Hindawi Publishing Corporation
Journal of Sensors
Volume 2015, Article ID 365863, 10 pages
http://dx.doi.org/10.1155/2015/365863

Research Article

A Study on High Secure and Efficient MANET Routing Scheme

Wei-Chen Wu[1] and Horng-Twu Liaw[2]

[1]*Computer Center, Hsin Sheng Junior College of Medical Care and Management, No. 418, Zhongfeng Road, Sec. Gaoping, Longtan District, Taoyuan City 325, Taiwan*
[2]*Department of Information Management, Shih Hsin University, No. 1, Lane 17, Sec. 1, Muja Road, Wenshan Chiu, Taipei 116, Taiwan*

Correspondence should be addressed to Wei-Chen Wu; wwu@hsc.edu.tw

Received 29 November 2014; Accepted 15 February 2015

Academic Editor: Young-Sik Jeong

In mobile ad hoc networks (MANETs), the more applications we use, the more security is required. In this paper, we propose a high secure and efficient routing scheme that not only satisfies the properties of anonymity, security, authentication, nonrepudiation, and unforgeability that the previous paper achieved for ad hoc networks, but also satisfies other necessary properties such as confidentiality, traceability, and flexibility for multipaths in order to make the ad hoc environment more secure and practicable.

1. Introduction

In the near future, wireless networks will play an impartment role in information communication and transmission. Compared to wired network environments, wireless networks are more convenient for users in that users can connect to the Internet with mobility. Once the foundation equipment has been established, the user can use the mobile devices such as PDAs or notebooks to access resources on the Internet anywhere. Many practical daily problems can be solved in the mobile environment—for example, finding locations, booking seats on a plane, or finding the shortest path to a destination. Users can obtain answers to questions quickly in the mobile environment through the use of mobile devices and wireless networks [1]. The more applications are found for wireless networks, the more security issues are being discussed for wireless networks. In 2013, this paper proposed an anonymous authentication scheme which ensures user unlinkability. It is impossible for attackers to know that particular sessions, which have already occurred several times, are originated from one same user [2]. The findings in this paper are useful for identifying the factors that must be managed for NFC-based mobile payment services [3, 4]. In 2013, the paper proposed an energy efficient method for clustering the nodes in the network [5]. The paper proposed an ETSI complaint geonetworking protocol layer and discussed the architecture of our implementation [6]. This paper proposed a network-based handover scheme for Host Identity Protocol in the mobile networks, in which the access routers of the mobile node will establish a handover tunnel and will perform the route optimization for data transmission [7]. Recently, this paper proposed the security based algorithmic approach in the mobile ad hoc networks [8, 9].

Routing is an important networking function in mobile ad hoc networks [10, 11]. Therefore, an enemy can collapse a network operation easily by attacking the routing protocol. Many researchers have proposed secure routing protocols for ad hoc networks [12–15]. The security of those protocols has been analyzed by informal means or formal methods that have never been intended for the analysis of this kind of protocol [16]. Other attacks can be found in [12]. There are two functions for routing: route discovery and packet forwarding. Route discovery is concerned with discovering routes between nodes, and packet forwarding is concerned with sending data packets through the previously discovered routes. There are different types of ad hoc routing protocols. One can distinguish proactive (e.g., OLSR [13]) and reactive (e.g., AODV [17] and DSR [18]) protocols.

In previous studies, wireless security studies [11, 19–22], researchers have mainly proposed that security issues be considered in the wireless network. These researchers have suggested that there are two kinds of attack behavior: (1) passive attack behavior [23] and (2) active attack behavior

[24]. Passive attacks involve attackers who do not transmit packets to attack the victim's computer but rather eavesdrop on messages sent to/from the victim's computer to affect the privacy and the anonymity between the sender and the receiver. Active attacks involve transmitting packets to attack and affect the operation of the victim's computer. From the viewpoint of the user, there are two kinds of denial-of-service (DoS): (1) routing-disruption attacks, in which the attacker attempts to forge the legal packet that is transmitted, and (2) resource-consumption attacks, in which the attacker transmits mass meaningless packets to occupy the user's bandwidth, waste memory and computing resources, and paralyze the service and operation ability of the computer.

2. Literature Review

Compared to wired networks, the wireless network is easy to attack because of its openness, its dynamic network topology, and its lack of the central monitoring and management. Security issues are becoming more and more important in wireless networks. Several wireless security studies [25–27] have discussed the properties of security, such as confidentiality, integrity, authentication, availability, fairness, anonymity, non-source-based routing, resilience against path hijacking, lack of source control over route length, and privacy. There are several properties of ad hoc routing security: reliability, confidentiality, integrity, and verification [28, 29].

Reliability. The attacker interferes with the physical layer and makes the data cannot be delivered. Or the attacker breaks down the network routing function and creates topology splits. Alternatively, the network could incur a DoS attack.

Confidentiality. Because the MANET was originally applied in military environments, not only the general information, but also the routing information must be kept confidential. If this is done, the enemy will be unable to find the target position through the routing information.

Integrity. To ensure that delivered data will not be forged or modified by the attacker, some secure methods can be applied to retain information integrity [30].

Verification. In MANETs, each node plays the role of routing the path or verifying the data. Because the data must be delivered by trusted nodes, it is very important to verify whether the nodes are trusted.

In Boukerche's scheme [31], an anonymous wireless protocol was proposed. The source generated a path discovery phase. The source sent a request packet with some information including a trust requirement, a one-time public/private key pair, and a destination identity. Each middle node had a mapping table to map the session and the session key that the node should use. If the middle node had received the request packet previously, it would ignore the packet. The node would decrypt the packet using the session key and forward the packet in the direction of the destination. Each middle node that received the packet would try to decrypt it using its own private key. Then it will forward the encrypted packets to its next hop.

Then the destination will collect all the identities and session keys, encrypt the information using each middle node's session key, and forward the message back to the source. In the path reverse phase, the middle nodes in the reverse path will receive the packets. After that, each middle node will decrypt the packet using its privacy key and then will obtain the temporal privacy key. The middle node will decrypt and obtain the identities session key and random numbers of all the middle nodes. Then the receiver will compose the messages from the destination to the source; this includes all the random numbers and session keys. All the information will be encrypted by each node's session key and operated using the next node's random number sequentially. Thus, each node on the reverser path will execute an exclusive-OR (XOR) operation with its random number and obtain the session key. After the source obtains all the messages from the middle nodes, it will generate a mapping table through the middle node identity and the random number of each middle node identity and then will transmit the mapping table to its next node in the reserve path. Then, in the data transfer phase, it will use a secure forwarding protocol such as Onion [22]. After the source obtains the reserve path message and verifies that the message is correct, it will encrypt the message using the middle nodes' session key sequentially. Each middle node simply needs to decrypt the message by its own session key and forward the message to its next node to the destination.

In this scheme, the packets are anonymous, secure, and private in the transmitting phase. The scheme can prevent the attacker from analyzing the network flow and provide the privacy of the sender and receiver. All the nodes can establish the location information and the anonymous routing paths by exchanging routing information. In the ad hoc environment, because of node mobility, it is difficult to establish all of the location information for all the nodes. Thus, the scheme proposes a distributed routing scheme to establish an anonymous routing path and achieve the security properties of non-source-based routing and resilience against path hijacking. But Boukerche's scheme still has some aspects that could be improved, such as efficiency. Boukerche's scheme uses lots of public key system operations, and this could lead to increased power consumption, wasted memory space, and increased operating time for each node. Wu et al.'s scheme [32] proposes a zone-based anonymous positioning routing protocol (ZAP) and includes three wireless transmitting systems with different degrees of anonymity. In this scheme, the client generates an anonymous zone (AZ) and sends a data request to the server. After the server receives the data request, it follows the three different anonymous systems to execute the wireless transmission. The following are the three anonymous wireless transmitting systems.

2.1. ZAP with Pseudo Destination (PD-ZAP). The destination will generate a pseudo destination (PD) randomly, not far from the destination. The PD location will be marked in the packet and sent to the server. The server then sends the data packets in the direction of the pseudo destination. Finally,

the data packets are sent to a node that is the closest node to the pseudo destination. Then the node is set as a proxy. The proxy broadcasts the data packets to its neighbors by its maximum transmission range. Because the real destination is not far from the pseudo destination, the real destination can receive the data packets.

2.2. Geocasting Anonymous Approach (G-ZAP). The distance between the destination and the pseudo destination will not be too far. The node chooses a circle area as the destination-anonymous zone (D-AZ). The server sends the data to the center of the D-AZ, and the first node to receive the data in the circle will be the proxy. The proxy will flood the data to each node in the D-AZ.

2.3. ZAP with Route Redundancy (RR-ZAP). As in the PD-ZAP, in the RR-ZAP, the destination generates a pseudo destination (PD) randomly, not far from the destination. But the distance between the destination and the pseudo destination, in this case, is kept for several hops, and the location of the destination is closer than that of the pseudo destination. Because Wu et al.'s scheme uses a DSR-like protocol [29], the server's data has to pass the real destination to the pseudo destination, and the real destination can thus obtain the data.

Wu et al.'s scheme can only achieve anonymity for a destination; the server cannot be anonymous. Additionally, the degree of anonymity in Wu et al.'s system depends on the node numbers in the anonymity zone (AZ). We can estimate the approximate location of the destination because the scheme uses the greedy geoforwarding protocol. Every forwarder can know the approximate location of the client. Lastly, Wu et al.'s scheme has only one single path; if the path is jammed, it will cause a transmission delay, which is not flexible.

Furthermore, there are some aspects of Boukerche's scheme that could be improved, such as efficiency. Boukerche's scheme expends a large amount of computing resources to the nodes for public key system processing, and this also costs memory and consumes power. In the next section, we will introduce a new wireless ad hoc scheme with both security and efficiency properties. This scheme is adaptive to the real wireless environment.

3. A High Secure and Efficient MANET Routing Scheme

In this section, a new wireless ad hoc scheme with both security and efficiency properties will be proposed. This scheme is adaptive to the real wireless environment. Our scheme not only satisfies the requirements of previous schemes, such as security, authentication, unforgeability, and nonrepudiation, but also includes source anonymity, destination anonymity, middle node anonymity, confidentiality, traceability, and flexibility for multipaths. It also offers improved efficiency in order to make the wireless environment more practical. There are four phases in the proposed secure routing scheme: (1) the transmitting request phase, (2) the request reply phase, (3) the data transmitting phase, and (4) the data transmitted phase. The details of this proposed secure routing scheme follow.

3.1. Transmitting Request Phase. (i) $S \rightarrow$ All: $E_{R_{PK}}[Sign_{S_{SK}}[S_{id}, R_{id}, Time_s, Data_{size}, r_S], r_S \oplus Session_{SR}, r_S \oplus SR_{SK}]$, $Tran_{id}$, SR_{PK}. Firstly, the sender S broadcasts the packets to others in the ad hoc networks to announce that he wants to transmit the data. The sender S then generates a shared session key $Session_{SR}$ for the receiver R, a shared pair public key SR_{PK} for the receiver R, a shared pair private key SR_{SK} for the receiver R, a random number r_S chosen by the sender S, the data transmitting serial number $Tran_{id} = r_S \oplus h(S_{id}, R_{id}, Time_s, Data_{size})$, and the identities of forwarding nodes on the sender's routing table $Node_{S1\_id} = h(r_S)$, $Node_{S2\_id} = h(2r_S)$, and $Node_{S3\_id} = h(3r_S)$.

Then, the sender S signs the sender's identity S_{id}, the receiver's identity R_{id}, the transmit requesting time $Time_S$, the data size $Data_{size}$, and the sender's random number r_S by the sender's private key S_{SK} and executes a XOR operation $\oplus$ with the session key $Session_{SR}$ and the shared secret key SR_{SK}. All the messages above would be encrypted by the receiver's public key R_{PK}. Then the data is added by transmitting the serial number $Tran_{id}$ and the shared public key SR_{PK}. All the messages are broadcasted throughout the wireless ad hoc networks. We would replace $E_{R_{PK}}[Sign_{S_{SK}}[S_{id}, R_{id}, Time_s, Data_{size}, r_S], r_S \oplus Session_{SR}, r_S \oplus SR_{SK}]$ with *Initial*.

(ii) $Node_1 \rightarrow Node_2$: $E_{SR_{PK}}[Node_{1\_id}, time_1, r_{N1}, Count_1]$, *Initial*, $Tran_{id}$, SR_{PK}. By broadcasting, a node $Node_1$ at one hub distance from the sender S uses the shared public key SR_{PK} to encrypt the identity number $Node_{1\_id}$ of the $Node_1$, the time the $Node_1$ receives the message $time_1$, $Node_1$'s random number r_{N1}, and the total forwarding time $Count_1$. The ciphertext above would add the original information: the data transmitting serial number $Tran_{id}$ and the shared public key SR_{PK}. It then broadcasts the message continually to the node $Node_2$ at two-hub distance from the sender S. We would replace the ciphertext $E_{SR_{PK}}[Node_{1\_id}, time_1, r_{N1}, Count_1]$ with $Node_{1\_info}$.

(iii) $Node_2 \rightarrow Node_3$: $E_{SR_{PK}}[Node_{2\_id}, time_2, r_{N2}, Count_2]$, *Initial*, $Tran_{id}$, SR_{PK}, $Node_{1\_info}$. Then, the node $Node_2$ would encrypt the identity number $Node_{2\_id}$ of the $Node_2$, the data received time $time_2$ of $Node_2$, the random number r_{N2} chosen by $Node_2$, and the total forwarding time $Count_2$. The messages above add the data transmitting serial number $Tran_{id}$, the shared public key SR_{PK}, and all the information about node1 $Node_{1\_info}$. Then the message would be broadcasted continually to the node $Node_3$ at three hubs away from the sender S. We would replace the ciphertext $E_{SR_{PK}}[Node_{2\_id}, time_2, r_{N2}, Count_2]$ with $Node_{2\_info}$.

(iv) $Node_n \rightarrow Node_R$: $E_{SR_{PK}}[Node_{n\_id}, time_n, r_{Nn}, Count_n]$, *Initial*, $Tran_{id}$, SR_{PK}, $Node_{1\_info}$, $Node_{2\_info}, \ldots, Node_{n-1\_info}$. Finally, the receiver R obtains the broadcasted message and decrypts the message *Initial* using the receiver's secret key R_{SK} to obtain the transmitting request message from the sender S. The receiver R then verifies whether the message $Sign_{S_{SK}}[S_{id}, R_{id}, Time_s, Data_{size}, r_S]$ was sent from

Table 1: The receiver's routing table.

	Routing_table						
Path	Message						
$Path_1$	r_{pseudo_11}	$r_{N3} \oplus Node_{2\_id}$	r_{pseudo_12}	$\cdots$	$r_{N2} \oplus Node_{1\_id}$	r_{pseudo_1n}	$r_{N1} \oplus Node_{S1\_id}$
$Path_2$	r_{pseudo_21}	$r_{N6} \oplus Node_{5\_id}$	$r_{N5} \oplus Node_{4\_id}$	$\cdots$	r_{pseudo_2n-1}	r_{pseudo_2n}	$r_{N4} \oplus Node_{S2\_id}$
$Path_3$	$r_{N10} \oplus Node_{9\_id}$	r_{pseudo_31}	$r_{N9} \oplus Node_{8\_id}$	$\cdots$	$r_{N8} \oplus Node_{7\_id}$	$r_{N7} \oplus Node_{S3\_id}$	r_{pseudo_3n}

the sender S using the sender's public key S_{PK} and obtains the sender's identity S_{id}, the receiver's identity R_{id}, the transmit requesting time $Time_s$, the data size $Data_{size}$, and the sender's random number r_S.

3.2. Request Reply Phase. After the receiver R obtains the sender's random number r_S, the receiver R would execute a XOR operation $\oplus$, respectively, with the messages $r_S \oplus Session_{SR} \oplus r_S = Session_{SR}$ and $r_S \oplus SR_{SK} \oplus r_S = SR_{SK}$ to get the shared secret key SR_{SK} and the session key $Session_{SR}$. The receiver R would decrypt the ciphertext including the information about the $Node_{1\_info}$, $Node_{2\_info}, \ldots, Node_{n\_info}$ by the shared secret key SR_{SK} in order to obtain the identity of each node $Node_{x\_id}$, the data transmitted time of each node $time_x$, the random number chosen by each node r_{Nx}, and the total forwarding time $Count_x$.

Receiver R would establish the different path in routing table from the sender S to the receiver R. The receiver R would decrypt $Node_{n\_info}$ in order to obtain the identity of each node $Node_{x\_id}$ and the total forwarding time $Count_x$ to know the paths from the sender S to the receiver R. Because the total forwarding time for each path would be different, the receiver R would choose the paths $Path_x$ that have the different nodes for each path (without the same node in two paths).

Then the total forwarding time $Count_x$ would be ordered from few to many and the sender S could consider some issues in order to decide to receive the data. Considering the specific number of hub-count, wanting to finish the transmitting as soon as possible, wanting to save power, or wanting to avoid the malicious middle node, the receiver R could choose a multipath. Considering reducing the traffic jam for a specific area/time slot in the ad hoc networks, the receiver R could choose a multipath.

Figure 1 shows the environment of the ad hoc networks. For the secure reason, they are assumed to choose the multipath because there are two paths for the fewest number, the fewest number is 3, of the total forwarding time $Count_x$:

$$Path_1: Node_{S\_id}\text{-}Node_{1\_id}\text{-}Node_{2\_id}\text{-}Node_{3\_id}\text{-}Node_{R\_id},$$
$$Path_2: Node_{S\_id}\text{-}Node_{4\_id}\text{-}Node_{5\_id}\text{-}Node_{6\_id}\text{-}Node_{R\_id}.$$

There is one path for the total forwarding time $Count_x$:

$$Path_3: Node_{S\_id}\text{-}Node_{7\_id}\text{-}Node_{8\_id}\text{-}Node_{9\_id}\text{-}Node_{10\_id}\text{-}Node_{R\_id}.$$

In order to avoid a malicious middle node to guess the hub-count range by the amounts of the value in the information path from the sender S to the receiver R, the pseudo values r_{pseudo_11} to r_{pseudo_3n} would be added to confuse the malicious middle node. The pseudo value could be added arbitrarily. In $S \rightarrow$ All, for avoiding the collusion of the middle nodes, the identity of each node would be replaced as follows:

$$Node_{S1\_id} = h(r_S),\ Node_{S2\_id} = h(2r_S),\ Node_{S3\_id} = h(3r_S).$$

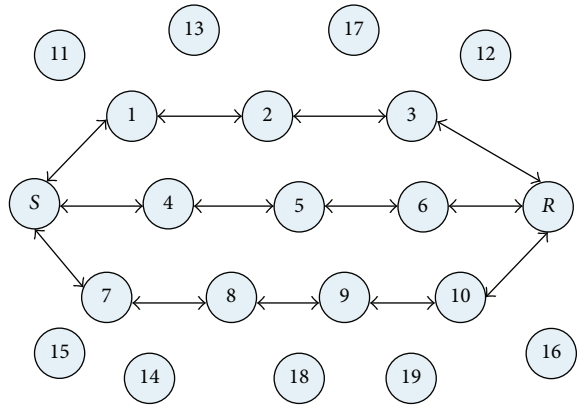

Figure 1: The environment of the ad hoc network.

Therefore, as even the middle nodes collude, it is impossible to map out the same identity for the middle node in order to protect the receiver R. In each path, the receiver R would, respectively, choose a random number r_{Nn} for his one-hub-count distance neighbor and a random number $Node_{n-1\_id}$ for his two-hub neighbor and then execute a XOR operation $\oplus$, for example, $r_{N3} \oplus Node_{2\_id}$. Therefore, when the message from the receiver R is sent back to the sender S, because the middle nodes know their chosen random number r_X, when the node executes a XOR $\oplus$ operation with the message and his random number r_X, the node can know what the identity is of the next node and forwards the message to the next node. For example, $r_{N3} \oplus Node_{2\_id} \oplus r_{N3} = Node_{2\_id}$. Table 1 shows the routing table of receiver *Routing_table* computed by the receiver R:

(i) $R \rightarrow Node_3$: $E_{Session_{SR}}[S_{id}, R_{id}, r_S, r_R, r_1, \ldots, r_{10}, Data_{size}, time_S, time_R, Routing\_table] \oplus r_S$, $r_{N3} \oplus [time_3, Tran_{id}, Path_1]$, $r_{N2} \oplus [time_2, Tran_{id}, Path_1]$, $r_{N1} \oplus [time_1, Tran_{id}, Path_1]$, $r_{pseudo\_41}, \ldots, r_{pseudo\_4n}$,

(ii) $R \rightarrow Node_6$: $E_{Session_{SR}}[S_{id}, R_{id}, r_S, r_R, r_1, \ldots, r_{10}, Data_{size}, time_S, time_R, Routing\_table] \oplus r_S$, $r_{N6} \oplus [time_6, Tran_{id}, Path_2]$, $r_{N5} \oplus [time_5, Tran_{id}, Path_2]$, $r_{N4} \oplus [time_4, Tran_{id}, Path_2]$, $r_{pseudo\_51}, \ldots, r_{pseudo\_5n}$,

(iii) $R \rightarrow Node_{10}$: $E_{Session_{SR}}[S_{id}, R_{id}, r_S, r_R, r_1, \ldots, r_{10}, Data_{size}, time_S, time_R, Routing\_table] \oplus r_S$, $r_{N10} \oplus$

$[time_{10}, Tran_{id}, Path_3]$, $r_{N9} \oplus [time_9, Tran_{id}, Path_3]$, $r_{N8} \oplus [time_8, Tran_{id}, Path_3]$, $r_{N7} \oplus [time_7, Tran_{id}, Path_3]$, $r_{pseudo\_61}, \ldots, r_{pseudo\_6n}$.

The receiver R would encrypt the sender's identity S_{id}, the receiver's identity R_{id}, the sender's random number r_S, the receiver's random number r_R, the data size $Data_{size}$, transmit requesting time $time_S$, data received time $time_R$, and the routing table of receiver $Routing\_table$ by the session key $Session_{SR}$ and would then execute a XOR operation $\oplus$ with the sender's random number r_S. In $Path_1$, the middle nodes would execute a XOR operation $\oplus$ with each node's data transmitted time $time_X$, the data transmitting serial number $Tran_{id}$, and the $Path_1$. Therefore, only the node could know these messages by its random number r_X and computes the next node in order to forward the message. $Path_1$ shows that the receiver R transmits the message to the $Node_3$ and the $Node_3$ could compute to obtain the message $[time_3, Tran_{id}, Path_1]$ by its random number r_{N3}. After the $Node_3$ confirms the data transmitted time $time_3$ and the data transmitting serial number $Tran_{id}$, it could execute a XOR operation $\oplus$ with the identity $Node_{2\_id}$ of $Node_2$ by $Node_3$'s random number r_{N3}:

$$r_{N3} \oplus Node_{2\_id} \oplus r_{N3} = Node_{2\_id}.$$

Then $Node_3$ would forward the message to $Node_2$. The pseudo values $r_{pseudo\_41}, \ldots, r_{pseudo\_4n}$ could be increased or decreased dynamically. The forwarding method for $Path_2$ and $Path_3$ is similar to that for $Path_1$. We would replace the message $E_{Session_{SR}}[S_{id}, R_{id}, r_R, r_S, r_1, \ldots, r_{10}, Data_{size}, time_S, time_R, Routing\_table] \oplus r_S$ with $Back$:

(i) $Node_3 \rightarrow Node_2$: $Back, r_{N3} \oplus [time_3, Tran_{id}, Path_1]$, $r_{N2} \oplus [time_2, Tran_{id}, Path_1]$, $r_{N1} \oplus [time_1, Tran_{id}, Path_1]$,

(ii) $Node_6 \rightarrow Node_5$: $Back, r_{N6} \oplus [time_6, Tran_{id}, Path_2]$, $r_{N5} \oplus [time_5, Tran_{id}]$, $r_{N4} \oplus [time_4, Tran_{id}, Path_2]$,

(iii) $Node_{10} \rightarrow Node_9$: $Back$, $r_{N10} \oplus [time_{10}, Tran_{id}, Path_3]$, $r_{N9} \oplus [time_9, Tran_{id}, Path_3]$, $r_{N8} \oplus [time_8, Tran_{id}, Path_3]$, $r_{N7} \oplus [time_7, Tran_{id}, Path_3]$.

After $Node_2$ receives the message from $Node_3$, $Node_2$ would do the same thing to confirm the data transmitted time $time_2$ and the data transmitting serial number $Tran_{id}$ and would then obtain the message identity $Node_{1\_id}$ of $Node_1$ in $Path_1$ by executing a XOR operation with its random number r_2 and sending the message to $Node_1$. The method for $Path_2$ and $Path_3$ is similar to that for $Path_1$:

(i) $Node_2 \rightarrow Node_1$: $Back, r_{N3} \oplus [time_3, Tran_{id}, Path_1]$, $r_{N2} \oplus [time_2, Tran_{id}]$, $r_{N1} \oplus [time_1, Tran_{id}, Path_1]$,

(ii) $Node_5 \rightarrow Node_4$: $Back, r_{N6} \oplus [time_6, Tran_{id}, Path_2]$, $r_{N5} \oplus [time_5, Tran_{id}, Path_2]$, $r_{N4} \oplus [time_4, Tran_{id}, Path_2]$,

(iii) $Node_9 \rightarrow Node_8$: $Back$, $r_{N10} \oplus [time_{10}, Tran_{id}, Path_3]$, $r_{N9} \oplus [time_9, Tran_{id}, Path_3]$, $r_{N8} \oplus [time_8, Tran_{id}, Path_3]$, $r_{N7} \oplus [time_7, Tran_{id}, Path_3]$.

After $Node_1$ receives the message from $Node_2$, $Node_1$ would do the same thing to confirm the data transmitted time $time_1$ and the data transmitting serial number $Tran_{id}$ and then obtains the message identity $Node_{S1\_id}$ of $Node_1$ in $Path_1$ by executing a XOR operation with its random number r_1 and sending the message to the sender S. The method for $Path_2$ and $Path_3$ is similar to that for $Path_1$:

(i) $Node_1 \rightarrow S$: $Back, r_{N3} \oplus [time_3, Tran_{id}, Path_1], r_{N2} \oplus [time_2, Tran_{id}, Path_1], Tran_{id}, Path_1$.

(ii) $Node_4 \rightarrow S$: $Back, r_{N6} \oplus [time_6, Tran_{id}, Path_2], r_{N5} \oplus [time_5, Tran_{id}, Path_2], Tran_{id}, Path_2$.

(iii) $Node_7 \rightarrow S$: $Back$, $r_{N10} \oplus [time_{10}, Tran_{id}, Path_3]$, $r_{N9} \oplus [time_9, Tran_{id}, Path_3]$, $r_{N8} \oplus [time_8, Tran_{id}, Path_3]$, $r_{N7} \oplus [time_7, Tran_{id}, Path_3]$.

After all the messages of $Path_1$, $Path_2$, and $Path_3$ have been received by the sender S, the sender S would execute a XOR operation $\oplus$ with the sender's random number r_S and $Back$ and then decrypt the ciphertext $E_{Session_{SR}}[S_{id}, R_{id}, r_S, r_R, r_1, \ldots, r_{10}, Data_{size}, time_S, time_R, Routing\_table]$ in order to obtain the sender's identity S_{id}, the receiver's identity R_{id}, the sender's random number r_S, the receiver's random number r_R, the random number of the middle nodes $r_1, \ldots, r_{10}$, the data size $Data_{size}$, the transmit requesting time $time_S$, the data received time $time_R$, and the routing table of receiver $Routing\_table$:

$$D_{Session_{SR}}[E_{Session_{SR}}[S_{id}, R_{id}, r_S, r_R, r_1, \ldots, r_{10}, Data_{size}, time_S, time_R, Routing\_table] \oplus r_S \oplus r_S] = S_{id}, R_{id}, r_S, r_R, r_1, \ldots, r_{10}, Data_{size}, time_S, time_R, Routing\_table.$$

(i) S: To Obtain the Information for Each Node, S Executes a XOR Operation with r_1 to r_{10}. The sender S would, respectively, execute the XOR operation $\oplus$ with the random number $r_1, \ldots,$ to r_{10}, for $Node_1$ and the message $r_{N1} \oplus Node_{S1\_id}$ on the routing table of the receiver's $Routing\_table$, for $Node_2$ and the message $r_{N2} \oplus Node_{S2\_id}, \ldots,$ to node 10 and the message $r_{N10} \oplus Node_{9\_id}$, in order to obtain the identity of each middle node. For example, $r_{N1} \oplus Node_{S1\_id} \oplus r_{N1} = Node_{S1\_id}, r_{N2} \oplus Node_{1\_id} \oplus r_{N2} = Node_{1\_id}$.

3.3. Data Transmitting Phase. S could establish the different routing paths for different requirements. The sender S would execute the one-way hash function for the $Node_{R1\_id} = h(r_R)$, $Node_{R2\_id} = h(2r_R)$, and $Node_{R3\_id} = h(3r_R)$. The same method that the receiver R uses in the request reply phase in order to avoid the guessing from the malicious middle node to know the possible range of the receiver R and the sender S and the pseudo values $r_{sending\_11}$ to $r_{sending\_3n}$ would be added to expand the guessing range. The pseudo value r_{pseudo} could be added dynamically. For avoiding the collusion of the middle nodes, the sender S would execute the one-way hash function for the $Node_{S1\_id} = h(r_S)$, $Node_{S2\_id} = h(2r_S)$, and $Node_{S3\_id} = h(3r_S)$.

Even though the middle nodes collude to guess for the sender S, they could not find the same identity for the sender S. In each path, the sender S would, respectively, choose a random number r_{Nn} for his one-hub-count distance

Table 2: The sender's routing table.

	Sending_table						
Path	Message						
$Path_1$	$r_{sending\_11}$	$r_{N1} \oplus Node_{2\_id}$	$r_{sending\_12}$	$\cdots$	$r_{N2} \oplus Node_{3\_id}$	$r_{sending\_1n}$	$r_{N3} \oplus Node_{R1\_id}$
$Path_2$	$r_{N4} \oplus Node_{5\_id}$	$\cdots$	$r_{sending\_21}$	$r_{N5} \oplus Node_{6\_id}$	$r_{sending\_2n-1}$	$r_{sending\_2n}$	$r_{N6} \oplus Node_{R2\_id}$
$Path_3$	$r_{sending\_31}$	$r_{N10} \oplus Node_{9\_id}$	$r_{N9} \oplus Node_{8\_id}$	$r_{N8} \oplus Node_{7\_id}$	$\cdots$	$r_{N7} \oplus Node_{R3\_id}$	$r_{sending\_3n}$

neighbor and a random number $Node_{n-1\_id}$ for his two-hub neighbor and would then execute a XOR operation $\oplus$, for example, $r_{N3} \oplus Node_{2\_id}$. Therefore, when the message from the sender S transmits to the receiver R, because all the middle nodes know their random number r_X, the middle nodes execute a XOR operation $\oplus$ with the message. Table 2 shows the routing table of sender *Sending_table* computed by the sender S. For example, $r_{N1} \oplus Node_{2\_id} \oplus r_{N1} = Node_{2\_id}$:

(i) $S \rightarrow Node_1$: $E_{Session_{SR}}[Data_{1\_id}, time_S, time_{data}, r_S, r_R, Tran_{id}] \oplus r_R$, $Path_1 \oplus r_{n2}$, $Path_1 \oplus r_{n1}$, $Path_1 \oplus r_{n3}$,

(ii) $S \rightarrow Node_4$: $E_{Session_{SR}}[Data_{5\_id}, time_S, time_{data}, r_S, r_R, Tran_{id}] \oplus r_R$, $Path_2 \oplus r_{n4}$, $Path_2 \oplus r_{n5}$, $Path_2 \oplus r_{n6}$,

(iii) $S \rightarrow Node_7$: $E_{Session_{SR}}[Data_{10\_id}, time_S, time_{data}, r_S, r_R, Tran_{id}] \oplus r_R$, $Path_3 \oplus r_{n7}$, $Path_3 \oplus r_{n8}$, $Path_3 \oplus r_{n9}$, $Path_3 \oplus r_{n10}$.

When the sender S starts to transmit the data by $Path_1$, it would encrypt $Data_{1\_id}$, the transmit requesting time $time_S$, the data transmitted time $time_{data}$, the sender's random number r_S, the receiver's random number r_R, and the data transmitting serial number $Tran_{id}$ by the session key $Session_{SR}$. The ciphertext above would execute a XOR operation $\oplus$ with the receiver's random number r_R and then the ciphertext would execute a XOR operation $\oplus$ with each node's random number r_x, respectively. Finally, all the messages would be transmitted to $Node_1$. $Node_2$ and $Node_3$ on $Path_1$ would do the same thing to execute the similar operation and forward the message to the receiver R. The nodes in $Path_2$ and $Path_3$ would do the same thing to execute the similar operation and forward the message to the receiver R.

(i) $R \rightarrow S$: $E_{Session_{SR}}[S_{id}, R_{id}, r_S, r_R, Tran_{id}, time_{data}, time_{resend}] \oplus r_S$. If the receiver R does not obtain all the data from the sender S, the receiver R would encrypt the sender's identity S_{id}, the receiver's identity R_{id}, the sender's random number r_S, the receiver's random number r_R, the data transmitting serial number $Tran_{id}$, the data transmitted time $time_{data}$, and the resend request $time_{resend}$ by the session key $Session_{SR}$ to request that the sender S resend the lacking data.

3.4. Transmitted Finish Phase

(i) $R \rightarrow S$: $E_{Session_{SR}}[Data_n, time_S, time_{data}, r_S, r_R, ACK] \oplus r_R, Path_x$. Finally, after all of the data has been transmitted to the receiver R, the receiver R would encrypt the $Data_n$, the transmit requesting time $time_S$, the data transmitted time $time_{data}$, the sender's random number r_S, the receiver's random number r_R, and the acknowledgment information ACK by the session key $Session_{SR}$. Then, the ciphertext above would execute a XOR operation with the receiver's random number r_R, added the path information $Path_x$, and transmitted it to the sender S to inform that the data transmitting is finished.

4. Security and Property Analysis

In this section, we analyze the property of security and the function for the proposed scheme.

4.1. Anonymity

4.1.1. Source Anonymity. In Boukerche's path discovery phase, the source ID had been encrypted by its session key and the source's session key had been encrypted by the destination's public key; only the destination can be decrypted and obtains the source's session key by its secret key and the destination can be decrypted and obtain the source ID by the source's session key. In the scheme of PD-ZAP, G-ZAP, and RR-ZAP, because the packets do not add the destination ID information and the node mobility property in a mobile ad hoc environment, even the proxy has changed, the broadcasting or flooding area has changed, and the destination area still can be found and the data can be obtained.

In the transmitting request phase of our scheme, because the sender's identity S_{id} is encrypted by the receiver's public key, only the receiver in $Node_n \rightarrow Node_R$ can decrypt the message by using the receiver's secret key. Therefore, the sender's identity S_{id} is anonymous. In the request reply phase, because the sender's identity S_{id} is encrypted by the session key $Session_{SR}$, in $R \rightarrow Node_3$, $R \rightarrow Node_6$, and $R \rightarrow Node_{10}$, and executed a XOR operation with the receiver's random number r_R, the sender's identity S_{id} is therefore anonymous.

4.1.2. Destination Anonymity. In the path discovery phase of Boukerche's scheme, the destination ID would be encrypted by the destination public key and the destination can decrypt and obtain the destination ID by using its secret key.

In the scheme of PD-ZAP, the closest node to the destination will be chosen as a proxy and it will broadcast the data to its neighbors. Because the destination is not far from the pseudo destination, the destination still can receive the data. But the malicious node can realize that the destination is near the proxy if it detects that a proxy broadcasts data.

For the G-ZAP scheme, the first node that receives the data will be the proxy in the destination-anonymous zone. The proxy will flood the data to every node in the anonymous

zone. Because the destination is not far from the pseudo destination, the malicious node will in the distance of Proxy's maximum transmitting range.

For the RR-ZAP scheme, the destination will generate a pseudo destination (PD) randomly and place itself in the middle between the source and the pseudo destination. The destination will keep several hops of distance from the pseudo destination. Wu et al.'s scheme uses the GPRS-likescheme as its routing protocol; therefore, when the source sends the data to the pseudo destination, the data must pass the real destination. So the malicious node can still know that the destination is in the path from the source to the pseudo destination.

In the transmitting request phase of our scheme, because the receiver's identity R_{id} is encrypted by the receiver's public key R_{PK}, only the receiver R in $Node_n \rightarrow Node_R$ can decrypt the message by using the receiver's secret key. Therefore, the receiver's identity R_{id} is anonymous. In the request reply phase, because the receiver's identity R_{id} is encrypted by the session key $Session_{SR}$, in $R \rightarrow Node_3$, $R \rightarrow Node_6$, and $R \rightarrow Node_{10}$, and has been executed by a XOR operation with the receiver's random number r_R, the receiver's identity R_{id} is therefore anonymous.

4.1.3. Middle Node Anonymity. In the path discovery phase of Boukerche's scheme, whatever node obtained the temporal secret key can decrypt the data forwarded from the other node. And the node will know that the data is coming from the destination and which nodes are the forwarding nodes.

In the PD-ZAP scheme, the middle node will choose a next forwarding node by exchanging the location information. And each node has the neighbors list so a node can obtain some information about other middle nodes.

The G-ZAP and RR-ZAP scheme are just like PD-ZAP and the middle node will obtain a neighbors list by exchanging the location information. So it can obtain some information of other middle nodes.

In $Node_1 \rightarrow Node_2$ and $Node_2 \rightarrow Node_3$, the middle nodes $Node_{1\_id}$ to $Node_{10\_id}$ use the common secret key to encrypt the data and only the source and the receiver in $Node_n \rightarrow Node_R$ have the common secret key SR_{SK}, so the middle nodes $Node_{1\_id}$ to $Node_{10\_id}$ are anonymous. For the identity $Node_{x\_id}$ of any middle node, besides the source and receiver, only the middle node has its own random number $Node_{x\_id}$, so the middle node can decrypt the $r_{Nn} \oplus [timen_n, Tran_{id}, Path_x]$—for example, $r_{N3} \oplus [time_3, Tran_{id}, Path_1]$.

4.2. Security. In Boukerche's scheme, PD-ZAP, G-ZAP, and RR-ZAP schemes all use both the symmetrical encryption system and asymmetrical encryption system. So they all are satisfied with the security property.

In $S \rightarrow$ All, $Node_1 \rightarrow Node_2$, $Node_2 \rightarrow Node_3$, and $Node_n \rightarrow Node_R$, the messages transmitted to each node are based on the public cryptosystem; in the equations $R \rightarrow Node_3$, $R \rightarrow Node_6$, $R \rightarrow Node_{10}$, $Node_3 \rightarrow Node_2$, $Node_6 \rightarrow Node_5$, $Node_{10} \rightarrow Node_9$, $Node_2 \rightarrow Node_1$, $Node_5 \rightarrow Node_4$, $Node_9 \rightarrow Node_8$, $S \rightarrow Node_1$, $S \rightarrow Node_4$, $S \rightarrow Node_7$, and $R \rightarrow S$, the messages transmitted to each node are based on the session key cryptosystem and random number r_{Nn}. Therefore, the property of the security of the messages could be satisfied.

4.3. Confidentiality. In the path discovery phase of Boukerche's scheme, the message is encrypted by the symmetric cryptosystem and asymmetric cryptosystem. In the path reverse phase and data transfer phase of Boukerche's scheme, the message is encrypted by the symmetric cryptosystem.

In the PD-ZAP, G-ZAP, and RR-ZAP schemes, the destination will generate a symmetric key first and encrypt the request message and the symmetric key with the server's public key. Then, the destination will send the encrypted message to the server. After the server obtains the symmetric key, all the messages will be encrypted with the symmetric key.

In our scheme, the equations $S \rightarrow$ All, $Node_1 \rightarrow Node_2$, $Node_2 \rightarrow Node_3$, and $Node_n \rightarrow Node_R$ are encrypted/decrypted by the symmetric cryptosystem, to ensure data confidentiality. The $R \rightarrow Node_3$, $R \rightarrow Node_6$, $R \rightarrow Node_{10}$, $Node_3 \rightarrow Node_2$, $Node_6 \rightarrow Node_5$, $Node_{10} \rightarrow Node_9$, $Node_2 \rightarrow Node_1$, $Node_5 \rightarrow Node_4$, $Node_9 \rightarrow Node_8$, $Node_1 \rightarrow S$, $Node_4 \rightarrow S$, and $Node_7 \rightarrow S$ are encrypted/decrypted by the session key cryptosystem, to ensure data confidentiality, and only node X and receiver R know the node's random number r_X, so only the previous node will know the identity $Node_{n\_id}$ for its next node on the forwarding path. Other nodes on the path cannot know the identity $Node_{n\_id}$. In $S \rightarrow$ All, the other nodes only know that there are some messages that need to be forwarded. By the receiver's routing table *Routing_table*, there are some extra pseudo values $r_{pseudoB}$, so the middle nodes do not know exactly how many middle nodes exist on the forwarding path, not to mention the identity of the sender and receiver.

4.4. Authentication. In the path discovery phase, the sender will encrypt the symmetric key with the receiver's public key and sends it to the receiver. After the receiver obtains the symmetric key, it can obtain and verify the identity of the sender. The message is signed by the sender's secret key, so the receiver can verify the message.

In the schemes of PD-ZAP, G-ZAP, and RR-ZAP, there is the HMAC from destination, and the server can verify all the data by the HMAC. Only the destination and server have the symmetric key, so if some packets can be decrypted by the symmetric key, each of them can confirm that the packets do come from the other.

In $S \rightarrow$ All, the message $[S_{id}, R_{id}, Time_s, Data_{size}, r_S]$ would be signed by the sender's private key S_{SK}; in the request reply phase, the receiver R could authenticate whether these messages are from the sender S by the sender's public key S_{PK}. The sender S could not deny the messages. Therefore, the property of nonrepudiation could be satisfied. In the request reply phase, because the receiver R has the receiver's routing table *Routing_table*, there are the random numbers $r_1, \ldots, r_{10}$ for each node in the message *Back*. Each node $Node_1$–$Node_{10}$ could not deny the messages for the sender S and the receiver R, and only the sender S and the receiver R have the session key $Session_{SR}$, the sender's identity S_{id}, the receiver's identity

R_{id}, the sender's identity r_R, the receiver's identity r_R, the transmit requesting time $time_S$, and the data received time $time_R$. Therefore, both the sender S and the receiver R could not deny the messages.

4.5. Traceability. In the path discovery phase of Boukerche's scheme, each middle node will transmit its own identity to receiver, so the receiver will know who the exact forwarding nodes between source and receiver are. In the path reverse phase and data transfer phase, the replied messages from the receiver do not include the identity of the middle nodes, but rather the session keys of the middle nodes, so the resource cannot know who the middle nodes are.

In the schemes of PD-ZAP, G-ZAP, and RR-ZAP, although the middle node has exchanged information with its neighbors and has a neighbors list, it does not transmit its own identity information when forwarding the data, so the server and destination cannot know who the middle nodes are.

In $S \rightarrow$ All, the sender's identity S_{id}, the receiver's identity R_{id}, the data request time $Time_S$, the data size $Data_{size}$, and the sender random number r_S will be signed by the sender's secret key S_{SK} and encrypted by the receiver's public key R_{PK}, so the receiver in the request reply phase can verify if the forwarding data in multipath from $Node_1 \rightarrow Node_2$, $Node_2 \rightarrow Node_3$, and $Node_n \rightarrow Node_R$ is correct or not.

4.6. Nonrepudiation. In Boukerche's phase of Boukerche's scheme, because the source has signed with the sender's secret key, the source cannot deny it does not send the data. The middle node also signs the data it has received before and forwards the data, so the middle node cannot deny that it does not receive the data. In the path reverse phase, because the source can obtain the encrypted session key and it sends the source session key to the receiver in the path discovery phase, the receiver cannot deny that it does not transmit the data, similarly to the middle nodes.

In the schemes of PD-ZAP, G-ZAP, and RR-ZAP, the destination has encrypted and sent the data to the server, but there is no identity of destination, destination signature, or identity-verified information, so the destination can deny that it does not receive the forwarding message. The middle node is only in charge of forwarding the data encrypted by the symmetric key, so the middle node can deny that it does not receive the forwarding data. The server encrypted by with the symmetric key and sent the data to the destination, so the server cannot deny that it does not receive the destination's request message and has forwarded the data to the destination.

In $S \rightarrow$ All, the message $[S_{id}, R_{id}, Time_s, Data_{size}, r_S]$ has been signed with the source's secret key. In the request reply phase, the receiver can verify if the data came from the sender with the sender's public key S_{PK}, and the sender cannot deny that it does not transmit the data. In the request reply phase, because the receiver has the receiver routing table and the back has a lot of middle nodes' random numbers r_1 to r_{10}, each middle node $Node_1$–$Node_{10}$ cannot deny that it does not forward the data for the source and receiver. Because of the session key $Session_{SR}$, sender's identity S_{id}, receiver's identity R_{id}, and the sender's random number r_R which only the sender and receiver can own, the sender and receiver cannot deny that they have not sent the messages.

Table 3: The comparison table of secure function.

	Boukerche et al. [31]	PD-ZAP	G-ZAP	RR-ZAP	Ours
Anonymity					
Source	○	※	※	※	○
Destination	○	△	△	△	○
Middle node	X	△	△	△	○
Security	○	○	○	○	○
Confidentiality	○	○	○	○	○
Authentication	○	○	○	○	○
Traceability	△	X	X	X	○
Nonrepudiation	○	△	△	△	○
Unforgeability	○	X	X	X	○
Uncounterfeit	○	X	X	X	○
Flexibility for multipaths	X	X	X	X	○

○: achieved, X: unachieved, △: partially achieved, and ※: not considered.

4.7. Unforgeability. In the path discovery phase of Boukerche's scheme, because the message has been signed by the sender's secret key and the message has been encrypted by the receiver's public key, the malicious node, the middle nodes, and the receiver cannot forge the message. The malicious node and the middle node obtain the temporal public key, encrypt the message with it, and attach the hash-verified data to the message, so the malicious node and other middle nodes cannot forge the message, for example, the identity ID and the key K. Because the middle node has attached the hash-verified data to the message in the path reverse phase, the malicious node and other middle nodes cannot forge the message, for example, the identity ID and the key K.

In the schemes of PD-ZAP, G-ZAP, and RR-ZAP, the destination does not attach any identity information, for example, the signature or identity.

The server sends the data to the destination encrypted with the symmetric key, so the server cannot deny that it does not receive the destination's request message and has sent to the destination.

In $S \rightarrow$ All, the messages $r_S \oplus Session_{SR}$, $r_S \oplus SR_{SK}$ and the sender's random numbers r_S have been signed by the sender's secret key S_{SK}, so the request reply phase cannot be forged; the nodes will generate a random number r_X depending on different source and destination, so each node's transmitting message cannot be forged. In the equations $S \rightarrow$ All, $Node_1 \rightarrow Node_2$, $Node_2 \rightarrow Node_3$, and $Node_n \rightarrow Node_R$, only the sender and receiver have the session key $Session_{SR}$, so, in the request reply phase, the encrypted value *Back* cannot be forged. If someone wants to replace or forge the value *Back*, the sender can send the sender's random

Table 4: The efficiency comparison for each scheme.

Phase	Boukerche et al. [31]	PD-ZAP	G-ZAP	RR-ZAP	Ours
3.1	$(3n+2)TB_{expB}$ $+(n+1)TB_{SSLB}$ $+TB_{hB}$	TB_{expB} $+TB_{SSLB}$ $+TB_{hB}$	TB_{expB} $+TB_{SSLB}$ $+TB_{hB}$	TB_{expB} $+TB_{SSLB}$ $+TB_{hB}$	$(n+2)TB_{exp}$ $+2TB_{\oplus B}$
3.2	$2TB_{expB}$ $+(2n+1)TB_{SSLB}$	TB_{expB} $+TB_{SSLBB}$	TB_{expB} $+TB_{SSLBB}$	TB_{expB} $+TB_{SSLBB}$	$(n+1)TB_{expB}$ $+(n+1)TB_{SSL}$ $+(2n+4)TB_{\oplus B}$
3.3	$2nTB_{SSLB}$	nTB_{SSLB}	nTB_{SSLB}	$n+TB_{SSLB}$	$(n+3)TB_{SSLB}$ $+(n+3)TB_{\oplus B}$
3.4	None	None	None	None	$TB_{SSLB}+TB_{\oplus B}$
Total cost	$(3n+4)TB_{expB}$ $+(5n+2)TB_{SSLB}$ $+TB_{hB}$	$2TB_{expB}$ $+(n+2)TB_{SSLB}$ $+TB_{hB}$	$2TB_{expB}$ $+(n+2)TB_{SSLB}$ $+TB_{hB}$	$2TB_{expB}$ $+(n+3)TB_{SSLB}$ $+TB_{hBB}$	$(2n+3)TB_{expB}$ $+(2n+5)TB_{SSLB}$ $+(3n+10)TB_{hB}$

TB_{expB}: one-time asymmetric encrypting/decrypting operation.
TB_{SSLB}: one-time symmetric encrypting/decrypting operation.
TB_{hB}: one-time one-way hash function operation.
$TB_{\oplus B}$: one-time exclusive-OR operation.

number r_S and the session key $Session_{SR}$ to know that the value *Back* has been replaced or forged.

4.8. Uncounterfeit. In the path discovery phase of Boukerche's scheme, because the sender's messages have been signed with its secret key and encrypted with the receiver's public key, the malicious node cannot counterfeit the sender. The middle node has signed the hash value with its secret key before it forwards the packets, so the malicious node cannot counterfeit the middle node. In the path discovery phase, because the receiver's and middle node's messages have a hash value and the middle node has signed the hash value before forwarding the message, the malicious node cannot counterfeit the middle nodes or the receiver.

In the schemes of PD-ZAP, G-ZAP, and RR-ZAP, the destination has encrypted the message with the server's public key and sent the message to the server, but there is no information about the destination-verifying message, signature, or identity. The malicious node can counterfeit the destination and request the server to broadcast messages into any area the malicious node wants.

4.9. Flexibility for Multipaths. In Boukerche's scheme, there is only one forwarding path, and it does not consider electricity consumption or the traffic-load balance by multipath. The schemes of PD-ZAP, G-ZAP, and RR-ZAP do not consider electricity consumption or the traffic-load balance by multipath but just forward the message with the GPRS [33]. In our scheme, after the receiver R decrypts the value $Node_{n\_info}$, the receiver R could know the identity number of the node $Node_{id}$ between the sender S and the receiver R and the total forwarding time $Count_x$ from different paths. Because the different path has the different total forwarding time $Count_x$, the receiver R would choose the path $Path_x$ with unique identity number of node $Node_{X\_id}$ and sort the order for each path $Path_x$ by the total forwarding time $Count_x$. Then, the sender S would decide to forward the data with single path or multipath with the consideration of the sender S. For example, if the sender S wants to finish the transition faster, saving each node's power, or to avoid a malicious node between the sender S and the receiver R, it would choose the multipath; if the sender S wants to reduce the traffic flow for the area/duration, the sender R would choose lesser paths or even single path. The comparison of secure functions is in Table 3.

5. Efficiency Analysis

Table 4 is the efficiency comparison table for each scheme. Boukerche's scheme and Wu et al.'s scheme do not consider the confirming process after the data transmission is finished. But, in our scheme, we have considered that. In Table 4, the efficiency in our scheme is better than that of Boukerche's. Although it is worse than Wu et al.'s scheme, but the efficiency and security could be tradeoff, and we think that security is more important. In the near future, with the improvement of CPU operation speed and the memory space, the operation speed for asymmetric encryption system could be decreased soon.

6. Conclusions

The proposed scheme is real and practical for the ad hoc environment. Its routing rule and transmission scheme are not only satisfied with the security properties of previous schemes, but also satisfied with the efficiency property. The proposed scheme is satisfied with the security properties of source anonymity, destination anonymity, middle node anonymity, security, confidentiality, authentication, traceability, nonrepudiation, unforgeability, uncounterfeit, and flexibility for multipaths.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] C. E. Perkins, "Ad hoc networking: an introduction," in *Ad hoc Networking*, pp. 1–28, 2001.

[2] Y. Chung, S. Choi, and D. Won, "Lightweight anonymous authentication scheme with unlinkability in global mobility networks," *Journal of Convergence*, vol. 4, pp. 23–29, 2013.

[3] S.-W. Park and I.-Y. Lee, "Anonymous authentication scheme based on NTRU for the protection of payment information in NFC Mobile environment," *Journal of Information Processing Systems*, vol. 9, no. 3, pp. 461–476, 2013.

[4] J.-S. Oh, C.-U. Park, and S.-B. Lee, "NFC-based mobile payment service adoption and diffusion," *Journal of Convergence*, vol. 5, pp. 8–14, 2014.

[5] A. Sinha and D. K. Lobiyal, "Performance evaluation of data aggregation for cluster-based wireless sensor network," *Human-Centric Computing and Information Sciences*, vol. 3, pp. 1–17, 2013.

[6] Z. C. Taysi and A. G. Yavuz, "ETSI compliant GeoNetworking protocol layer implementation for IVC simulations," *Human-centric Computing and Information Sciences*, vol. 3, pp. 1–12, 2013.

[7] M. Gohar and S.-J. Koh, "A network-based handover scheme in HIP-based mobile networks," *Journal of Information Processing Systems*, vol. 9, no. 4, pp. 651–659, 2013.

[8] B. Sharma and A. K. Singh, "A token based protocol for mutual exclusion in mobile ad hoc networks," *Journal of Information Processing Systems*, vol. 10, pp. 36–54, 2014.

[9] R. Singh, P. Singh, and M. Duhan, "An effective implementation of security based algorithmic approach in mobile adhoc networks," *Human-Centric Computing and Information Sciences*, vol. 4, article 7, 2011.

[10] A. Avižienis, J. C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 1, no. 1, pp. 11–33, 2004.

[11] M. N. Lima, A. L. D. Santos, and G. Pujolle, "A survey of survivability in mobile Ad hoc Networks," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 66–77, 2009.

[12] L. Buttyán and I. Vajda, "Towards provable security for Ad hoc routing protocols," in *Proceedings of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN '04)*, pp. 94–105, 2004.

[13] B. Kannhavong, H. Nakayama, Y. Nemoto, N. Kato, and A. Jamalipour, "SA-OLSR: security aware optimized link state routing for mobile ad hoc networks," in *Proceedings of the IEEE International Conference on Communications (ICC '08)*, pp. 1464–1468, May 2008.

[14] J. Kim and G. Tsudik, "SRDP: secure route discovery for dynamic source routing in MANETs," *Ad Hoc Networks*, vol. 7, no. 6, pp. 1097–1109, 2009.

[15] M. Yoon, Y.-K. Kim, and J.-W. Chang, "An energy-efficient routing protocol using message success rate in wireless sensor networks," *Journal of Convergence*, vol. 4, pp. 15–22, 2013.

[16] M. Burrows, M. Abadi, and R. M. Needham, "A logic of authentication," *Proceedings of the Royal Society of London Series A: Mathematical and Physical Sciences*, vol. 426, pp. 233–271, 1989.

[17] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '99)*, pp. 90–100, New Orleans, La, USA, Febuary 1999.

[18] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*, vol. 353 of *The Kluwer International Series in Engineering and Computer Science*, pp. 153–181, Springer, New York, NY, USA, 1996.

[19] F. Adelstein, S. K. S. Gupta, G. Richard, and L. Schwiebert, *Fundamentals of Mobile and Pervasive Computing*, McGraw-Hill, 2005.

[20] D. Djenouri, L. Khelladi, and N. Badache, "A survey of security issues in mobile ad hoc and sensor networks," *IEEE Communications Surveys and Tutorials*, vol. 7, no. 4, pp. 2–28, 2005.

[21] P. Papadimitratos and Z. Haas, "Handbook of ad hoc wireless networks," in *Securing Mobile Ad Hoc Networks*, CRC Press, 2002.

[22] M. G. Reed, P. F. Syverson, and D. M. Goldschlag, "Anonymous connections and onion routing," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 4, pp. 482–493, 1998.

[23] L. Zhou and Z. J. Haas, "Securing ad hoc networks," *IEEE Network*, vol. 13, no. 6, pp. 24–30, 1999.

[24] H. Yang, H. Luo, F. Ye, S. Lu, and L. Zhang, "Security in mobile ad hoc networks: challenges and solutions," *IEEE Wireless Communications*, vol. 11, no. 1, pp. 38–47, 2004.

[25] P. G. Argyroudis and D. O'Mahony, "Secure routing for mobile ad hoc networks," *IEEE Communications Surveys and Tutorials*, vol. 7, no. 3, pp. 2–21, 2005.

[26] R. B. Bobba, L. Eschenauer, V. Gligor, and W. Arbaugh, "Bootstrapping security associations for routing in mobile ad-hoc networks," in *Proceedings of the IEEE Global Telecommunications Conference*, pp. 1511–1515, December 2003.

[27] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Ariadne: a secure on-demand routing protocol for ad hoc networks," *Wireless Networks*, vol. 11, no. 1-2, pp. 21–38, 2005.

[28] B. Awerbuch, R. Curtmola, D. Holmer, C. Nita-Rotaru, and H. Rubens, "ODSBR: an on-demand secure Byzantine resilient routing protocol for wireless ad hoc networks," *ACM Transactions on Information and System Security*, vol. 10, no. 4, article 6, 2008.

[29] M. G. Zapata, "Secure ad hoc on-demand distance vector routing," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 6, no. 3, pp. 106–107, 2002.

[30] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[31] A. Boukerche, K. El-Khatib, L. Xu, and L. Korba, "An efficient secure distributed anonymous routing protocol for mobile and wireless ad hoc networks," *Computer Communications*, vol. 28, no. 10, pp. 1193–1203, 2005.

[32] X. Wu, J. Liu, X. Hong, and E. Bertino, "Anonymous geo-forwarding in MANETs through location cloaking," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, no. 10, pp. 1297–1309, 2008.

[33] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 243–254, ACM, August 2000.