combined with v_i 's signature. v_i cannot but complete message forwarding to RS_d in order to redeem the incentive. Therefore, the source server RS_s does not need to worry about dine and dash of v_i , and v_i also can be rewarded for its contribution to message delivery if v_i honestly follows the protocol run.

Moreover, as considering the problematic situation where v_i ceases delivering the message to the destination once RS_s published TX_1 incentive transaction, RS_s is allowed to make the transaction TX'_2 to withdraw the coins from TX_1 by putting time-locked script. When RS_s finds that TX_1 is not redeemed by v_i after the time-lock expired, it is regarded that v_i did not follow the protocol and could not redeem the incentive, so RS_s withdraws the coins. Such time-lock condition also provides another feature that RS_s cannot withdraw the coins earlier than the time-lock. As an example for 12-hour time-lock, v_i can acquire the coins of TX_1 if v_i

	[script]
	IF $2 < bpk_{v_i} > < bpk_{RS_d} > 2$ CHECKMULTISIG ELSE ENDIF
TRUE	- Data constants of unlocking script are pushed to the stack.
Sig_{RS_d}	- IF operator pops the top item out of the stack. That is TRUE.
Sig_{v_i}	- Executes the code block up to matching ELSE.
	[script]
	CHECKMULTISIG
2	- Values $(2, bpk_{v_i}, bpk_{RS_d})$ are pushed to the stack.
bpk_{RS_d}	- CHECKMULTISIG operator pops the top item (value 2), and pops 2 public keys out of the stack. Then, it pops the quorum 2 and the final 2 items (signatures).
bpk_{v_i}	- Verify the signatures with the listed public keys.
2	
Sig_{RS_d}	
Sig_{v_i}	

FIGURE 7: Script execution for validating v_i 's redemption condition.

	[script]
	IF ELSE $< time - lock >$ CHECKSEQUENCE ENDIF
FALSE	- IF operator pops the top item out of the stack. That is FALSE.
Sig_{RS_s}	- Skip and execute the code block from ELSE to ENDIF.
	[script]
	CHECKSEQUENCEVERIFY DROP $< bpk_{RS_s} >$ CHECKSIG
$time - lock$	- $< time - lock >$ value is pushed to the stack.
Sig_{RS_s}	- CHECKSEQUENCEVERIFY operator pops the top item and checks whether the time-lock was elapsed or not.
	[script]
	CHECKSIG
bpk_{RS_s}	- bpk_{RS_s} is pushed to the stack.
Sig_{RS_s}	- CHECKSIG operator checks that the signature Sig_{RS_d} matches the public key bpk_{RS_s} .

FIGURE 8: Script execution for validating RS_s 's redemption condition.

arrives at the destination point within 12 hours while RS_s cannot withdraw the coins, which guarantees kind of fairness to the vehicle.

4.1.2. Authorization. To deploy a practical VDTNs application of good quality of service in the real-world scenarios, it is needed to allow only authenticated users to take part in the system. We consider using Bitcoin public key cryptography based on ECDSA for our VDTN scenario instead of adopting vehicular-PKI for authenticated vehicular communications. However, ordinary Bitcoin public keys are not sufficient to incorporate trustworthiness from real-world entities into the system. Hence, in the proposed system, each vehicle v and each roadside unit RS are authenticated by using certified Bitcoin public keys (C_v and C_{RS} , respectively) issued by SM based on the technique of [18].

When a vehicle and a roadside unit communicate to forward a message and give incentives, they exchange their certified public keys, then the corresponding Bitcoin public keys are derived from SM's public key. Because the exchanged message and Bitcoin transactions include signatures verified

under the derived public keys, any other entities unauthorized by SM cannot join the system.

4.1.3. Anonymity of Vehicles. For secure vehicular communications, another security aspect is anonymity of vehicles which voluntarily take part in message store-carry-forwarding communications on VDTNs. The only thing required to the vehicles in the system is their valid Bitcoin public key to uniquely identify the vehicle and handle Bitcoin transactions for incentives. Those public keys used in vehicle to roadside unit communications can be viewed as vehicle's pseudonyms and do not contain any identity information of the vehicle even though the proposed scheme employs certified public keys.

When we say privacy preservation, we should consider not only anonymity of user identity but also unlinkability. However, the proposed system does not fully satisfy unlinkability requirement because we just assumed that each vehicle has a single public key, so that any outside observer can decide any two messages or two transactions originated from the same user by tracing the fixed same public key of the user.

TABLE 3: Comparison of the characteristics of the proposed system with the previous systems.

	Lee et al. [3]	Zhu et al. [15]	Lu et al. [16]	proposed
network service	simple broadcast	dtn	dtn	vdtm
trust model	centralized authority	centralized virtual bank (VB)	centralized (VB)	distributed (blockchain)
fault tolerance	weak	weak	weak	strong
fairness	escrow held by the authority	escrow held by the VB	escrow held by the VB	multisig and timelock script on a blockchain
incentive means	virtual coin, application dependent	virtual coin, application dependent	reputation, virtual coin, application dependent	Bitcoin, worldwide
payment contract	signed receipt by a receiver	layered credits by sender, forwarder, receiver in sequence	layered credits	Bitcoin transaction script
rewarding	exchange receipt to virtual cash through the authority	credit clearance through the VB	credit clearance through the VB	Bitcoin payment
identity	certificate	certificate	certificate	Bitcoin public key (address)
anonymity	low	n/a	n/a	medium

One simple solution to this challenge is for a vehicle to generate multiple Bitcoin public keys and use a different public key whenever a message forwarding and incentive protocol is performed on VDTNs. However, in this case, it is required to securely maintain lots of private keys as many as the number of public keys. Another flexible solution is to use a one-time public key technique such as CryptoNote in which an ephemeral key pair for each protocol run can be computed from a fixed long-term key pair[29]. Hence, the proposed scheme can be modified by using one-time public key technique to enhance the anonymity of vehicular communications and incentive transactions.

4.2. Comparison. To highlight the novelty of the proposed Bitcoin-based incentive system for VTDNs, we evaluate and qualitatively compare our system with previous systems. Table 3 summarizes the different characteristics of the proposed system as compared with Lee et al.'s [3], Zhu et al.'s [15], and Lu et al.'s [16], respectively.

The key difference of the proposed system results from the use of Bitcoin which is a decentralized cryptocurrency and a worldwide payment system whose transactions are verified by means of a blockchain, while each previous system implements its own application-dependent virtual coin relying on a centralized trusted authority or a bank to guarantee the validity of payment transactions. Hence, for the previous system, we cannot help but depend on the central authority to enjoy reliable payment service. However, it is widely believed that the distributed structure of blockchain network performs better robustness under the single point of failure, so the proposed system can provide strong fault tolerance.

In addition, the previous systems make use of public key certificate to identify the entities participating in the network service and to verify the layered credits, but they do not focus on the anonymity of users. On the other hand, in our system, the Bitcoin public key used in a payment contract as the form

of Bitcoin transaction script can be viewed as a pseudonym and we can generate multiple keys or adopt one-time public key technique to enhance the anonymity to some degree.

We also compare the incentive procedures of the proposed system to reward a node for message forwarding and briefly shows the processes assigned to each entity relating to incentive scheme in Table 4. In the previous systems of [15, 16], the incentive for message forwarding is endorsed by means of the layered credits which are sequentially generated by message sender, forwarder, and receiver during the message delivery phase. To securely handle incentive scheme, forwarder and receiver must check the validity of the given credit by themselves and add their own credit layers in sequence. Then, the VB verifies the collected credits and records amount of virtual coin in forwarder's account if the credits are valid.

On the other hand, in our system, the incentive is handled by means of Bitcoin transactions to pay the coin from the sender to the forwarder. Those Bitcoin transactions are validated by Bitcoin network in a distributed manner and added to a blockchain which serves as immutable distributed ledgers. The message forwarder just queries the validity of sender's payment transaction to Bitcoin network, instead of verifying sender's payment transaction by forwarder itself. Message sender can control that the payment would be redeemed by the honest forwarder which delivers the message to the receiver by putting MultiSig locking script to the payment transaction which must be resolved by both forwarder's and receiver's signatures.

As compared to the previous system, the processes of validating incentive transactions to reliably pay the coins from the sender to the forwarder as an incentive are not burdened to VANET but shifted to Bitcoin network. The required processes for the sender and the forwarder are just to publish Bitcoin transactions which will be validated through a blockchain network. Therefore, we can develop a practical credit-based incentive scheme on VANETs at a low

TABLE 4: Comparison of the incentive procedures to perform virtual coin rewarding.

	sender	VANET		VB / Blockchain
		forwarder	receiver	
Zhu et al. & Lu et al.	(1) generate a base layer credit	(2) check the validity of the base layer credit (3) generate and add its endorsed layer credit	(4) check the validity of the layered credits (5) generate and add its endorsed layer credit (6) send and request clearance of the credits to VB	(7) check the validity of the all layered credits (8) check sender's balance (9) record amount of coin in forwarder's account
proposed	(1) publish incentive transaction to blockchain network (7) publish withdraw transaction to blockchain network	(4) query the validity of the incentive transaction to Bitcoin network (7) publish redemption transaction to blockchain network	(6) sign forwarder's redemption transaction	(2), (8) validate the incoming transaction (5) respond to the transaction query (3), (9) add the transaction to a blockchain

cost by removing the necessary of implementing application-dependent virtual coin system but by taking advantage of the functionalities of the existing cryptocurrency system.

5. Conclusion

In this paper, we proposed a secure incentive scheme incorporating with Bitcoin for VDTNs to stimulate vehicles positively cooperating with other nodes and to reward their efforts. Based on the security features of the Bitcoin system, the incentives for volunteer vehicles are rewarded by means of Bitcoins which can be worldwide used as virtual cash, and the fairness to the source server is guaranteed by using MultiSig transaction so that a message relaying vehicle can redeem the coins of incentive transactions only if the vehicle correctly completes the message relaying to a destination. To achieve the fairness goal, we also implemented transaction scripts to deal with reliable incentive rewarding based on locking and unlocking scripts consisting of 2-of-2 MultiSig and time-lock condition. Especially, as compared to the previous systems, the proposed incentive scheme can be developed at a low cost because we do not need to implement our own virtual currency system on VDTNs.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea Government (MSIT) (2017-0-00156, The Development of a Secure Framework and Evaluation Method for Blockchain).

References

- [1] J. Zhou, X. Dong, Z. Cao, and A. V. Vasilakos, "Secure and privacy preserving protocol for cloud-based vehicular DTNs," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 6, pp. 1299–1314, 2015.
- [2] J. A. F. F. Dias, J. J. P. C. Rodrigues, and L. Zhou, "Cooperation advances on vehicular communications: a survey," *Vehicular Communications*, vol. 1, no. 1, pp. 22–32, 2014.
- [3] S.-B. Lee, J.-S. Park, M. Gerla, and S. Lu, "Secure incentives for commercial ad dissemination in vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 6, pp. 2715–2728, 2012.
- [4] X. Lin, R. Lu, and X. Shen, "Location-Release Signature for Vehicular Communications," in *Proceedings of the 2009 Proceedings of 18th International Conference on Computer Communications and Networks - ICCCN 2009*, pp. 1–7, San Francisco, CA, USA, August 2009.
- [5] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *Journal of Computer Security*, vol. 15, no. 1, pp. 39–68, 2007.
- [6] X. Lin, X. Sun, P.-H. Ho, and X. Shen, "GSIS: a secure and privacy-preserving protocol for vehicular communications," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 6 I, pp. 3442–3456, 2007.
- [7] R. Lu, X. Lin, H. Zhu, P.-H. Ho, and X. Shen, "ECPP: efficient conditional privacy preservation protocol for secure vehicular communications," in *Proceedings of the 27th IEEE Communications Society Conference on Computer Communications (INFOCOM '08)*, pp. 1229–1237, April 2008.
- [8] R. Lu, X. Lin, H. Zhu, P.-H. Ho, and X. Shen, "A novel anonymous mutual authentication protocol with provable link-layer location privacy," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 3, pp. 1454–1466, 2009.
- [9] Y. Park, C. Sur, C. D. Jung, and K.-H. Rhee, "An efficient anonymous authentication protocol for secure vehicular communications," *Journal of Information Science and Engineering*, vol. 26, no. 3, pp. 785–800, 2010.
- [10] Y. Sun, R. Lu, X. Lin, X. Shen, and J. Su, "An efficient pseudonymous authentication scheme with strong privacy preservation for vehicular communications," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 7, pp. 3589–3603, 2010.

- [11] Y. Park, C. Sur, S. Shin, K.-H. Rhee, and C. Seo, "A privacy preserving message delivery protocol using identity-hidden index in VDTNs," *Journal of Universal Computer Science*, vol. 19, no. 16, pp. 2385–2403, 2013.
- [12] X. Zhu, S. Jiang, L. Wang, and H. Li, "Efficient privacy-preserving authentication for vehicular Ad Hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 2, pp. 907–919, 2014.
- [13] Q. Li, A. Malip, K. M. Martin, S.-L. Ng, and J. Zhang, "A reputation-based announcement scheme for VANETs," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 9, pp. 4095–4108, 2012.
- [14] J. A. F. F. Dias, J. J. P. C. Rodrigues, L. Shu, and S. Ullah, "Performance evaluation of a cooperative reputation system for vehicular delay-tolerant networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2014, no. 1, pp. 1–13, 2014.
- [15] H. Zhu, X. Lin, R. Lu, Y. Fan, and X. Shen, "Smart: a secure multilayer credit-based incentive scheme for delay-tolerant networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 8, pp. 4628–4639, 2009.
- [16] R. Lu, X. Lin, H. Zhu, X. Shen, and B. Preiss, "Pi: a practical incentive protocol for delay tolerant networks," *IEEE Transactions on Wireless Communications*, vol. 9, no. 4, pp. 1483–1493, 2010.
- [17] J. Zhou and Z. Cao, "Secure incentive-based architecture for vehicular cloud," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM'12)*, IEEE, Anaheim, USA, December 2012.
- [18] G. Ateniese, A. Faonio, B. Magri, and B. de Medeiros, "Certified bitcoins," in *Proceedings of the International Conference on Applied Cryptography and Network Security*, vol. 8479 of *Lecture Notes in Computer Science LNCS*, pp. 80–96, Springer, 2014.
- [19] Y. Park, C. Sur, H. Kim, and K.-H. Rhee, "A reliable incentive scheme using bitcoin on cooperative vehicular ad hoc networks," in *Proceedings of the 2017 International Symposium on Mobile Internet Security (MobiSec'17)*, pp. 1–8, Jeju, Republic of Korea, October 2017.
- [20] "Bitcoin," <https://bitcoin.org>.
- [21] "Namecoin," <https://namecoin.org>.
- [22] "Litecoin," <https://litecoin.org>.
- [23] "Monero," <https://getmonero.org>.
- [24] "Ethereum," <https://ethereum.org>.
- [25] "Zcash," <https://z.cash>.
- [26] S. Nakamoto, *Bitcoin, A peer-to-peer electronic cash system*, 2008, <https://bitcoin.org/bitcoin.pdf>.
- [27] A. M. Antonopoulos, *Mastering Bitcoin - Programming the open blockchain*, O'Reilly Media, 2017.
- [28] J. B. Kenney, "Dedicated short-range communications (DSRC) standards in the United States," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1162–1182, 2011.
- [29] N. van Saberhagen, *Cryptonote v2.0*, 2013, <https://cryptonote.org/whitepaper.pdf>.