Research Article

Two Anonymous Cooperative Cache-Based Data Access Schemes in Mobile Ad Hoc Networks

Chang-Ji Wang, 1,2,3 Xi-Lei Xu, 1,2 and Dong-Yuan Shi 1,2

1 School of Information Science and Technology, Sun Yat-sen University, Guangzhou 510006, China
2 Guangdong Province Information Security Key Laboratory, Guangzhou 510006, China
3 Research Center of Software Technology for Information Service, South China Normal University, Guangzhou 501631, China

Correspondence should be addressed to Chang-Ji Wang; isswchj@mail.sysu.edu.cn

Received 8 October 2013; Accepted 20 November 2013

Academic Editor: Fatos Xhafa

Copyright © 2013 Chang-Ji Wang 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.

Mobile ad hoc network has been extensively studied in recent years due to its potential applications in civilian and military environments. Cooperative caching, which allows the sharing and coordination of cached data among multiple nodes, could be employed to improve data accessibility and reduce data access cost in mobile ad hoc networks. In this paper, we give anonymity requirements for cooperative cache-based data access in mobile ad hoc networks and present two efficient anonymous data access schemes based on onion message and pseudonym-based encryption, respectively. The proposed schemes can not only protect confidentiality of sensitive cache data but can also protect privacy of nodes and routes.

1. Introduction

A mobile ad hoc network (MANET) is a self-configuring temporary network of mobile devices connected by wireless links without infrastructure support. The benefits of flexible routing, global connectivity and a highly adaptive potential make MANETs suitable for a wide range of applications in both military and commercial environments, such as battlefields, disaster relief operations, personal area networking, mobile information sharing, and vehicular networks.

Similar to the wired network, security requirements for MANETs include availability, confidentiality, integrity, authentication, nonrepudiation. Compared to the wired network, MANET is more vulnerable to security attacks due to its features of open medium, dynamic changing topology, cooperative algorithms, lack of centralized monitoring and management point, and so forth [1].

Traffic analysis is one of the most serious security attacks in MANETs due to the broadcast nature of the wireless medium. In this attack, adversaries can obtain sensitive information about the applications even without revealing the contents of the messages just by observing network traffic pattern. For example, an attacker can identify the communicating parties and their positions by tracing and analyzing the network traffic patterns. This may lead to severe threats in security-sensitive applications. For instance, in a battlefield the enemy can physically destroy the important mobile nodes if they can identify and locate such nodes by traffic analysis. Ideally, a node should be able to keep its identity, its location and its correspondents private, that is, remain anonymous. Simple traffic padding approach by generating dummy traffic into the network does not aim to hide the identifiers of communicating nodes and so cannot completely prevent traffic analysis.

In order to thwart traffic analysis attack, anonymous mobile ad hoc routing protocols are developed and have been attracting the attention of more and more researchers in recent years. The most current anonymous routing protocols for MANETs assume that mobile users share pairwise secrets before they start an anonymous communication session. This assumption is impractical for many ad hoc scenarios where pairwise shared keys are difficult to be set up in advance.

Song et al. [2] presented an anonymous dynamic source routing protocol for MANETs. The protocol employs anonymous onion routing between the source and destination, and each intermediate node owns a shared session key with the
source and destination nodes when the protocol is completed. Zhang et al. [3] proposed an anonymous on-demand routing protocol based on identity-based encryption [4] for MANETs. In their approach, nodes use pseudonyms instead of their real identifiers, and each node can dynamically changes its pseudonyms. Each node self-generates a set of pseudonyms and submit them to the trusted authority, and the trusted authority generates corresponding private keys and sends them back to the node. The drawback of Zhang et al.'s approach is the underlying anonymous communications are not blind to the trusted private key generator (PKG). Later, Huang [5] presented an anonymous communication system for MANETs based on pseudonym-based cryptography without trusted PKG. Sheklabadi and Berenjkoub [6] proposed an anonymous version of authenticated routing for ad hoc networks (ARAN), which can provide a variety of anonymous functionalities such as identity privacy, location privacy, and route anonymity together with security features of ARAN. However, the proposed protocol is based on an impractical assumption that there is a shared secret between the source and the destination.

Although routing is an important issue in MANETs, other issues such as data access are also very important since the ultimate goal of using MANETs is to provide data access to mobile nodes. Caching techniques are an efficient way for increasing the performance in data communication, which has been widely used in different fields such as CPU design, multiprocessor, memory architecture, or router design. Internet uses cache placement and replacement in proxy servers and cooperative caching architecture to reduce the network traffic and average latency of data query significantly [7]. As mobile nodes in ad hoc networks may have similar tasks and share common interest, cooperative caching, which allows the sharing and coordination of cached data among multiple nodes, can be used to reduce the bandwidth and power consumption. However, cooperative caching techniques designed for wired networks may not be applied directly to MANETs due to their dynamic topology and constrained resources (i.e., bandwidth, battery power, and computational capacity).

Ting and Chang [8] proposed a cooperative caching scheme for wireless ad hoc networks called GroupCaching, which allows each mobile host and its 1-hop neighbors form a group, and the caching status is exchanged and maintained periodically in a group. Yin and Cao [9] proposed three cooperative caching schemes for ad hoc networks, called CachePath, CacheData, and HybridCache, respectively. Zhao et al. [10] proposed a novel asymmetric cooperative cache approach, where the data requests are transmitted to the cache layer on every node, but the data replies are only transmitted to the cache layer at the intermediate nodes that need to cache the data. Recently, Wu and Cao [11] proposed a Voronoi Diagram-based cooperative cache discovery approach to reduce data access cost by limiting the cache information update and query within a single Voronoi Region. Currently existing cooperative caching schemes in MANETs have not yet considered the problem of anonymity.

In this paper, we give anonymity requirements for cooperative cache based data access in MANETs, and present two efficient anonymous cooperative cache based data access schemes based on onion message and pseudonym-based encryption, respectively. The proposed schemes cannot only protect confidentiality of sensitive cache data but can also protect privacy of nodes and routes.

The rest of this paper is organized as follows. Some necessary background knowledge about bilinear pairings and pseudonym-based encryption are introduced in Section 2. The anonymity requirements and system model of anonymous cooperative cache based data access in MANETs are explained in Section 3. Our proposed anonymous cooperative cache based data access schemes in MANETs based on onion message and pseudonym-based encryption are described in Sections 4 and 5, respectively. Anonymity and security analysis of the two proposed cooperative cache based data access schemes in MANETs are presented in Section 6. We conclude our work in Section 7.

2. Preliminary Works

We first introduce some notations. If S is a set, then $x \in S$ denotes the operation of picking an element $x$ uniformly at random from S. A function $f(\lambda)$ is negligible if for every $c > 0$, there exists a $\lambda_c$, such that $f(\lambda) < 1/\lambda^c$ for all $\lambda > \lambda_c$.

2.1. Bilinear Pairings and the Bilinear Diffie-Hellman Assumption

Let $G_1$ be an additive cyclic group of large prime order $q$ and $G_2$ be a multiplicative cyclic group of the same order $q$. In particular, $G_1$ is a subgroup of the group of points on an elliptic curve over a finite field and $G_2$ is a subgroup of the multiplicative group over a finite field. Let $P$ be a generator of $G_1$. A bilinear pairing is an admissible map $\hat{e} : G_1 \times G_1 \rightarrow G_2$ and satisfies the following properties.

(i) **Bilinearity.** For $a, b \in \mathbb{Z}_q^*$, we have $\hat{e}([a]P, [b]P) = e(P, P)^{ab}$.

(ii) **Nondegeneracy.** The map does not send all pairs in $G_1 \times G_1$ to the identity in $G_2$, this implies that $\hat{e}(P, P)$ is a generator of $G_2$.

(iii) **Computability.** There is an efficient algorithm to compute $\hat{e}(P_1, P_2)$ for $P_1 \in G_1$ and $P_2 \in G_2$.

The bilinear group generator $\bar{q}(1^\lambda)$ is an algorithm that takes as input a security parameter $\lambda$ and outputs a bilinear group $(q, G_1, G_2, \hat{e})$, where $q$ is a prime of size $2^\lambda$, $G_1$ and $G_2$ are cyclic groups of order $q$, and $\hat{e}$ is an admissible bilinear map.

Given a bilinear group $(q, G_1, G_2, \hat{e})$ generated by $\bar{q}(\lambda)$, and $(P, aP, bP, cP)$ for some $a, b, c \in \mathbb{Z}_q^*$. The BDH problem in the bilinear group $(q, G_1, G_2, \hat{e})$ is to compute $W = \hat{e}(P, P)^{abc} \in G_2$. The advantage of an algorithm $\mathcal{A}$ in solving the BDH problem in the bilinear group $(q, G_1, G_2, \hat{e})$ is defined by

$$\text{Adv}_{\hat{e}, \text{BDH}}(\lambda) = \Pr \left[ \mathcal{A}(P, aP, bP, cP) = \hat{e}(P, P)^{abc} \right].$$ (1)

where the probability is over the random choice of $a, b, c$ in $\mathbb{Z}_q^*$, the random choice of $P \in G_1$, and the random bits of $\mathcal{A}$. 
We say that the BDH assumption holds in the bilinear group \((q, G_1, G_2, \bar{e})\) if no probabilistic polynomial time (in \(\lambda\)) algorithm has a nonnegligible advantage in solving the BDH problem in the bilinear group \((q, G_1, G_2, \bar{e})\).

### 2.2. Pseudonym-Based Encryption Scheme.

Huang proposed a pseudonym-based encryption (PBE) scheme without trusted PKG to provide anonymous communication [5], which is described as follows.

(i) **Setup.** The PKG generates and publish system parameters as

\[
\text{params} = \left( G_1, G_2, \bar{e}, q, P, P_{\text{pub}} = [s] P, n, H, H_2, H_3, H_4 \right),
\]

where \(n\) denotes bit length of plaintext, \(H : G_1 \times G_2 \rightarrow G_1, H_2 : G_2 \rightarrow \{0, 1\}^n, H_3 : \{0, 1\}^{2n} \rightarrow Z_q^2,\) and \(H_4 : \{0, 1\}^n \rightarrow \{0, 1\}^n\) denotes cryptographic hash functions, respectively. The PKG keeps the master secret key \(s \in_R Z_q^*\) secret.

(ii) **Extract.** To self-generate a pseudonym and corresponding private key, an anonymous user first chooses \(k = r \in_R G_1\) and \(\bar{e} = e(Q_A, P)\), then he computes \(k^{-1}\) satisfying \(k \cdot k^{-1} \equiv 1 \mod q\). Then, he sets \(Q_A = \bar{e}^{-1} P\) as his pseudonym, and sets \(S_{PD_A} = [sk] P\) as the corresponding private key.

(iii) **Encrypt.** To encrypt the plaintext \(M \in \{0, 1\}^n\) for an entity \(A\) with pseudonym \(PD_A\), the sender chooses a random \(r \in_R \{0, 1\}^n\), sets \(t = H_3(r, M)\), computes and outputs the ciphertext \(C = \langle U, V, W \rangle\), where \(U = [t] P, V = r \oplus H_2(e(Q_A, P_{\text{pub}}) e(\bar{e}^{-1})),\) and \(W = M \oplus H_4(r)\).

(iv) **Decrypt.** To decrypt \(C\), the intended receiver \(A\) first computes \(t' = V \oplus H_2(e(S_{PD_A}, U))\) and \(M' = W \oplus H_4(t')\), then \(A\) sets \(t' = H_3(t', M')\) and tests if \(U = [t'] P\). If it does not hold, \(A\) rejects the ciphertext. Otherwise, \(A\) outputs \(M'\) as the decryption of \(C\).

Huang [5] proved that the above scheme is semantically secure pseudonym based encryption in the random oracle model assuming BDH is hard in groups generated by \(\bar{e}\).

### 3. System Model

The system model for our proposed anonymous cooperative cache based data access schemes in MANETs is illustrated as Figure 1. There are \(n\) intermediate nodes between the source node and the destination node along the routing path. We denote the source node, nodes en route, and the destination node as \(S, X_i\) (\(1 \leq i \leq n\)), and \(D\), respectively.

The proposed anonymous cooperative cache based data access schemes in MANETs consist of the following four steps.

(i) **Route Request Phase.** The source node first constructs and initializes the route request message packet (RREQ) and broadcasts to search for the destination node who has a copy of required data cache.

(ii) **Route Reply Phase.** The destination node constructs and sends a route reply packet (RREP) back to the source node along the path found by the first step.

(iii) **Data Request Phase.** The source node sends data cache request message packet (DREQ) to the destination node.

(iv) **Anonymous Data Transmission Phase.** The destination node transfers the requested data cache back to the source node.

We define an internal adversary as a node that is compromised and on the routing path. An external adversary is a compromised node not on the path, or an external node not directly participating in the MANET, that is, it only eavesdrops on traffic between nodes.

We make the following two assumptions for our anonymous cooperative cache-based data access schemes in MANETs.

(i) All wireless links are symmetric, which means that if node \(A\) is in transmission range of some node \(B\), so \(B\) is in transmission range of \(A\) as well.

(ii) Adversaries have unbounded eavesdropping capability but bounded computing and node intrusion capabilities.

To resist outsider attack, our anonymous cooperative cache based data access schemes should prevent outsider adversary from learning the identity of source node and destination node, and intermediate nodes en route. To resist insider attack, our anonymous cooperative cache based data access schemes should prevent insider adversary from determining whether another node is the source or the destination, or an intermediate node en route of a particular message. So, we consider the following three types of anonymity for our anonymous cooperative cache based data access schemes in MANETs.

(i) **Anonymity of source and destination nodes.** Both the source and destination nodes is anonymous to others, and no one knows the location of the source node or destination node.

(ii) **Anonymity of intermediate nodes.** Nobody knows the real identities of intermediate nodes, even the node en route cannot know the real identities and locations of other nodes.

(iii) **Anonymity of the route.** Adversaries cannot trace a packet flow back to its source or the destinations. Nobody knows the real routing path between the source and destination nodes. For adversaries not in the route, they have no information on any part of the route.
For convenience of presentation, we introduce the notations that will be used in our proposed anonymous cooperative cache based data access schemes in notations section.

4. Anonymous Data Access Scheme Based on Onion Packet

4.1. Route Request Phase. The route request phase allows a source node $S$ to discover and establish a routing path to a destination node $D$ through a number of intermediate nodes. To keep communication anonymity, none of the intermediate nodes participating in this phase should discover the identities of $S$ and $D$. The source node $S$ constructs a route request message packet with the following format and broadcasts to all its neighboring nodes:

$$[\text{RREQ}, \text{SEQ}, \text{PDN}_s, \text{CID}, \text{PDN}_{i-1}, p_{k_i}] . \quad (3)$$

RREQ field indicates that it is the the route request message packet, SEQ field indicates the packet sequence number that serves as the unique identifier of the request, and it is also used by the intermediate nodes to validate whether an route reply is generated by the real destination in the route reply phase. PDN$_s$ field indicates the pseudonym of the source node, which is the temporary identity of node $S$.

When a node $X_i$ receives a route request message packet with the following format

$$[\text{RREQ}, \text{SEQ}, \text{PDN}_s, \text{CID}, \text{PDN}_{i-1}, p_{k_i}] , \quad (4)$$

it processes the packet according to the following steps.

1. Check whether the packet has already been received by using the fields SEQ as the unique identifier for the packet.
2. If the packet has not been received, then $X_i$ checks whether it has the requested cache data or not.
   
   (i) If the node has the requested cache data (i.e., node $X_i$ is the destination node), then the process jumps to the route reply phase.
   
   (ii) If node $X_i$ has not the requested cache data, it records SEQ, PDN$_s$, PDN$_{i-1}$ into its routing table and generates a new routing table entry as follows:

$$[\text{SEQ}, \text{PDN}_s, \text{PDN}_{i-1}, \text{PDN}_i, K_i, \text{TTL}] . \quad (5)$$

The field PDN$_s$ denotes a pseudonym generated by node $X_i$ for this session, the field $K_i$ denotes a symmetric session key generated randomly by node $X_i$ that will be used to encrypt the data from neighbor (this field is not set in this phase). Finally node $X_i$ replaces the field PDN$_{i-1}$ with PDN$_s$ in the received route request message packet and rebroadcasts the request to all its neighbors.

3. If the packet has been received, then $X_i$ checks whether the pseudonym PDN$_{i-1}$ has been recorded in the routing table associated with SEQ.

(i) If PDN$_{i-1}$ is one of the nodes in the routing table, drop the packet and stop.

(ii) Otherwise, node $X_i$ records the pseudonym PDN$_{i-1}$ into its routing table as one of the reverse node. The reverse node is the next node in the reverse path towards the source in the route reply phase.

4.2. Route Reply Phase. The destination node $D$ first chooses a random symmetric key $K_d$ and a pseudonym PDN$_d$ generates and appends a new routing table entry with the following format:

$$[\text{SEQ}, \text{PDN}_s, \text{PDN}_{i-1}, \text{PDN}_d, K_d, \text{TTL}] , \quad (6)$$

where PDN$_{i-1}$ is the pseudonym that the destination node $D$ got the route request message packet from the previous hop. Then the destination node generates a route reply message packet with the following format:

$$[\text{RREP}, \text{SEQ}, \text{PDN}_s, \text{PDN}_{i-1}, p_{k_d}, \text{ONI}] \quad (7)$$

where $\text{ONI} = \{ E_{p_{k_d}}(K_d), E_{K_d}(\text{PDN}_d) \}$.

Finally, the destination node broadcasts the route reply message packet to all its neighbors.

For every intermediate node $X_i$ who receives the route reply message packet, it first checks whether there is an entry in its anonymous routing table whose SEQ and PDN$_s$ fields are the same as those of the received packet. If there exists such an entry, then node $X_i$ checks whether the field PDN$_i$ of this entry is equal to the field PDN$_{i-1}$ of the packet. If yes, node $X_i$ chooses a symmetric key $K_i$ and fills it into the field $K_i$ of this entry. Furthermore, node $X_i$ replaces the field PDN$_{i-1}$ of the packet with the field PDN$_{i-1}$ recorded by itself, and changes field ONI of the packet into the following format:

$$\text{ONI} = \{ E_{p_{k_i}}(K_i), E_{K_i}(\text{ONI}) \} . \quad (8)$$

Finally, node $X_i$ broadcasts the packet to its neighbors. This process loops until the source node $S$ gets the packet. The source node $S$ decrypts the received ONI by using its corresponding private key $s_{K_i}$, thus the source node can get all the symmetric encryption keys $K_i$ and pseudonym of nodes PDN$_i$ along the path.

4.3. Data Request Phase. The source node $S$ generates and broadcasts the data cache request packet with the following format:

$$[\text{DREQ}, \text{PESU}_i, \text{ONI}] , \quad (9)$$

where

$$\text{ONI} = E_{K_i} [\text{IMN}, \text{PDN}_j, \ldots, E_{K_{d'}} [\text{END}, \text{RT}] \ldots]$$

$$\text{RT} = E_{p_{k_d}} [\text{PDN}_1, K_1, \ldots, \text{PDN}_i, K_i, \ldots, \text{PDN}_d, K_d] . \quad (10)$$

International Journal of Distributed Sensor Networks
The intermediate node $X_i$ determine whether he is in the path by checking the field PDN$_i$ in the received data packet against his pseudonym. If yes, node $X_i$ decryps the ONI and gets the pseudonym for the next hop. Then node $X_i$ replaces the field PDN$_i$ with decrypted pseudonym. For the destination node $D$, when it decrypts the ONI, it gets the route information.

4.4. Anonymous Data Transmission Phase. The process of data cache transmission is similar with the process of the data cache request. The difference is that the route path information are included in the process of data cache request, while the cache-related data are included in the process of data cache transmission.

Every node has to broadcast the request packets to the entire MANET to get the cache data in our proposed scheme. Recently, Wu and Cao [11] proposed a Voronoi Diagram based cooperative cache discovery approach to reduce data access cost by limiting the cache information update and query within a single Voronoi Region. We can further extend our anonymous data access scheme according to the following idea: for each type of cache, all nodes send the request packets. If nodes receive route reply packets, they compute the length of hops for each received packet and choose the shortest two. If the two shortest paths are equal, then we assure that the node is the edge node of the Voronoi Region. So for that type of cache, the node will stop broadcasting the request packet to these nodes.

5. Anonymous Data Access Scheme Based on PBE

5.1. Route Request Phase. The source node $S$ cannot find some data in the local cache, it asks for its neighbor nodes by broadcasting a route request message packet with the following format:

$$[\text{RREQ}, \text{SEQ}, \text{CID}, \text{PDN}_0, \text{Sig}_{\text{PDN}_0}],$$

(11)

where PDN$_0$ and PDN$_i$ are two pseudonyms generated by the source node, and PDN$_i$ is used to represent source node’s identity, while PDN$_i$ will be used to encrypt the requested data.

Once the intermediate node $X_i$ received the route request message packet, $X_i$ first verifies the signature of the previous hop to ensure the validity of the pseudonym PDN$_{i-1}$. If the verification fails, it discards the route request message packet. Otherwise, $X_i$ checks whether it has a cache copy of the corresponding CID. If yes, node $X_i$ is the destination node. If not, node $X_i$ replaces the fourth field, PDN$_0$, in the route request message packet with its own pseudonym PDN$_i$, and add a new route record with the following format into its local routing table:

$$[\text{SEQ}, \text{CID}, \text{PDN}_{i-1}, \text{PDN}_i, 0, \text{TTL}],$$

(12)

where the fifth field is set for the pseudonym of the next hop PDN$_{i+1}$. Afterwards, it goes on broadcasting the route request message packet.

5.2. Route Response Phase. Suppose that the node $D$ finds the cache copy of the requested data according to its SEQ and CID after received the route request packet. It goes on broadcasting with certain probability to confuse adversaries in order to protect the privacy that it is the destination node. If the destination node $D$ receives many packets with the same keywords (SEQ, CID), it only responses to the first route request packet by generating a route reply message packet with the following format:

$$[\text{RREP}, \text{SEQ}, \text{CID}, \text{E}_{\text{PDN}_{i-1}} (\text{PDN}_i), \text{PDN}_{i-1}, \text{Sig}_\text{PDN}_{i-1}].$$

(13)

At the same time, it add a new route record with the following format into its local routing table:

$$[\text{SEQ}, \text{CID}, \text{PDN}_{i-1}, \text{PDN}_i, 0, \text{TTL}].$$

(14)

The intermediate node $X_i$ in the replying path firstly checks whether the field PDN$_{i-1}$ is equal to its own pseudonym or not. If it is not, node $X_i$ then discards the packet. Otherwise node $X_i$ continues to verify the signature in the packet to ensure the validity of the PDN$_{i-1}$. And then node $X_i$ checks the route table against keywords (SEQ, CID) after received the route reply packet. It discards the packet if there is not a corresponding item. Otherwise it decrypts the item $\text{E}_{\text{PDN}_{i-1}} (\text{PDN}_i)$ to get PDN$_i$ using the corresponding private key. Then node $X_i$ encrypts PDN$_{i-1}$ (the pseudonym of the node itself) using PDN$_{i-2}$ to get the new route reply packet and goes on broadcasting. The function of encryption is to keep the privacy of the identity of PDN$_{i-1}$.

5.3. Data Request Phase. After receiving the route reply packet, the source node $S$ broadcasts the data request message packet with following format:

$$[\text{DREQ}, \text{SEQ}, \text{CID}, \text{E}_{\text{PDN}_{i+1}} (\text{PDN}_i, \text{PDN}_j)] .$$

(15)

The intermediate node checks the anonymous route table to decide whether it is the corresponding node by matching the PDN$_{i+1}$ of the received packet and PDN$_i$ of the node itself. If it is the correct match, it can decrypt to obtain PDN$_i$ of the packet. Then it matches it with the PDN$_{i-1}$ in the route table to decide whether it is the right path to follow. It discards the packet if they do not match. Otherwise it decrypts the packet and replace the field PDN$_i$ with PDN$_{i+1}$ (actually the pseudonym of the node itself) and encrypts (PDN$_{i+1}$, PDN$_i$) with PDN$_{i+2}$ and keeps on broadcasting.

5.4. Anonymous Data Transmission Phase. The destination node $D$ gets the requested data by searching in its route table against keywords (SEQ, CID) after received the data request packet. Firstly it goes on broadcasting with a certain probability. Simultaneously it encrypts the data with PDN$_i$ and encrypts its pseudonym with PDN$_{i-1}$. Then it broadcasts the data cache reply message packet with the following format:

$$[\text{DREP}, \text{SEQ}, \text{CID}, \text{E}_{\text{PDN}_{i-1}} (\text{data}), \text{E}_{\text{PDN}_{i-1}} (\text{PDN}_i)].$$

(16)
The intermediate node $X_i$ decrypts the field $E_{PDN_{i-1}}(PDN_i)$ of the received data cache reply message packet to get $PDN_i$, then node $X_i$ try to match it with field $PDN_{i+1}$ in the local routing table. If there is no match, node $X_i$ discards the packet. Otherwise, node $X_i$ generates new data cache reply message packet by replacing the field $PDN_{i-1}$ and $PDN_i$ with the pseudonym of the previous hop and the current node $X_i$, respectively. Then node $X_i$ goes on broadcasting the new data cache reply message packet. Finally, the source node $S$ receives the data cache reply message packet, gets the required data by decrypting the packet with the private key corresponding to $PDN_s$, and broadcasts the packet with a certain probability.

6. Analysis on the Proposed Anonymous Data Access Schemes

In the context of anonymity analysis, we assume that adversaries may be en route as well as out of the route and they are interested in the privacy information about the route and nodes en route.

We use pseudonyms to identify the nodes without unveiling the real identities and locations of all the nodes, thus the privacy of locations and identities of the nodes are well protected. In our anonymous data access scheme based on onion packet, the information of route is encrypted, the whole routing path are only known by the source node and destination node. The path consists of pseudonyms of each node, intermediate nodes may only know the number of hops, the proposed anonymous data access scheme based on onion packet provides weak anonymity. In our anonymous data access scheme based on PBE, the information of the routing path is only recorded separately in the route table of each node with pseudonyms. Even adversaries en route can get nothing about the route except pseudonyms of its previous hop and next hop. The pseudonym of the source node used for encrypting the data is transferred along the path, however, it uses another pseudonym to participate in the route path. So even the next hop of the source node knows nothing about the fact it is the source node, thus our proposed anonymous data access scheme based on PBE provides strong anonymity.

In our proposed two anonymous data access schemes, passive eavesdropper can only catch packets encrypted by a public key or a session key. For active adversaries who may pretend to be the destination node to send fake route reply packet to the requested node and control some nodes along the routing path, they could only get the pseudonyms of the previous hop and next hop and the encrypted data packet.

In our proposed anonymous data access scheme based on onion packet, DOS attacker cannot control which node they get the cache data from, and they do not know the true identity of the cache node. In our proposed anonymous data access scheme based on PBE, the node has to be authenticated before it requests the cache data. Thus our proposed anonymous data access schemes can resist DOS attack.

7. Conclusion

Secure anonymity cooperative caching is one of the important task in mobile ad hoc networks in certain privacy vital environments. In this paper, we give anonymity requirements for cooperative cache-based data access in mobile ad hoc networks and present two efficient anonymous data access scheme based on onion message and pseudonym-based encryption, respectively. The proposed schemes cannot only protect the privacy of nodes and routes but can also resist both passive and active attacks.

**Notations**

- **RREQ**: the flag of route request message packet
- **RREP**: the flag of route reply message packet
- **DREQ**: the flag of data cache request message packet
- **DREP**: the flag of data cache reply message packet
- **IMN**: the flag of intermediate node
- **END**: the flag of destination node
- **CID**: the identifier of requested cache data
- **SEQ**: the sequence number of the current session
- **PDN_s**: the pseudonym of the source node $S$
- **PDN_d**: the pseudonym of the destination node $D$
- **PDN_i**: the pseudonym of the intermediate node $X_i$
- **TTL**: time-to-live values
- **K**: the symmetric key chosen by the intermediate node $X_i$
- **E_K(m)**: encrypt message $m$ by symmetric encryption with key $K$
- **pK_s**: the public key of the source node $S$
- **pK_d**: the public key of the destination node $D$
- **E_{pK_d}(m)**: encrypt message $m$ by asymmetric encryption with public key $pK_d$
- **E_{PDN_i}(m)**: encrypt message $m$ by using pseudonym based encryption scheme with pseudonym $PDN_i$
- **ONI**: the onion data packet
- **Sig_{PDN_i}^r**: the signature generated by node with pseudonym $PDN_i$.

**Acknowledgment**

This research is jointly funded by the National Natural Science Foundation of China (Grant no. 61173189) and the Guangdong Province Information Security Key Laboratory Project.

**References**

[1] W. J. Lou and Y. Fang, “A survey on wireless security in mobile ad hoc networks: challenges and available solutions,” in *Ad Hoc Wireless Networking*, pp. 68–73, Kluwer Academic, New York, NY, USA, 2003.

[2] R. Song, L. Korba, and G. Yee, “AnonDSR: efficient anonymous dynamic source routing for mobile ad-hoc networks,” in *ACM*
[3] Y. Zhang, W. Liu, and W. Lou, “Anonymous communications in mobile ad hoc networks,” in Proceedings of IEEE Information Communications Conference (INFOCOM ’05), pp. 1940–1951, March 2005.

[4] D. Boneh and M. Franklin, “Identity-based encryption from the Weil pairing,” in Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology (CRYPTO ’01), vol. 2139 of Lecture Notes in Computer Science, pp. 213–229, Springer, Santa Barbara, Calif, USA, 2001.

[5] D. J. Huang, “Pseudonym-based cryptography for anonymous communications in mobile ad hoc networks,” International Journal of Security and Networks, vol. 2, no. 3–4, pp. 272–283, 2007.

[6] E. Sheklabadi and M. Berenjkoub, “An anonymous secure routing protocol for mobile ad hoc networks,” in Proceedings of the International Symposium on Computer Networks and Distributed Systems (CNDS ’11), pp. 142–147, February 2011.

[7] D. B. Johnson, “Routing in ad hoc networks of mobile hosts,” in Proceedings of the Workshop on Mobile Computing Systems and Applications, pp. 158–163, December 1994.

[8] Y.-W. Ting and Y.-K. Chang, “A novel cooperative caching scheme for wireless ad hoc networks: GoupCaching,” in Proceedings of the 2nd International Conference on Networking, Architecture, and Storage (NAS ’07), pp. 62–68, July 2007.

[9] L. Yin and G. Cao, “Supporting cooperative caching in ad hoc networks,” IEEE Transactions on Mobile Computing, vol. 5, no. 1, pp. 77–89, 2006.

[10] J. Zhao, P. Zhang, G. Cao, and C. R. Das, “Cooperative caching in wireless P2P networks: design, implementation, and evaluation,” IEEE Transactions on Parallel and Distributed Systems, vol. 21, no. 2, pp. 229–241, 2010.

[11] W. G. Wu and J. N. Cao, “Efficient cache discovery for cooperative caching in wireless ad hoc networks,” in Proceedings of the 18th IEEE International Conference on Parallel and Distributed Systems (ICPADS ’12), pp. 323–330, 2012.