IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON SOCIAL COMPUTING APPLICATIONS FOR SMART CITIES

Received May 31, 2018, accepted July 1, 2018, date of publication July 23, 2018, date of current version August 20, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2858237

Secure and Efficient Large Content Broadcasting in Mobile Social Networks

TINGTING FU[1,*], PENG LIU[1,*], (Member, IEEE), YUE DING[1], AND YUAN ZHANG[2]

[1]Key Laboratory of Complex Systems Modeling and Simulation, National Demonstration Center for Experimental Computer Education, School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou 310018, China
[2]Center for Cyber Security, School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

Corresponding author: Yuan Zhang (zy_loye@126.com)

This work was supported in part by the Natural Science Foundation of China under Grant 61601157, in part by the Cross-Discipline Innovation Team Building Project of Hangzhou Dianzi University (Intelligent Decision Optimization and System Operating Security), in part by the National Demonstration Center for Experimental Computer Education, Hangzhou Dianzi University, and in part by the Chinese Scholarship Council under Grant 201208330096.

ABSTRACT Most of existing mobile social systems are built on the presence of communication infrastructures and thereby rely on the reliability of the underlying communication infrastructures. As a consequence, if the infrastructures are inaccessible to users, the data generated by users in these systems cannot be transmitted. Furthermore, existing mobile social systems bear a strong assumption that a data owner can determine the destinations before she/he transmits the data. This is not practical when the underlying communication infrastructures are compromised. This problem is further exacerbated by the fact that there are adversaries in the systems, where adversaries may attempt to extract the content of transmitted data and stop the data transmission. As such, the functionality and security of these systems cannot be guaranteed. In this paper, we propose a secure and efficient data broadcasting scheme for mobile social networks that enable data owners to broadcast large-size data in a store-and-forward way, which does not require reliable communication infrastructures. Even if the data owner cannot pre-determine the all destinations, the data owner can transmit the data to all destinations as soon as possible with the aid of data receivers. Our scheme ensures that the data transmission cannot be stopped by adversaries, and the honest user would extract the data no later than adversaries. In our scheme, if the adversary attempts to learn the data content, he has to assist the data owner in transmitting the data. We provide the security analysis to demonstrate the security of our scheme. We also implement our scheme and conduct a comprehensive performance analysis, which proves that our scheme is effective and efficient.

INDEX TERMS Mobile computing, social network services, large content, broadcasting, secret sharing.

I. INTRODUCTION

Mobile social network enables users to share their data to others with the aid of modern communication technologies. For example, users can directly exchange data, such as voice, photos, videos, and files, with each other by using smartphones via the Internet. Consider a demand for the content broadcasting in the mobile social network, where a content owner attempts to broadcast a large volume of data content to as many friendly users as possible. Although he may not know each of the friendly user, he can post the information on the Facebook, Wechat, or Blog. Any user who ''follows'' him could see the information and download the content. Encryption and authentication can be done through the APPs or the servers. Typical applications include a movie star interacting with her fans, a politician training his supporters, and a user sharing something with his friends.

However, it mainly relies on communication infrastructures, e.g., femtocells, base stations, switchers, servers and etc [1]. Once these infrastructures are destroyed or disabled, either due to nature disasters or adversarial attacking, people lose their direct and instant contacts to their friends, families, and colleagues. In that case, device-to-device (eg., phone-to-phone, vehicle-to-vehicle [2]) communication can be utilized to rebuild connections as it is one of the important means of message exchange in mobile social networks. Data will be passed from one device to another one within a short wireless communication range. It could also be the most practical method to maintain part of network functionality when the network infrastructures are compromised during the military crisis or destroyed by a catastrophe, where the messages are disseminated in a carry-and-forward manner. In such scenario, the information forwarding processes are facing

serious security and privacy challenges, since centered or third-party servers are not able to provide services such as encryption, authentication, and user information retrieval without networking environment. Even so, the smart city should support emergency communication requirements [3]. On the other hand, phone-to-phone based information forwarding is of infrastructure independence and less cost, so that it can also be a very useful auxiliary to the current 5G network structure.

There are some real events where the mobile social network showed its importance, such as the coup crisis. The president notified his supporters the current situation via Facebook since the traditional media had been compromised. But what if the communication infrastructures, such as the internet and the cellular network, were also unavailable at that time. How to achieve the goal of getting the information delivered to as many supporters as possible in a timely manner without the support of communication infrastructures while being intercepted by minimal malicious attackers, still remains as a very challenging problem. Pocket Switched Networks (PSNs) utilize human-carried mobile devices as the message senders and receivers. Human mobility and spatiotemporal connectivity play an important role in the exchange of messages. For example, routing protocols, such as [4] and [5], utilize the users mobilities to design forwarding algorithms so as to reduce latency and forwarding cost. On the other hand, the message delivery cost which is another major concern, is mainly affected by two factors, i.e., the transmission energy consumption of devices [6], [7] and the number of transmissions. Thus, the appropriate communication technologies such as WiFi [8] and Bluetooth [9], should be carefully selected according to the application scenario. When the data package itself is of very large volume and has multiple destinations, to avoid unnecessary forwardings is the top priority.

The assumption that the users in the network are friendly and willing to participate in relaying processes which requires their computing resources, memory space and energy, is a common consensus in most related work. To deal with the situation that users may not be so cooperative, incentive based schemes are used to stimulate users to participate in relaying [10]. However, the incentive based schemes are not very effective to malicious users since their main purpose is trying to eavesdrop, destroy, and stop forwarding the content. In those research work which mainly focus on the security side, malicious users are excluded from other users by using encryption methods. However, encryption and decryption themselves need spend a lot of extra computational energy, especially when the data volume is very large. In most cases, a trusted server is required to store keys, encrypt/decrypt the message, or cash a reward, which is not feasible during the failure of network infrastructure. In the proposed scenario, we observe that, compared with the broad spreading of the message, limited leaking of the content is acceptable. Furthermore, the delivery delay and ratio are the main metrics we should consider. Therefore, instead of identifying malicious users, is it possible to utilize them in the broadcasting process in a risky mobile social network?

The motivation of the paper is to develop an efficient broadcasting scheme in an infrastructure-free network, where malicious users and friendly users coexist in it. The objective of the source user, is to broadcast a large data chunk (such as a multimedia file) to as many desired friendly users as possible. The malicious users are trying to make trouble by filching the content of the file disseminated in the network and refusing to help forwarding. As mentioned in the above, there implies three major facts. First, there is no network infrastructure support (could be destroyed or disabled) so that not only normal users but also malicious users have to communicated with each other using carry-and-forward method. Second, the size of data is big enough so that it can be easily divided into small pieces. Third, the broadcast operation requires this content to be sent to all friendly users. To cope with the challenges of secure and efficient broadcasting in the risky mobile social network, in this paper, we design a decentralized scheme to slow down the forwarding process between malicious users and urge them to help broadcast the content effectively while remain security performance.

The main contributions of this paper are four-fold:

- We consider the secure-aware broadcasting of large contents in a risky mobile social network where communication infrastructures are not functional. The goal is to maximize the number of friendly users obtained the content and control the number of malicious users filched the content.
- A decentralized forwarding scheme is proposed to ensure secure and efficient broadcasting in phone-to-phone communication manner. It runs in a distributed manner and does not rely on any third part servers.
- Not like existing researches which focus on strictly preventing malicious users from obtaining anything, in our scheme, malicious users are motivated to help in the dissemination without necessity of identifying them.
- We evaluate our method using real data trace and analyz the performance in two different scenarios where malicious users act on their own, and tend to cooperate with each other respectively.

The rest of the paper is organized as follows: we discuss related work in Section II. Then Section III gives the problem formulation, threat model and design goals. After that, we illustrate the implementation details of our scheme in Section IV. We further perform security analysis in Section V. The effectiveness and improvement of our method, are analyzed with the experimental results in Section VI, based on the real data trace. Section VII summarizes this paper and provides suggestions for future perspective.

II. RELATED WORK

Data routing is one of the hot research topics in Mobile Opportunistic Networks (MONs). The most popular forwarder selection criterion is the data delivery latency and cost. During intermittent contacts between data holders and

potential forwarders, many algorithms are proposed to fulfill this goal, such as the contact history [11], contact quality [5], social-awareness [12], and so on. Such algorithms need historical information of users' movement and interactions. Some work further mine the inherent feature of social contact information, and propose data forwarding schemes such as multi path [13], UP-N-DOWN [14], and time sensitive [15]. Data delivering to multiple destinations are also considered in MONs [16], [17], which reflects the basic demand with the development of MON applications. Users are assumed to be cooperative in these works. However, in practice, many users are selfish and adversarial. Also, there is another topic which is not fully studied, i.e., the data content itself could be a very large file or chunk. In that case, the memory space in intermediate nodes will be another constraint for relaying.

Some studies have focused on the topic of promoting users' motivation and proposed some efficient algorithms, such as incentive and reputation based methods. For example, Pi [18] is one of such kind of algorithms which can ensure the fairness during the data forwarding. The relaying nodes will get credits when the packets are delivered. At the same time, they will also gain reputation for undelivered data which they relayed. The security of payment is another concern, the reward should be paid to the exact user with appropriate amount. RACE is proposed in [19]. The concept of evidence is used to secure the payment. To reduce overhead in clearing the payment, cryptographic operations are only required in case of cheating. The drawback of this method is that it needs to set up the end-to-end route before data transmission which is often not practical in Mobile Social Networks. Chen *et al.* [20] argue that current incentive based algorithms fail to further encourage nodes to follow defined rules so as to realize the desired performance objective, e.g., minimal average delay, maximal hit rate, minimal maximal delay, etc. They propose a game theory based incentive scheme, Multicent, which not only provides cooperative incentives but also is adjustable to different performance goals. However, apparently, incentive based strategies have little impact on malicious users. The security schemes adopted in incentive based methods only make sure no one can cheat in the transactions. In [21], both the incentive and security are enabled. The proposed method encourages edge nodes to cooperatively provide caching services with incentives as well as evaluates the reliability of the selected candidate of edge node by considering the direct trust evaluation.

Secure data forwarding is another major concern in Mobile Social Networks or Delay Tolerant Networks. A novel key generation mechanism based on information of relay nodes is proposed in [22]. It can secure routing pathes and prevent the data from wormhole and black hole attacks. In [23], the proposed secure geographic routing protocol PrivHab+ learns about the mobility habits of the nodes of the network and uses this information in a secure manner. It utilizes cryptographic techniques from secure multi-party computation to preserve nodes' privacy while taking routing decisions. The disadvantage is that, it works properly only when the extra information, such as mobility habits, is known. Guo *et al.* [24] tackle the privacy issues where the existing work needs user's personal information as one of the important factors in the data exchange scheme design. They propose PSaD, which establishes users' social relationship by matching their identical and verifiable attributes in a privacy-preserving way. Furthermore, the content is kept encrypted during the dissemination process, in order to let those who have the corresponding attributes decrypt the content. To provide routing properties in DTNs such as confidentiality of the nodes' routing metric, anonymous authentication, and efficient key agreement for pairwise communication, Zhang *et al.* [25] propose an advanced framework for opportunistic routing. They consider both security and privacy in this work. The content of the data and the information of the participated users are protected. However, both these works either need the support from the third part server, or just try to preserve the privacy information of the target nodes. We presented our preliminary work in [26] where a fragmentation and credit based method is proposed. The large content is divided into m data chunks together with the fragment of the key. Only when all m data chunks and key fragments are collected can the original content be decrypted. It is a big challenge to implement a secure broadcast for large contents without knowing the destinations. In the scenario, we assume that malicious users also have no knowledge of the identities of other users. This paper is an extension from our previous work. We improve theory foundation of the setup phase and also improve the data exchanging rule during the forwarding process. New experiments are conducted to test the performance.

III. PROBLEM FORMULATION

In the Mobile Social Networks, there are three kinds of users, namely, the content owner (which is also the source node, denoted by S), friendly users (denoted by the human icon), and malicious users (denoted by the devil icon), as shown in Fig. 1. The content owner is going to broadcast an important content file to all the friendly users. This is a non-trivial problem since the content owner itself cannot know each friendly user. The relationship between them, for example, like a rescue team and victims, a movie star and his fans, and a politician and his supporters. In this paper, we assume that friendly users do NOT recognize malicious users, neither do they know other friendly users. However, the malicious users may know each other, since the total number of them is relatively small and they tend to cooperate to do something bad. In some cases, the content owner may not have a lot of chances to encounter other users so that those who get the content file from him can be regarded as seeds. From these seeds, the content file can be copied and spread in the network. If there are malicious users in the group of seeds, the information may leak very fast and cannot reach many friendly users. Therefore, the broadcasting scheme should be robust enough to work against malicious users during the seeding process.

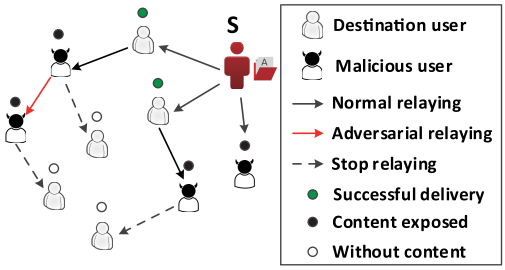

FIGURE 1. An illustration of the content broadcasting in a risky mobile social network.

In Fig. 1, the source firstly meets three users, two of which are friendly users and another one is a malicious user. Since the source does not know malicious users, he gives each of them the content file. Those friendly users having the content will continue to pass the content to all the users they meet since they cannot identify the malicious users as well. As a result, two more malicious users obtain the content. The malicious users definitely do not want to participate in data forwarding, so they just stop relaying the content upon encountering the friendly users (denoted by dashed line). However, if they encounter a malicious user they will continue to forward the content to him (denoted by red solid line). Therefore, finally four malicious users have the content while only two friendly users receive the content. Without any counterplan, we can expect that a lot of malicious users can filch the content while the number of friendly users successfully received the content will be less. The delivery latency will also be very considerable large.

A. THREAT MODEL

In the proposed scenario, the objectives of malicious and friendly users are summarized as shown in Table. 1. Malicious users attempt to filch the content and hold the content from relaying. To do so, they need hide themselves among users. During content exchanges, malicious users can hide contents that they own and only ask for what they do not have, since it is not required to exchange catalog before data. Furthermore, malicious users cannot be identified by the broadcasting requestor, and friendly users, since the attack performed by malicious users is passive, and such attack is hardly to be detected. However, part of malicious users may know each other because they tend to form a strict organization by which the attack is easier. As such, the collusion between malicious users must be considered in the threat model.

TABLE 1. Objects of adversarial and friendly users.

Malicious user	Friendly user
Hide itself	Identify malicious user
Filch the content	Broadcast the content
Modify the content	Protect the content
Offer minimum relay help	Try its best effort to help

Note that our primary objective is to enable as many as friendly users to receive the content sent by the broadcasting requestor in a timely manner. Therefore, there is no strict requirement and we do not guarantee the protection of the content against the malicious users.

B. DESIGN GOALS

In this paper, we target the secure and efficient content broadcasting in mobile social networks, in which there exist a challenge:

- How to ensure the content transmission in an untrusted environment. Since malicious users attempt to stop content transmission and are hardly to be detected, the content transmission should be ensured. As a malicious user may impersonate a friendly user to interact with the broadcasting requestor for the content, traditional method that splits the content into multiple chunks and each chunk is sent to each user cannot be directly adopted. In particular, to stop the content transmission, the malicious user only needs to delete the chunk he received, and other users would not collect all chunks to retrieve the content. Furthermore, to ensure the content should be sent to friendly users as soon as possible, an incentive mechanism that promotes the content transmission is highly expected.

To enable secure and efficient content broadcasting under the aforementioned model, the following objects should be achieved.

- Security. The malicious user cannot stop the content transmission by protecting the chunk(s) he has from transmission; The content should be extracted by the malicious user as slow as possible.
- Efficiency. The content should be sent to friendly users as soon as possible; The expected time that friendly users receive the content should be adjustable; The communication and computation overhead should be as efficient as possible.

IV. PROPOSED SOLUTION SEBMOSO

To urge the malicious users participating in relaying data and guarantee the security, we are going to analyze the behavior of them in the first place. First, they try to pretend to be friendly users because they want to hide themselves among users. Second, they try to fetch data from other users but are not willing to share theirs. Based on that, we propose our basic idea against these two characteristics. The scheme must ensure that there is little benefit if malicious users chose to hide what they really have in their buffer. The scheme will also force the malicious users share their data before they can ask for new data from other users. Consequently, the large multimedia content will be divided into several small data chunks, which is also accordance with the "smaller packet, faster delivery" principle. The malicious users need to share

some data chunks with other users so as to accumulate credits which is a key factor in future data exchange.

The detail is implemented as a method called **S**ecure and **E**fficient **B**roadcasting in **Mo**bile **So**cial Networks, *SEBMoSo*. It is composed of two stages, namely, **setup** and **forwarding**. Initially, a set of n users $\{\mathcal{U}_1, \ldots, \mathcal{U}_n\}$ are involved in the scheme to communicate with the source user. These users would forward the data received from the source user to other users.

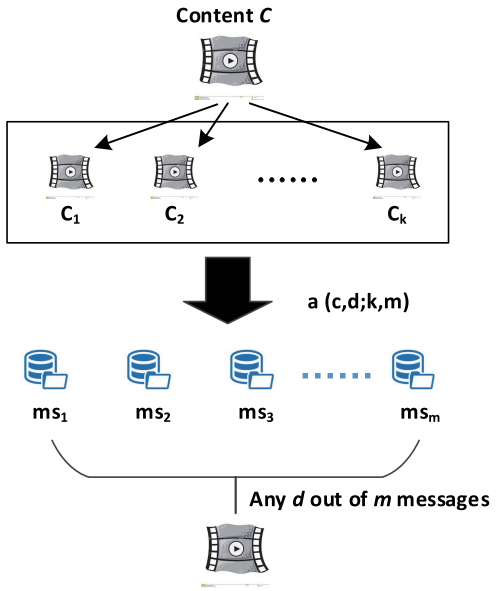

FIGURE 2. A demo to show the fragmentation of the content and secret sharing scheme.

As shown in Fig. 2, the first stage is to divide the content into k data chunks in an adjustable way. However, the size of k should be accordance with the efficiency of fragment based data broadcasting. For example, in [27], if the best number of fragments calculated by the algorithm is 6, then $k \geq 6$ holds. And we will discuss how to decide the size of k in later sections. The key technique used here is a $(c, d; k, m)$-multi-secret sharing scheme [28], where a dealer shares k secrets with n players such that any subset of at least d players can reconstruct all k secrets from their received messages and no subset of at most c players can deduce anything about the k secrets from their received messages. The message sent from the dealer to a player is a multi-share of the k secrets [28]. Our key observation is that the data generated by the source user can be split into k chunks, and each chunk can be considered as a secret that the source user shares with friendly users who will further send the received message to other user. If a friendly user collects d messages from d different users, she/he can reconstruct all k chunks and extract the data sent from the source user; a malicious user collects any subset of at lease d messages from d different users cannot deduce anything about the data. To control the number of messages injected into the network (tradeoff between overhead and security performance), the k data chunks will be further repacked into m messages where $m \geq k$. The size of m correlates to many factors such as the size of n, average encountering ratio, and so on. The source node can also determine the time that friendly users extract the data by setting the corresponding threshold number d. Generally, the size of d should follow $d \geq k$. Furthermore, even if the source user sends some multi-shares to some malicious users, i.e., parts of data generated by the source user would not be delivered at the **forwarding** stage, friendly users can eventually extract the data. In this way, both the data security and the delivery efficiency are ensured [29]. The selection of k, m, and d will be discussed in the experiment section.

Specifically, given the data C, the source user split it into k chunks as $C = \{C_1, \ldots, C_k\}$. Each first round user $\mathcal{U}_i (i \in [1, m])$ is associated with a public element $\alpha_i \in F$, where F is a finite field. The source user randomly chooses a polynomial $q(x)$ with degree $t - m$. Then the source user determines m elements $e_1, \ldots, e_m \in F$ such that $p(e_i) = C_i$, and calculates a multi-share $ms_i = p(\alpha_i)$ for $\mathcal{U}_i (i \in [1, m])$, where $p(x) \in F[x]$ is an otherwise random polynomial with degree t such that $p(e_i) = C_i$, and

$$p(x) = q(x) \prod_{i=1}^{m} (x - e_i) + \sum_{i=1}^{m} s_i \left(\frac{\prod_{j=1, j \neq i}^{m} (x - e_j)}{\prod_{j=1, j \neq i}^{m} (e_i - e_j)} \right).$$

The second stage is to forward messages. To prevent the malicious users from extracting data quickly, we design a credit based forwarding scheme. It is composed of two parts. The first part is for the source user as shown in Alg. 1. Let $\mathbb{D}(i)$ denote the set of multi-shares on user $\mathcal{U}_i$. Φ is the set of the entire multi-shares. To reduce the risk that a malicious user gets d multi-shares messages easily, the source user will only forward one multi-share to the user it meets upon encountering. Therefore, even if the data owner meet a few malicious users in the beginning, there are still chances that the original content can be extracted by friendly users. To avoid the situation that some users are repeated given messages by the source, we make the rule that each user can only get one multi-share message from the source no matter totaly how many times they meet.

Algorithm 1 Forwarding Strategy of Source

1: **FOR** each encountering with a user $\mathcal{U}_j$ **DO**
2: **IF** $\mathcal{U}_j$ is not met before **THEN**
3: Randomly forward a multi-share message $ms_i \in (\Phi - \mathbb{D}(j))$ to $\mathcal{U}_j$
4: **END IF**
5: **END FOR**

Below are the credit based rules for data exchanging among users. Users cannot forge credits and only earn credits during relaying.

A. RULE ONE: CREDIT OBTAINING RULE

Let $R(i)$ denote the credit obtained by a user $\rangle$. During the encountering between $\mathcal{U}_i$ and $\mathcal{U}_j$, if $\mathcal{U}_i$ forwards a multi-share to $\mathcal{U}_j$, then $\mathcal{U}_j$ will provide $\mathcal{U}_i$ a receipt, i.e., a signature on the multi-share. The signature algorithm used here can be the BLS signature [30] which enables multiple receipts to be verified simultaneously. Then, the number of credits of $\mathcal{U}_i$ will increase by 1. Since the credit is based on the signature of an individual user, say $\mathcal{U}_j$, no matter how many multi-shares have been offered to $\mathcal{U}_j$, its signature will be counted once. By this means, it is difficult for malicious users plot together to cheat credits.

B. RULE TWO: MULTI-SHARE MESSAGE EXCHANGE RULE

When two users meet, the user with more credits (denoted by $R(i), i \in [1, n]$) can ask the user with less credits $R(i)-R(j)+1$ messages the latter one owns while the latter one can only require one multi-share from the former one. This ensures that malicious users have to help in forwarding otherwise they cannot get the original content and may be exposed. If their credits are equal, then each one can only ask one multi-share message from the other user. Without loss of generality, suppose $R(i)$ is not less than $R(j)$. Let $ms_1, ms_2, \ldots, ms_m$ be all multi-shares. The process is shown in Alg. 2.

Algorithm 2 Forwarding Strategy Between Users

1: **FOR** each encountering between $\mathcal{U}_i$ and $\mathcal{U}_j$ **DO**
2: **IF** $R(i) > R(j)$ **THEN**
3: **FOR** count = $R(i) - R(j) + 1$ **DO**
4: randomly pick a $ms_\nu \in (\mathbb{D}(j) - (\mathbb{D}(j) \cap \mathbb{D}(i)))(\nu \in [1, m])$, forward ms_ν to $\mathcal{U}_i$
5: **END FOR**
6: **IF** $\mathcal{U}_j$ does not have the signature of $\mathcal{U}_i$ **THEN**
7: $R(j)++$
8: **END IF**
9: randomly pick a $ms = ms_\nu \in (\mathbb{D}(i) - (\mathbb{D}(i) \cap \mathbb{D}(j)))(\nu \in [1, m])$, forward ms to j, $R(i)++$
10: **ELSE IF** $R(i) = R(j)$ **THEN**
11: randomly pick a $ms = ms_\nu \in (\mathbb{D}(i) - (\mathbb{D}(i) \cap \mathbb{D}(j)))(\nu \in [1, m])$, a $ms' = ms_\nu \in (\mathbb{D}(j) - (\mathbb{D}(i) \cap \mathbb{D}(j)))(\nu \in [1, m])$, forward ms to $\mathcal{U}_j$, $R(j)++$ forward ms' to $\mathcal{U}_i$, $R(i)++$
12: **END IF**
13: **END FOR**

In the algorithm 2, as shown in line 2, the first step is to compare the credits of two encountering users. Then the user with higher credits will ask the other user for $R(i) - R(j) + 1$ multi-share messages that he dose not have. At the same time, the user with lower credits can request only one multi-share message. If the credits are equal, each one can ask for a data chunk from the other user.

After collecting d multi-shares, a user can interpolate these multi-shares to recover $p(x)$, compute $C_i = p(e_i)$ and retrieve $C = \{C_1, \ldots, C_k\}$.

V. SECURITY ANALYSIS

In this section, we discuss security issues of the proposed scheme. Note that the receipt is constructed on the BLS signature, we assume each user in the scheme only has one identity, and impersonating attacks have already been prevented by other security policy in the system.

- The proposed scheme ensures the data transmission under the threat model. Note that there are two objects for the malicious user. The first one is to enable friendly users to extract the source user's data as late as possible and another one is to extract the source user's data as soon as possible. To extract the data, the malicious user has to collect multi-shares from different users. In the proposed scheme, to collect multi-shares from other users, the malicious user needs to send the multi-shares he has to other users. As such, the data transmission is guaranteed. We want to further stress that the data transmission is guaranteed even if the source user transmits a multi-share to a malicious user in the **setup** stage, since only if a friendly user collects d multi-shares in the **forwarding** stage, she/he can retrieve the source user's data.
- The proposed scheme protects the data against malicious users. In the proposed scheme, a malicious user cannot extract the source user's data before he collects a threshold number (i.e., d) of multi-shares from different users. Generally, if the malicious user obtains a new multi-share from a user, he needs to send a multi-share he has but the user does not have to the user. If a malicious user extracts the source user's data, any user whose costs in terms of communication and computation overhead are more than those of the malicious user can extract the source user's data with a high probability.
- The proposed scheme is secure against credit forgery attacks. In the proposed scheme, the credit is built on the BLS signature scheme [30]. Since the BLS signature is existentially unforgeable, i.e., it is computationally infeasible to forge at least one valid message/signature pair for the malicious user, where the signature was not produced by the friendly user (legitimate signer). Therefore, the number of credits a user has cannot be forged in the proposed method. This ensures that the proposed scheme is secure against credit forgery attacks.
- The proposed scheme has better performance in the scenario where malicious users conspire to attack. Without credit based scheme, if the number of malicious users who initially get multi-share messages from the broadcast requestor is large or equal to $m - d$, then none of the friendly user will be able to obtain the original data content.

VI. EVALUATION AND RESULTS

We evaluate our *SEBMoSo* scheme using the data trace Infocom 06 derived from a real social event. In the trace, there are totaly 72 users in which one user is randomly selected as the source node. Malicious users and friendly users are also randomly selected and for each experiment this random selection is repeated for 100 times. The content is initially fragmented into three data chunks. Next, total 20 ($m = 20$) multi-share

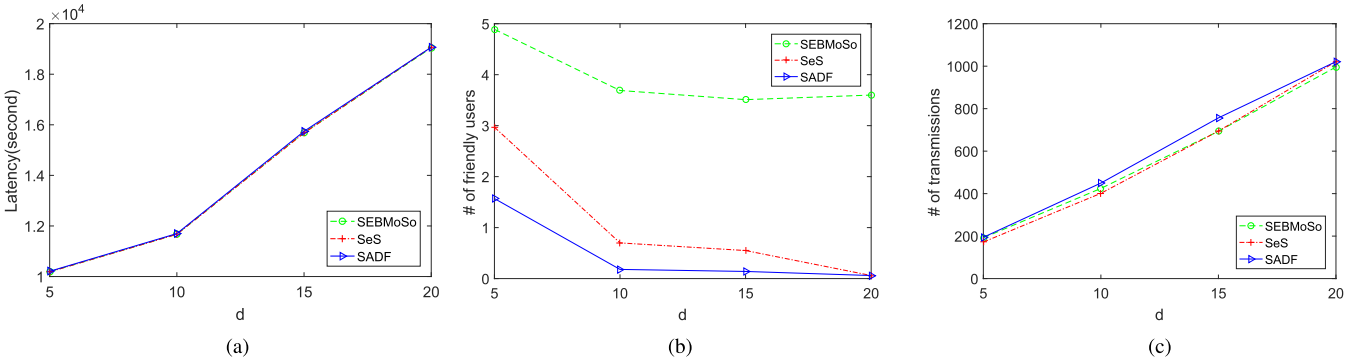

FIGURE 3. The performance evaluation upon any malicious user gets the original content under condition of 57 friendly users and 14 malicious users. (a) Time elapsed. (b) Number of delivered friendly users. (c) Number of transmissions.

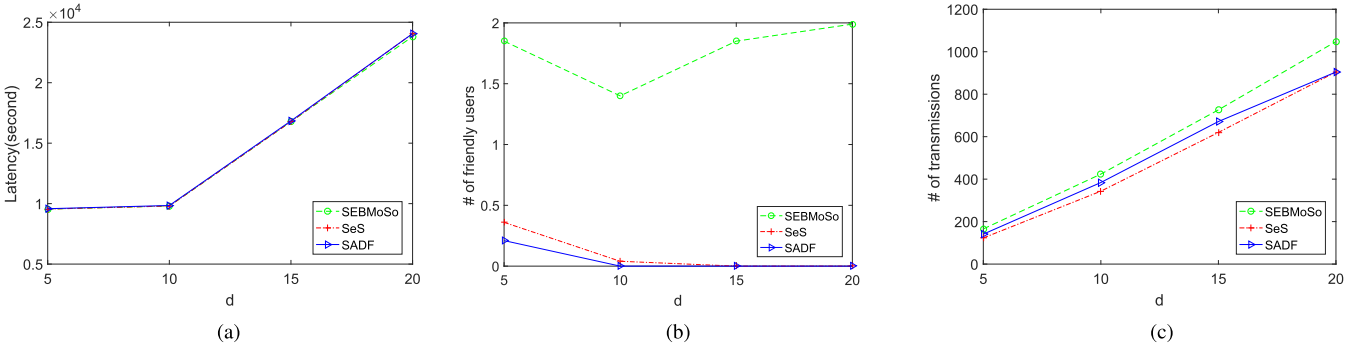

FIGURE 4. The performance evaluation upon any malicious user gets the original content under condition of 47 friendly users and 24 malicious users. (a) Time elapsed. (b) Number of delivered friendly users. (c) Number of transmissions.

messages are generated based on these data chunks. Then we do experiments on different selection of number of d where d equals to 5, 10, 15, 20 respectively. Two sets of data, where the number of malicious users is 14 and 24 respectively, are conducted to observe the performance metrics. The first one represents the situation where the number of malicious users is not posting a heavy threat. The second one represents the worse situation where it is very challenging for broadcasting the content to all friendly users. In the simulation, we assume that friendly users have no knowledge of malicious users while each malicious user knows the rest of its peers.

We first investigate the number of friendly users who have successfully received the content when any malicious user has decrypted the content. For this metric, we want to evaluate the security and efficiency performance of algorithms under the extreme condition. We then inspect the number of malicious users who have successfully filched the content under the condition that all friendly users have received the content or till the time is up. The third metric is the time performance of each algorithm. Finally, we analyzed the broadcasting cost by counting the number of total transmissions.

Two competitor algorithms are implemented to evaluate the effectiveness of our proposed *SEBMoSo*. The first one is called *SeS* [29] which uses multi-share secret scheme to protect the data content but not utilizing the credit strategy. The other one is *SADF* [27] which does not use any encryption techniques to protect the information during forwarding. It only divides the data content into several fragments. To be fair, we set the number of fragments as same as the corresponding d in each experiment setup. There is also no limitation about how many data chunks can be exchanged during an encountering event.

We first evaluate the impact of different choosing of d under the termination condition of any malicious user filching the content. In the simulation there are 57 friendly users and 14 malicious users. As shown in Fig. 3(a), when the first malicious user decodes the original content, all three methods have the same time performance which means our method does not put burden on the latency. With the increasing of d, the latency also increases accordingly. It means when d increases, it becomes more difficult to collect at least d multi-share messages to reconstruct the k data chunks. Fig. 3(b) shows the number of delivered friendly users. We can tell that *SEBMoSo* always archives the best performance. In the beginning, our scheme's performance almost triples that of *SADF* method and doubles that of *SeS* method. When d increases, the number of delivered friendly users drops but our *SEBMoSo* method is very stable. The other two methods can hardly achieve one successful friendly user when d is 20. The transmission cost is shown as in Fig. 3(c). The *SEBMoSo* method has similar transmission times with the *SeS* method but be a little better than the *SADF* method.

As can be seen in Fig. 4, we increase the number of malicious users to 24. The trend of curves of each method

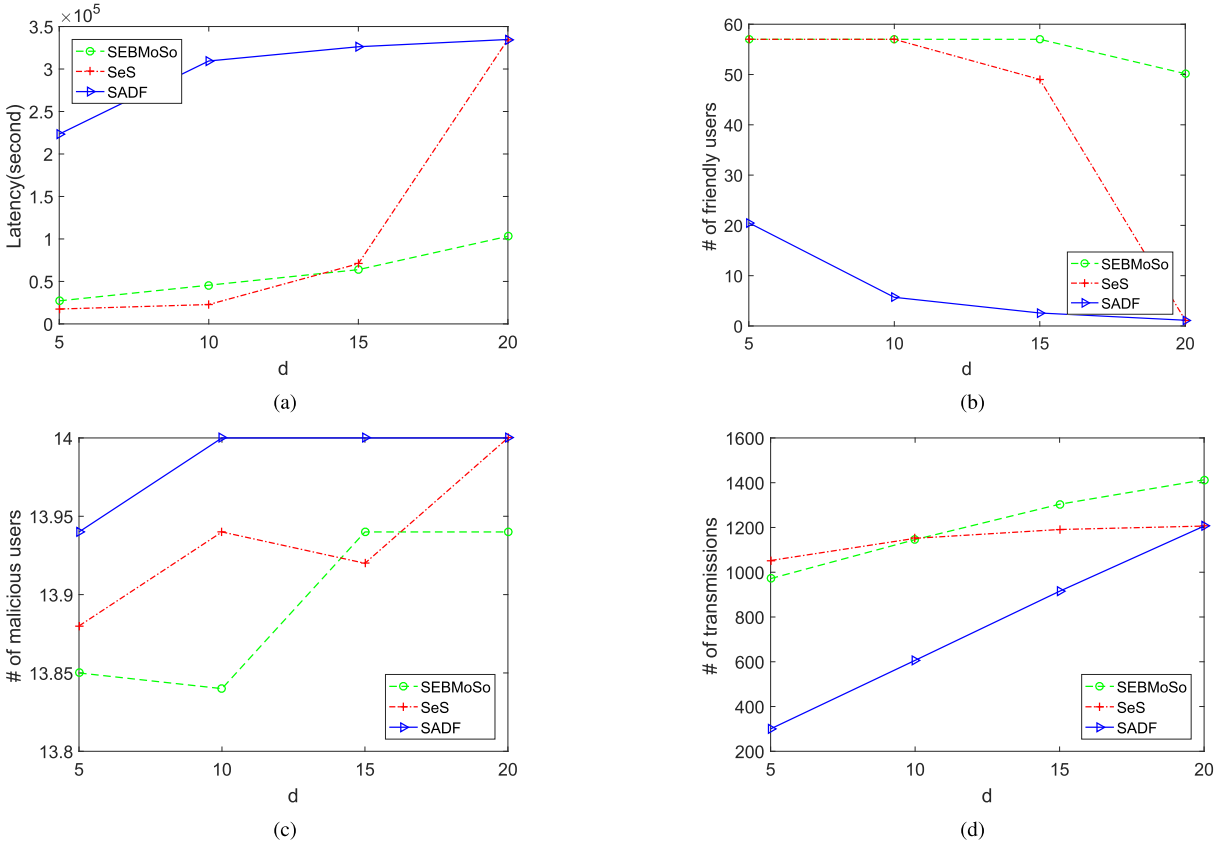

FIGURE 5. The performance upon all friendly users get the original content under condition of 57 friendly users and 14 malicious users. (a) Time elapsed. (b) Number of delivered friendly users. (c) Number of delivered malicious users. (d) Number of transmissions.

remains similar. For the time performance, the proposed *SEBMoSo* method is slightly better than the other two. Regarding the number of delivered friendly users, even *SEBMoSo* method can only achieve around 2 users. The other two can not get any friendly users obtaining a whole content when a malicious user have successfully filched the content. The strange thing is that when d increases our *SEBMoSo* method can even deliver the whole content to more friendly users. It mainly because it also becomes more difficult for malicious users to obtain the whole content with the increase of d. In the Fig. 4(c), the *SEBMoSo* uses more transmissions mainly because it delivered the content to more friendly users.

Fig. 5 shows the case that when all the friendly users get the original content or the time is up for the simulation. We can see the latency of our *SEBMoSo* is very low so that the efficient broadcasting of data content is achieved (as shown in Fig. 5(a)). Since it uses the credit scheme so that when d is small, it slightly costs more time than the *SeS* Method. However, when the d is very large, *SeS* and *SADF* experience a large delay. Fig. 5(b) shows the number of delivered friendly user of each method. The *SEBMoSo* can achieve 100% delivery ratio when d is smaller than 20. The two competitors cannot delivery as many as friendly users as *SEBMoSo*. When d is 20, they can only deliver the original content to a few friendly users. In Fig. 5(c), *SADF* results in all malicious users obtaining the original content since it does not provide security. The *SeS* method is better than *SADF* but not as good as *SEBMoSo*. Finally, in Fig. 5(d), we can see that *SEBMoSo* method has reasonable number of transmissions since it delivers to more friendly users.

Fig. 6 shows the case that when all the friendly users get the original content or the time is up for the simulation under the condition that there are 47 friendly users and 24 malicious users. Compared with a considerable increasing in latency for the other two methods, our *SEBMoSo* method almost has no change compared with the situation when there are less malicious users in the network (as shown in Fig. 6(a)). Also, as shown in Fig. 6(b), *SEBMoSo* method can still achieve 100% deliver ratio of friendly users when d is less than 20. The number of delivered friendly users drops very sharply for the two competitors. For *SADF* method, it can hardly deliver just one friendly user. Regarding protecting the content as can be seen in Fig. 6(c), our *SEBMoSo* method even outperforms the other two methods even better when the network becomes more risky. Still, the number of transmissions is reasonable for all three methods as shown in Fig. 6(d).

In Fig. 7, we want to see some detail of the situation, e.g., how friendly users and malicious user are increasing along the the time scale so that we fix $d = 5$ and adopt 24 malicious users. We can see that in early part of the

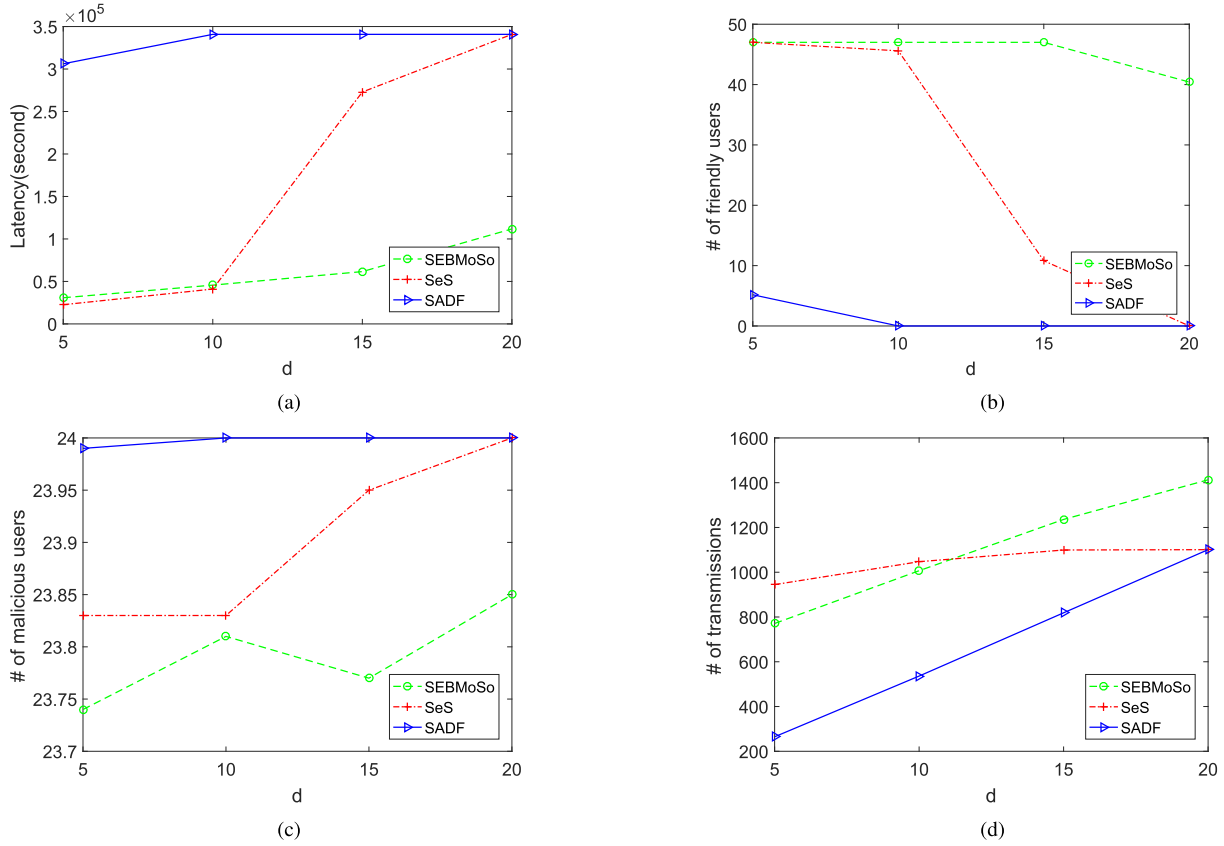

FIGURE 6. The performance upon all friendly users get the original content under condition of 47 friendly users and 24 malicious users. (a) Time elapsed. (b) Number of delivered friendly users. (c) Number of delivered malicious users. (d) Number of transmissions.

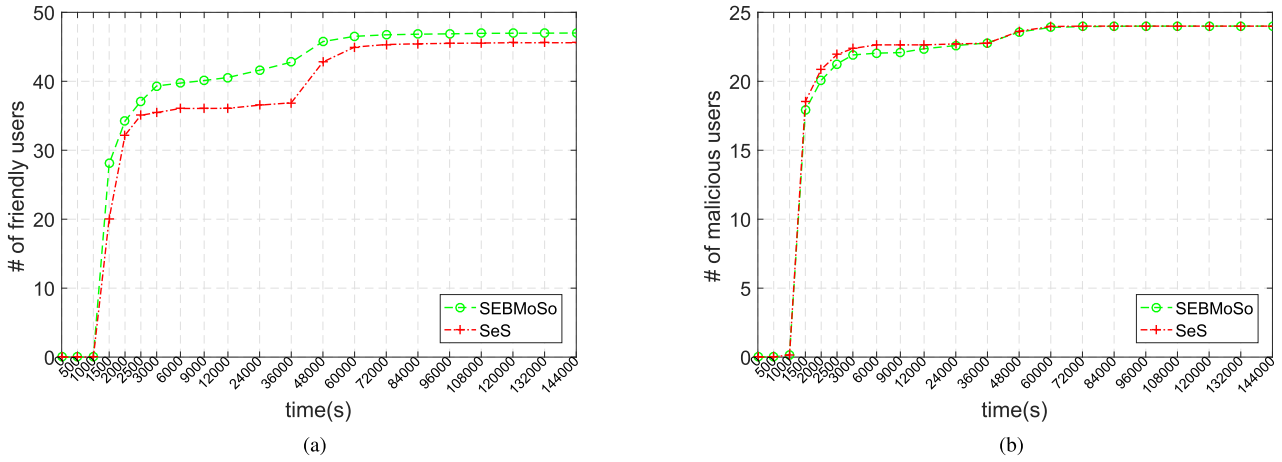

FIGURE 7. The number of delivered users changing along the time scale under the condition that there are 47 friendly users and 24 malicious users. (a) Time elapsed. (b) Number of delivered friendly users.

time, *SEBMoSo* outperforms *SeS* but when the time goes, the results get closer. It tells us we can apply Time-to-Live (TTL) to each message to achieve better performance.

We can draw a conclusion from the experiment, *SADF* method exposes the original content to all malicious users in a short time while only part of friendly users get the content. *SeS* achieves better balance, the content reaches almost all friendly users in a reasonable time before serious information leakage. Our *SEBMoSo* outperforms the two competitors in security and delay metrics. Especially, when d equals to 10, it has the best performance. When d increases, actually the performance of the method drops. Therefore, the choice of d should be based on k, not be too large. Then both efficiency and security can be achieved. Consider the system parameter

setup, we can set $d = k$ and $m = 2 \times d$. The complexity of all three methods are the same which is O(n).

VII. CONCLUSIONS

It is challenging to broadcast contents with large size such as a multimedia file in a risky mobile social network. Strict secure strategies could prevent malicious users from participating in the dissemination process while lightweight security stand could lead the exposure of the information. Furthermore, in a DTN like mobile social network, the message forwarding relies on the device-to-device communication so that any centralized security or privacy preserving schemes are difficult to apply. In this work, we propose a credit and secret-sharing based data forwarding approach to deal with the trade-off between the efficiency and security in a decentralized manner. The experimental results show that it archived fast and wide spreading while limiting the exposure of the information. In the future, we aim to optimize the total number of multi-share messages and the threshold for content reconstruction. We will find the hidden rule inside the system parameters.

ACKNOWLEDGMENT

(Tingting Fu and Peng Liu contributed equally to this work.)

REFERENCES

[1] N. Zhang, N. Cheng, A. T. Gamage, K. Zhang, J. W. Mark, and X. Shen, "Cloud assisted HetNets toward 5G wireless networks," *IEEE Commun. Mag.*, vol. 53, no. 6, pp. 59–65, Jun. 2015.

[2] C. Guo, D. Li, G. Zhang, and M. Zhai, "Real-time path planning in urban area via vanet-assisted traffic information sharing," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 5635–5649, Jul. 2018, doi: 10.1109/TVT.2018.2806979.

[3] Z. Su, Q. Xu, J. Luo, H. Pu, Y. Peng, and R. Lu, "A secure content caching scheme for disaster backup in fog computing enabled mobile social networks," *IEEE Trans. Ind. Informat.*, to be published, doi: 10.1109/TII.2018.2849984.

[4] M. Xiao, J. Wu, and L. Huang, "Community-aware opportunistic routing in mobile social networks," *IEEE Trans. Comput.*, vol. 63, no. 7, pp. 1682–1695, Jul. 2014.

[5] Q. Ayub, S. Rashid, M. S. M. Zahid, and A. H. Abdullah, "Contact quality based forwarding strategy for delay tolerant network," *J. Netw. Comput. Appl.*, vol. 39, pp. 302–309, Mar. 2014.

[6] P. Liu, Y. Wu, J. Qiu, G. Dai, and T. Fu, "elighthouse: Enhance solar power coverage in renewable sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 9, no. 11, p. 256569, 2013.

[7] G. Zhang, W. Zhang, Y. Cao, D. Li, and L. Wang, "Energy-delay tradeoff for dynamic offloading in mobile-edge computing system with energy harvesting devices," *IEEE Trans. Ind. Informat.*, to be published, doi: 10.1109/TII.2018.2843365.

[8] S. Hu *et al.*, "Towards automatic phone-to-phone communication for vehicular networking applications," in *Proc. INFOCOM*, Apr./May 2014, pp. 1752–1760.

[9] R. Ferreira, W. Moreira, P. Mendes, M. Gerla, and E. Cerqueira. (2014). "Improving the delivery rate of digital inclusion applications for Amazon riverside communities by using an integrated Bluetooth DTN architecture." [Online]. Available: https://arxiv.org/abs/1405.7084

[10] H. Chen, W. Lou, Z. Wang, and Q. Wang, "A secure credit-based incentive mechanism for message forwarding in noncooperative DTNs," *IEEE Trans. Veh. Technol.*, vol. 65, no. 8, pp. 6377–6388, Aug. 2016.

[11] V. Erramilli, M. Crovella, A. Chaintreau, and C. Diot, "Delegation forwarding," in *Proc. MobiHoc*, 2008, pp. 251–260.

[12] A. Mei, G. Morabito, P. Santi, and J. Stefa, "Social-aware stateless routingin pocket switched networks," *IEEE Trans. Parallel Distrib. Syst*, vol. 26, no. 1, pp. 252–261, Jan. 2015.

[13] J. Wu and Y. Wang, "Hypercube-based multipath social feature routing in human contact networks," *IEEE Trans. Comput.*, vol. 63, no. 2, pp. 383–396, Feb. 2014.

[14] H. Zheng and J. Wu, "Up-and-down routing in mobile opportunistic social networks with bloom-filter-based hints," in *Proc. IWQoS*, May 2014, pp. 1–10.

[15] M. Xiao, J. Wu, C. Liu, and L. Huang, "TOUR: Time-sensitive opportunistic utility-based routing in delay tolerant networks," in *Proc. INFOCOM*, Apr. 2013, pp. 2085–2091.

[16] Y. Wang, J. Wu, and W.-S. Yang, "Cloud-based multicasting with feedback in mobile social networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 12, pp. 6043–6053, Dec. 2013.

[17] Z. Su, Y. Hui, Q. Xu, T. Yang, J. Liu, and Y. Jia, "An edge caching scheme to distribute content in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5346–5356, Jun. 2018, doi: 10.1109/TVT.2018.2824345.

[18] R. Lu, X. Lin, H. Zhu, X. Shen, and B. Preiss, "Pi: A practical incentive protocol for delay tolerant networks," *IEEE Trans. Wireless Commun.*, vol. 9, no. 4, pp. 1483–1493, Apr. 2010.

[19] M. M. E. A. Mahmoud and X. Shen, "A secure payment scheme with low communication and processing overhead for multihop wireless networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 2, pp. 209–224, Feb. 2013.

[20] K. Chen, H. Shen, and L. Yan, "Multicent: A multifunctional incentive scheme adaptive to diverse performance objectives for DTN routing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 6, pp. 1643–1653, Jun. 2015.

[21] Q. Xu, Z. Su, Q. Zheng, M. Luo, and B. Dong, "Secure content delivery with edge nodes to save caching resources for mobile users in green cities," *IEEE Trans. Ind. Informat.*, vol. 14, no. 6, pp. 2550–2559, Jun. 2018, doi: 10.1109/TII.2017.2787201.

[22] C. A. Baburaj and K. Alagarsamy, "An efficient secure routing mechanism for preventing wormhole and black hole attacks in a trusted DTN environment," *Int. J. Wireless Mobile Comput.*, vol. 9, no. 2, pp. 140–147, 2015.

[23] A. Sánchez-Carmona, S. Robles, and C. Borrego, "PrivHab+: A secure geographic routing protocol for DTN," *Comput. Commun.*, vol. 78, pp. 56–73, Mar. 2016.

[24] L. Guo, C. Zhang, H. Yue, and Y. Fang, "PSaD: A privacy-preserving social-assisted content dissemination scheme in DTNs," *IEEE Trans. Mobile Comput.*, vol. 13, no. 12, pp. 2903–2918, Dec. 2014.

[25] L. Zhang, J. Song, and J. Pan, "A privacy-preserving and secure framework for opportunistic routing in DTNs," *IEEE Trans. Veh. Technol.*, vol. 65, no. 9, pp. 7684–7697, Sep. 2016.

[26] Y. Ding, P. Liu, and Z. Guan, "Energy efficient multimedia content broadcasting in a risky mobile social network with adversarial users," in *Proc. GreenCom*, Jun. 2017, pp. 510–515.

[27] L. Feng, Y. Zhang, and H. Li, "Large file transmission using self-adaptive data fragmentation in opportunistic networks," in *Proc. CSNT*, Apr. 2015, pp. 1051–1055.

[28] M. Franklin and M. Yung, "Communication complexity of secure computation," in *Proc. 24th Annu. ACM Symp. Theory Comput.*, 1992, pp. 699–710.

[29] Z. Lu, X. Sun, and T. F. La Porta. (2016). "Cooperative data offload in opportunistic networks: From mobile devices to infrastructure." [Online]. Available: https://arxiv.org/abs/1606.03493

[30] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," in *Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur.*, 2001, pp. 514–532.

TINGTING FU received the M.S. degree in soft engineering from Zhejiang University, China, in 2005. From 2014 to 2015, she was a Visiting Scholar with Lehigh University, USA. She is currently a Lecturer with Hangzhou Dianzi University. Her interests are in the area of database system design, big data, and mobile computing.

PENG LIU received the B.S. and M.S. degrees in computer science and technology from Hangzhou Dianzi University in 2001 and 2004, respectively, and the D.Eng. degree in computer science and technology from Zhejiang University, China, in 2007. He was a Visiting Scholar with the University of Bradford, U.K., Temple University, USA, and the University of Waterloo, Canada. He was involved in the area of embedded systems, ubiquitous computing, Internet-of-Things, vehicular ad hoc networks, and edge computing. He is currently an Associate Professor with Hangzhou Dianzi University.

YUE DING is currently pursuing the master's degree with Hangzhou Dianzi University. Her research interests include delay tolerant networks and vehicular ad hoc networks.

YUAN ZHANG is currently pursuing the Ph.D. degree with the University of Electronic Science and Technology of China. He is a Visiting Student with the University of Waterloo, Canada. His research interests include cloud storage security and social privacy preserving.