This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2313180, IEEE Transactions on Vehicular Technology

AASR: Authenticated Anonymous Secure Routing for MANETs in Adversarial Environments

Wei Liu and Ming Yu

***Abstract*—Anonymous communications are important for many applications of the mobile ad hoc networks (MANETs) deployed in adversary environments. A major requirement on the network is to provide unidentifiability and unlinkability for mobile nodes and their traffics. Although a number of anonymous secure routing protocols have been proposed, the requirement is not fully satisfied. The existing protocols are vulnerable to the attacks of fake routing packets or denial-of-service (DoS) broadcasting, even the node identities are protected by pseudonyms. In this paper, we propose a new routing protocol, i.e., authenticated anonymous secure routing (AASR), to satisfy the requirement and defend the attacks. More specifically, the route request packets are authenticated by a group signature, to defend the potential active attacks without unveiling the node identities. The key-encrypted onion routing with a route secret verification message, is designed to prevent intermediate nodes from inferring a real destination. Simulation results have demonstrated the effectiveness of the proposed AASR protocol with improved performance as compared to the existing protocols.**

***Index Terms*—Anonymous Routing, Authenticated Routing, Onion Routing, Mobile Ad hoc Networks**

I. Introduction

Mobile ad hoc networks (MANETs) are vulnerable to security threats due to the inherent characteristics of such networks, such as the open wireless medium and dynamic topology. It is difficult to provide trusted and secure communications in adversarial environments, such as battlefields. On one hand, the adversaries outside a network may infer the information about the communicating nodes or traffic flows by passive traffic observation, even if the communications are encrypted. On the other hand, the nodes inside the network cannot be always trusted, since a valid node may be captured by enemies and becomes malicious. As a result, anonymous communications are important for MANETs in adversarial environments, in which the nodes identifications and routes are replaced by random numbers or pseudonyms for protection purpose.

Anonymity is defined as the state of being unidentifiable within a set of subjects. In MANETs, the requirements of anonymous communications can be described as a combination of unindentifiability and unlinkability [1]. Unindentifiability means that the identities of the source and destination nodes cannot be revealed to other nodes. Unlinkability means that the route and traffic flows between the source and destination nodes cannot be recognized or the two nodes cannot be linked. The key to implementing the anonymous communications is to develop appropriate anonymous secure routing protocols.

There are many anonymous routing protocols proposed in the past decade. Our focus is the type of topology-based on-demand anonymous routing protocols, which are general for MANETs in adversarial environments. To develop the anonymous protocols, a direct method is to anonymize the commonly used on-demand ad hoc routing protocols, such as AODV [2] and DSR [3]. For this purpose, the anonymous security associations have to be established among the source, destination, and every intermediate node along a route. The resulting protocols include ANODR [4], [5], SDAR [6], AnonDSR [7], MASK [8], [9], and Discount-ANODR [10].

After examining these protocols, we find that the objectives of unindentifiability and unlinkability are not fully satisfied. For example, ANODR focuses on protecting the node or route identities during a route discovery process, especially on the routing packets, e.g., Route REQuest (RREQ) and Route REPly (RREP). ANODR adopts a global trapdoor message in RREQ, instead of using the ID of the destination node. However, the route can be identified by a disclosed trapdoor message, which may be released to the intermediate nodes in backward RREP forwarding. The other protocols rely on the neighborhood detection and authentication, but may partially violate the anonymity requirements for performance considerations. For example, in SDAR, the node and its one-hop neighbors are made to know each other's ID during the routing procedures. In AnonDSR, the intermediate nodes en route may be revealed to the destination node. In MASK and Discount-ANODR, a clear node ID is used in the route discovery.

These protocols are also vulnerable to the denial-of-service (DoS) attacks, such as RREQ based broadcasting. Due to the lack of packet authentication, it is difficult for the protocols to check whether a packet has been modified by a malicious node. Recently, group signature is introduced to anonymous routing. In A3RP [11], the routing and data packets are protected by a group signature. However, the anonymous route is calculated by a secure hash function, which is not as scalable as the encrypted onion mechanism.

In this work, we focus on the MANETs in adversarial environments, where the public and group key can be initially deployed in the mobile nodes. We assume that there is no online security or localization service available when the network is deployed. We propose an authenticated anonymous secure routing (AASR) to overcome the pre-mentioned problems. We

This work has been in part supported by the Nature Science Foundation of China (NSFC) under Grant No.61371080 and No.61301127.

W. Liu is with the Dept. of Electronics & Information Engineering, Huazhong Univ. of Sci. & Tech., Wuhan 430074, China. (e-mail: liuwei@hust.edu.cn).

M. Yu is with the Dept. of Electrical & Computer Engineering, Florida State Univ., Tallahassee, FL 32310, U.S. (e-mail: mingyu@eng.fsu.edu).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2313180, IEEE Transactions on Vehicular Technology

adopt a key-encrypted onion to record a discovered route and design an encrypted secret message to verify the RREQ-RREP linkage. Group signature is used to authenticate the RREQ packet per hop, to prevent intermediate nodes from modifying the routing packet. Extensive simulations are used to compare the performance of AASR to that of ANODR, a representative on-demand anonymous routing protocol. The results show that, it provides more throughput than ANODR under the packet-dropping attacks, although AASR experiences more cryptographic operation delay.

The remainder of this paper is organized as follows. The background and related work of ad hoc anonymous routing are introduced in Section II. The network scenario is discussed in Section III. The design of AASR protocol is presented in Section IV. We evaluate AASR in Section V and provide the simulation results in Section VI. Section VII concludes this paper.

II. Background and Related Work

In this section, we introduce the basic concepts in anonymous routing, and provide a short survey on the existing routing protocols.

A. Anonymity and Security Primitives

We introduce some common mechanisms that are widely used in anonymous secure routing.

1) Trapdoor: In cryptographic functions, a trapdoor is a common concept that defines a one-way function between two sets [12]. A global trapdoor is an information collection mechanism in which intermediate nodes may add information elements, such as node IDs, into the trapdoor. Only certain nodes, such as the source and destination nodes can unlock and retrieve the elements using pre-established secret keys. The usage of trapdoor requires an anonymous end-to-end key agreement between the source and destination.

2) Onion Routing: It is a mechanism to provide private communications over a public network [13]. The source node sets up the core of an onion with a specific route message. During a route request phase, each forwarding node adds an encrypted layer to the route request message. The source and destination nodes do not necessarily know the ID of a forwarding node. The destination node receives the onion and delivers it along the route back to the source. The intermediate node can verify its role by decrypting and deleting the outer layer of the onion. Eventually an anonymous route can be established.

3) Group Signature: Group signature scheme [14] can provide authentications without disturbing the anonymity. Every member in a group may have a pair of group public and private keys issued by the group trust authority (i.e., group manager). The member can generate its own signature by its own private key, and such signature can be verified by other members in the group without revealing the signer's identity. Only the group trust authority can trace the signer's identity and revoke the group keys.

B. Anonymous On-demand Routing Protocols

There are many anonymous on-demand routing protocols. Similar to the ad hoc routing, there are two categories: topology-based and location-based [1], or in other words, node identity centric and location centric [15]. We compare the protocols in Table I, in terms of the key distribution assumption, node anonymity in route discovery, and packet authentication. Our observations are summarized as follows:

First of all, the routing protocols are designed to work in different scenarios. AO2P, PRISM, and ALERT are designed for location-based or location-aided anonymous communications, which require localization services. Since ours is for general MANETs, we focus on the topology-based routing rather than location-based routing.

Secondly, as mentioned in Section I, SDAR, AnonDSR, MASK, and D-ANODR have problems in meeting the unindentifiability and unlinkability. The node IDs in a neighborhood and along a route are possibly exposed in SDAR and AnonDSR, respectively. The plain node IDs are used in the route request of MASK and D-ANODR. In this work, we use the node's pseudonym instead of its real ID, to avoid the information leakage during RREQ and RREP processes.

Thirdly, some of the protocols adopt additional authentication schemes to sign the routing packets, including A3RP, RAODR [17], USOR [18], and PRISM [20]. Note that, although MASK provides neighborhood authentication, it cannot sign the routing packets. RAODR deploys a master key mechanism, which cannot provide the anonymity, traceability, and enforceability that are supported by a group signature. A3RP and USOR adopt a group signature and use secure hash functions to map the keys and node pseudonyms along a route. We choose the onion based routing to record the anonymous routes, because the onion is more scalable than other mechanisms and can be extended, for example to multiple paths.

Fourthly, we need to rethink the assumptions on the key distribution and node anonymity in route discovery. For example, ARM assumes that the source and destination nodes share a long-term session key in advance, which is not practical for real-world MANETs. We assume that the nodes are equipped with public and private keys during network initialization phase and can generate the shared symmetric key in an on-demand manner.

III. Network Scenario

In this section, we present our adversaries and attack models as well as the network assumptions and node model.

A. Adversaries and Attack Models

Without loss of generality, we assume that an adversary knows all the network protocols and functions. The attackers outside the network do not know the secret keys, but those inside the network may know the keys. We classify their attacks according to their behaviors (e.g., active or passive) and locations (e.g., inside or outside the network).

Passive outside attack: There may be an external global passive adversary, who can observe and record all the wireless communications in the network. It will try to reveal the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2313180, IEEE Transactions on Vehicular Technology

TABLE I
COMPARISON ON ANONYMOUS ON-DEMAND ROUTING PROTOCOLS

Protocol	Category	Info.known by Src.	Info. in route discovery	Authentication
ANODR [5]	Topology-based	Dest. trapdoor and public key	Route pseudonym	None
SDAR [6]	Topology-based	Dest. ID and public key	Dest. trapdoor	None
AnonDSR [7]	Topology-based	Node ID, Dest. public key	Dest. trapdoor	None
MASK [9]	Topology-based	Dest. ID	Dest. ID	None
D-ANODR [10]	Topology-based	Dest. ID and public key	Dest. ID	None
A3RP [11]	Topology-based	Dest. trapdoor and public key	Dest. trapdoor	Group signature
ARM [16]	Topology-based	Symm. key	Route pseudonym	None
RAODR [17]	Topology-based	Dest. ID	Dest.trapdoor	Symm. signature
USOR [18]	Topology-based	Dest. ID and public key	Dest.trapdoor	Group signature
AO2P [19]	Location-based	Dest. public key	Dest. position	None
PRISM [20]	Location-based	Group public key	Dest. area	Group signature
ALERT [21]	Location-based	Dest. public key	Dest. zone	None

identities of the source, destination, and en-route nodes of a particular flow, or infer the traffic flows by linking the packets to the source or destination nodes.

Active outside attack: The passive attackers avoid any attack that reveals their actions since they attempt to be invisible, but the active outside attackers do not have such restrictions. They may aim to disrupt the routing or launch a DoS attack. They can move from here to there and launch attacks randomly.

Passive inside attack: The attackers are legitimate MANET nodes. Similar to the passive outside attackers, they will try to infer the identities of the source, destination, or en-route nodes without exposing themselves. Since they can read the legitimate packets, the traffic pattern or node mobility information may be learned by them.

Active inside attack: They can modify, inject, and replay genuine messages. They can also masquerade as other nodes and launch the impersonation attacks. They can create one or more phantom nodes by generating valid routing packets.

B. Network Assumptions

We denote a MANET by $\mathbf{T}$ and make the following assumptions.

1) Public Key Infrastructure: Each node $\mathbf{T}$ initially has a pair of public/private keys issued by a public key infrastructure (PKI) or other certificate authority (CA). For node A ($A \in \mathbf{T}$), its public/private keys are denoted by K_{A+} and K_{A-}. Similar to the existing secure routing [22], we assume that there exists a dynamic key management scheme in $\mathbf{T}$, which enables the network to run without online PKI or CA services.

2) Group Signature: We consider the entire network $\mathbf{T}$ as a group and each node has a pair of group public/private keys issued by the group manager. The group public key, denoted by G_{T+}, is the same for all the nodes in $\mathbf{T}$, while the group private key, denoted by G_{A-} (for $A \in \mathbf{T}$), is different for each node. Node A may sign a message with its private key G_{A-}, and this message can be decrypted via the public key G_{T+} by the other nodes in $\mathbf{T}$, which keeps the anonymity of A [14]. We also assume that there exists a dynamic key management scheme working together with the admission control function of the network, which enables the group signature mechanism running properly. Such assumptions are also adopted in the existing work of military ad hoc networks [17], [23].

3) Neighborhood Symmetric Key: Any two nodes in a neighborhood can establish a security association and create a symmetric key with their public/private keys. This association can be triggered either by a periodical HELLO messages or by the routing discovery RREQ messages. For two nodes A and B ($A, B \in \mathbf{T}$), the shared symmetric key is denoted by K_{AB} and used for the data transmissions between them. There are some approaches supporting the establishment of one-hop shared key, such as MASK, RAODR, and USOR. In this work, we assume one of the approaches is available in $\mathbf{T}$.

The notations are summarized in Table II.

TABLE II
NOTATIONS FOR SECURITY PRIMITIVES

Notations	Descriptions
K_{A+}	Public key of node A
K_{A-}	Private key of node A
G_{T+}	Group public key of network $\mathbf{T}$
G_{A-}	Group private key of node A
K_{AB}	Symmetric key shared by nodes A and B
$\{d\}K_{A+}$	Data d is encrypted by key K_{A+}
$[d]K_{A-}$	Data d is signed by node A
$\langle d\rangle K_{AB}$	Data d is encrypted by shared key K_{AB}
$(d)K_A$	Data d is encrypted by one symm. key of A
$O_K(m)$	Encrypted onion for message m with key K
N_A	One-time Nym. generated by A to indicate itself
$dest$	A special bit-string tag denoting the destination

C. Node Model

1) Destination Table: We assume that a source node knows all its possible destination nodes. The destination information, including one of destination's pseudonym, public key, and the pre-determined trapdoor string $dest$ will be stored in the destination table. Once a session to the destination is established, the shared symmetric key is required for data encryptions in the session. Such symmetric key is generated by the source node before sending the route requests, and stored in the destination table after receiving the route reply. For example, a sample entry of the destination table is ($Dest\_Nym$, $Dest\_String$, $Dest\_Public\_Key$, $Session\_Key$).

2) Neighborhood Table: We assume that every node locally exchanges information with its neighbors. It can generate different pseudonyms to communicate with different neighbors. The neighbors security associations are established as well as the shared symmetric keys. The information is stored in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2313180, IEEE Transactions on Vehicular Technology

a neighborhood table. For example, a sample entry of the neighborhood table is ($Neighbor\_Nym$, $Session\_Key$).

3) Routing Table: When a node generates or forwards a route request, a new entry will be created in its routing table, which stores the request's pseudonym and the secret verification message in this route discovery. Such an entry will be marked in the status of "pending". If an RREP packet is received and verified, the corresponding entry in the routing table will be updated with the anonymous next hop and the status of "active". Meanwhile, a new entry will be created in the node's forwarding table. For example, a sample entry of the routing table is (Req_Nym, $Dest\_Nym$, Ver_Msg, $Next\_hop\_Nym$, $Status$). Note that, to simplify the notation, we ignore the timestamp information of the entry in the table.

4) Forwarding Table: The forwarding table records the switching information of an established route. We adopt the per hop pseudonym as the identifier for packet switching, similar to the VCI (virtual channel identifier) in ATM networks. In each entry of the forwarding table, the route pseudonym is generated by the destination node, while the node pseudonyms of the previous and next hop are obtained after processing the related RREQ and RREP packets. For example, a sample entry of the forwarding table is (Rt_Nym, $Prev\_hop\_Nym$, $Next\_hop\_Nym$).

IV. Protocol Design

In this section, we present the design of AASR protocol. Considering the nodal mobility, we take the on-demand ad hoc routing as the base of our protocol, including the phases of route discovery, data transmission, and route maintenance. In the route discovery phase, the source node broadcasts an RREQ packet to every node in the network. If the destination node receives the RREQ to itself, it will reply an RREP packet back along the incoming path of the RREQ. In order to protect the anonymity when exchanging the route information, we redesign the packet formats of the RREQ and RREP, and modify the related processes.

As an example, we use a five-node network to illustrate the authenticated anonymous routing processes. The network is shown in Fig.1, in which the source node S discovers a route to the destination node D.

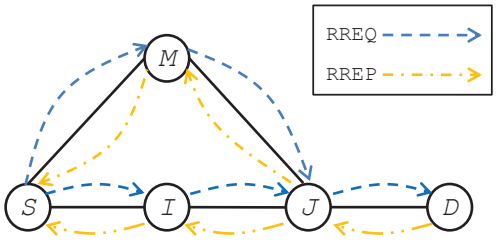


Fig. 1. Network topology

A. Anonymous Route Request

1) Source Node: We assume that S initially knows the information about D, including its pseudonym, public key, and destination string. The destination string $dest$ is a binary string, which means "You are the destination" and can be recognized by D. If there is no session key, S will generate a new session key K_{SD} for the association between S and D. The following entry will be updated in S's destination table.

Dest.Nym.	Dest.Str	Dest. Pub_Key	Session_Key
N_D	$dest$	K_{D+}	K_{SD}

Then, S will assemble and broadcast an RREQ packet in the format of (1). To simplify the notation, we ignore the timestamp information in the RREQ packet.

$$S \rightarrow *: [RREQ, N_{sq}, V_D, V_{SD}, Onion(S)]G_{S-} \quad (1)$$

where $RREQ$ is the packet type identifier; N_{sq} is a sequence number randomly generated by S for this route request; V_D is an encrypted message for the request validation at the destination node; V_{SD} is an encrypted message for the route validation at the intermediate nodes; $Onion(S)$ is a key-encrypted onion created by S. The whole RREQ packet is finally signed by S with its group private key G_{S-}.

The combination of V_D and V_{SD} works similarly to the global trapdoor used in ANODR. We introduce V_{SD}:

$$V_{SD} = (N_v)K_v \quad (2)$$

where N_v and K_v are two parameters created by S and sent to D for future route verification; N_v is a one-time nonce for the route discovery; and K_v is a symmetric key.

The secret message V_D is defined as:

$$V_D = \langle N_v, K_v, dest\rangle K_{SD}, \{K_{SD}\}K_{D+} \quad (3)$$

If D is the receiver of the message, D can decrypt the second part of V_D by its private key K_{D-}, and then decrypt the first part by the obtained K_{SD}. Otherwise, the receiver knows that it is not the intended destination.

If S and D have already established K_{SD} in a previous communication, the costly public encryption in the second part of V_D can be eliminated, and then V_D is defined as:

$$V_D = \langle N_v, K_v, dest\rangle K_{SD}, pad \quad (4)$$

where pad is a pre-defined bit-string that pads the message to a constant length.

V_{SD} and V_D are separated in the RREQ format (1). For a non-destination node, it can use V_{SD} as a unique identity for the route request.

Now we describe the encrypted onion $Onion(S)$. S creates the onion core as follow:

$$Onion(S) = O_{K_v}(N_S) \quad (5)$$

where N_S is a one-time nonce generated by S to indicate itself. The core is encrypted with the symmetric key of K_v, and can only be decrypted by D via K_v.

After sending the RREQ, S creates a new entry in its routing table, which looks like the following:

Req. Nym.	Dest.Nym.	Ver. Msg.	Next_hop	Status
N_{sq}	N_D	V_{SD}	N/A	Pending

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2313180, IEEE Transactions on Vehicular Technology

2) Intermediate Node: The RREQ packet from S is flooded in $\mathbf{T}$. Now we focus on an intermediate node I, as shown in Fig. 1. We assume that I has already established the neighbor relationship with S and J. I knows where the RREQ packet comes from. The following entries are stored in I's neighborhood table:

Neigh. Nym.	Session_Key
N_S	K_{SI}
N_J	K_{IJ}

Once I receives the RREQ packet, it will verify the packet with its group public key G_{T+}. As long as the packet is signed by a valid node, I can obtain the packet information. Otherwise, such an RREQ packet will be marked as malicious and dropped.

I checks the N_{sq} and the timestamp in order to determine whether the packet has been processed before or not. If the N_{sq} is not known in the routing table, it is a new RREQ request; if the N_{sq} exists in the table but with an old timestamp, it has been processed before and will be ignored; if the N_{sq} exists with a fresh timestamp, then the RREQ is a repeated request and will be recognized as an attack.

Then I tries to decrypt the part of V_D with its own private key. In case of decryption failure, I understands that it is not the destination of the RREQ. I will assemble and broadcast another RREQ packet in the following format:

$$I \rightarrow *: [RREQ, N_{sq}, V_D, V_{SD}, Onion(I)]G_{I-} \qquad (6)$$

where N_{sq} , V_D, and V_{SD} are kept the same as the received RREQ packet; the key-encrypted onion part is updated to $Onion(I)$. The complete packet is signed by I with its group private key G_{I-}.

I updates the onion in the following way:

$$Onion(I) = O_{K_{SI}}(N_I, Onion(S)) \qquad (7)$$

where N_I is a one-time nonce generated by I to indicate itself; $Onion(S)$ is obtained from the received RREQ packet; this layer of onion is encrypted with the symmetric key K_{SI}.

When I's RREQ reaches the next hop J, J will perform the same procedures and update the onion in the RREQ with one more layer, which is:

$$Onion(J) = O_{K_{IJ}}(N_J, Onion(I)) \qquad (8)$$

The routing tables of I and J will also be updated with a new entry as follow:

Req. Nym.	Dest.Nym.	Ver. Msg.	Next_hop	Status
N_{sq}	N/A	V_{SD}	N/A	Pending

3) Destination Node: When the RREQ packet reaches D, D validates it similarly to the intermediate nodes I or J. Since D can decrypt the part of V_D, it understands that it is the destination of the RREQ. D can obtain the session key K_{SD}, the validation nonce N_v, and the validation key K_v. Then D is ready to assemble an RREP packet to reply the S's route request.

B. Anonymous Route Reply

1) Destination Node: When D receives the RREQ from its neighbor J, it will assemble an RREP packet and send it back to J. The format of the RREP packet is defined as follow:

$$D \rightarrow *: (RREP, N_{rt}, \langle K_v, Onion(J)\rangle K_{JD}) \qquad (9)$$

where RREP is the packet type identifier; N_{rt} is the route pseudonym generated by D; K_v and $Onion(J)$ are obtained from the original RREQ and encrypted by the shared key K_{JD}. The intended receiver of the RREP is J.

2) Intermediate Node: We assume that J has already established a neighbor relationship with I, D, and M. The following entries are already in J's neighborhood table:

Neigh. Nym.	Session_Key
N_D	K_{JD}
N_I	K_{IJ}
N_M	K_{MJ}

If J receives the RREP from D, J will navigate the shared keys in its neighborhood table, and try to use them to decrypt $\langle K_v, Onion(J)\rangle K_{JD}$. In case of a successful decryption, J knows the RREP is valid and from N_D, and J also obtains the validation key K_v. Then J continues to decrypt the onion part. J knows the next hop for the RREP is N_I.

Then J will verify the linkage of the received RREP with its stored RREQ. It tries to use the obtained K_v to decrypt the verification message V_{SD} stored in its routing table. Once J finds the matched V_{SD}, it will update the corresponding routing entry as follows:

Req. Nym.	Dest.Nym.	Ver. Msg.	Next_hop	Status
N_{sq}	N/A	V_{SD}	N_D	Active

Since N_v in V_{SD} is not issued by J, J is not the source of the RREQ, then it has to assemble another RREP and forward it. The format of J's RREP towards the previous hop I is defined as:

$$J \rightarrow *: (RREP, N_{rt}, \langle K_v, Onion(I)\rangle K_{IJ}) \qquad (10)$$

where N_{rt} and K_v are obtained from the received RREP; $Onion(I)$ is obtained by from the decrypted $Onion(J)$; the shared key K_{IJ} is obtained from J's neighborhood table. The intended receiver of the RREP is I.

When the RREP packet travels according to the layers on the onion, it will start at the destination node and move back to its previous node. Each time the intermediate node can associate a value with the underlying wireless link on which the RREP travels, until the RREP packet reaches the source. In our protocol, every node records the one-time link pseudonyms announced by its neighbor node. Then the intermediate nodes' forwarding tables can be established after the RREP's trip.

Now we discuss the forwarding table in detail. After J updates its routing table, it will also create a new entry in its forwarding table. It may record the multiple paths found in the route discovery. According to the topology in Fig. 1, J's forwarding table may look like the following, in which $N_{X,i}$ stands for the ith one-time pseudonyms issued by node X:

D issues different pseudonyms $N_{D,1}$ and $N_{D,2}$ to J. There are two forwarding relationships at J. $N_{I,1} : N_{D,1}$ and $N_{M,1}$:

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2313180, IEEE Transactions on Vehicular Technology

Rt. Nym.	Pre_hop Nym.	Next_hop Nym.
$N_{rt,1}$	$N_{I,1}$	$N_{D,1}$
$N_{rt,2}$	$N_{M,1}$	$N_{D,2}$

$N_{D,2}$ describe the two routes of $I-J-D$ and $M-J-D$, as shown in Fig. 1. It can be seen that the forwarding table is made anonymous to any nodes, except for the switching node that owns the table. At the time of being anonymized, the switching relationship at each node en route can also be guaranteed.

3) Source Node: When the RREP packet reaches S, S validates the packet in a similar process to the intermediate nodes. If the decrypted onion core N_S equals to one of S's issued nonce, S is the original RREQ source. S will update its routing table as follow:

Req. Nym.	Dest.Nym.	Ver. Msg.	Next_hop	Status
N_{sq}	N_D	V_{SD}	N_I	Active

Then the route discovery process ends successfully. S is ready to transmit a data along the route indicated by N_{rt}.

C. Anonymous Data Transmission

Now S can transmit the data to D. The format of the data packet is defined as follows:

$$S \rightarrow D : (DATA, N_{rt}, \langle P_{data} \rangle K_{SD}) \tag{11}$$

where $DATA$ is the packet type; N_{rt} is the route pseudonym that can be recognized by downstream nodes; the data payload is denoted by P_{data}, which is encrypted by the session key K_{SD}.

Upon receiving a data packet, every node will look into its forwarding table. If N_{rt} in the data packet matches one entry in forwarding table, the node will forward the packet to the anonymous next hop. Otherwise, the data packet will be discarded. Following the similar mechanism as the VCI in ATM network, the data packet can be switched along the route until it arrives at the destination.

D. Routing Procedure

The routing algorithm can be implemented based on the existing on-demand ad hoc routing protocol like AODV or DSR. The main routing procedures can be summarized as follows:

1) During route discovery, a source node broadcasts an RREQ packet in the format of (1).
2) If an intermediate node receives the RREQ packet, it verifies the RREQ by using its group public key, and adds one layer on top of the key-encrypted onion, as (7). This process is repeated until the RREQ packet reaches the destination or expired.
3) Once the RREQ is received and verified by the destination node, the destination node assembles an RREP packet in the format of (9), and broadcasts it back to the source node.
4) On the reverse path back to the source, each intermediate node validates the RREP packet of (2) and updates its routing and forwarding tables. Then it removes one layer on the top of the key-encrypted onion, and continues broadcasting the updated RREP in the format of (10).
5) When the source node receives the RREP packet, it verifies the packet, and updates its routing and forwarding tables. The route discovery phase is completed.
6) The source node starts data transmissions in the established route in the format of (11). Every intermediate node forwards the data packets by using the route pseudonym.

V. Protocol Evaluation

In this section, we check whether AASR can achieve the anonymity goals and defend the above mentioned attacks.

A. Anonymity Analysis

We check three types of anonymities of AASR, namely identity anonymity, route anonymity, and location anonymity. In the anonymity analysis, we assume that all the nodes, including those on the discovered route, are potential adversaries and interested in the privacy information about the two communication parties that discover the route.

1) Identity Anonymity: Our routing protocol can function without using the nodes' identities. All the nodes generate random nonce to indicate themselves. Consequently, any intermediate or adversary nodes cannot acquire the identities of the source and destination nodes. Besides the tropdoor information $dest$ in the RREQ packet, there is no identity-related information involved in routing and forwarding processes.

However, in ANODR, $dest$ is also used in the RREPs backward forwarding. An intermediate malicious node can use it to infer the destination. In AASR, we adopt an encrypted secret V_{SD} as the verification message in the RREP phase. Although N_v and K_v will be known by the intermediate nodes in route discovery, they are not related to the destination's identity. Thus, AASR provides better unindentifiability and unlinkability than ANODR.

2) Route Anonymity: During the route discovery, the source, intermediate, and destination nodes only have information about the nodes' pseudonyms of the previous and next hop. Even if a node participates in route discovery, it has no idea about the entire route, neither an exterior adversary. Since the nonce of destination node is one-time randomly generated and only known by its neighborhood, it is hard for the cooperative and malicious nodes to infer the multi-hop route.

3) Location Anonymity: The packet format of AASR does not include any information related to the network topology and the number of participating nodes (such as TTL and sequence). Thus the inside malicious node cannot infer the network topology.

One potential problem of our protocol is that the size of the key-encrypted onion may increase with the number of hops along the RREQs broadcasting path. By assuming a maximum number of hops, and fixed message size, and random TTL technique [11], [16], such problem can be resolved. Due to the space limit, we do not present the details here. With the deployment of the technique, the external malicious node cannot infer the hop count by observing the packet size.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2313180, IEEE Transactions on Vehicular Technology

B. Security Analysis

1) Passive Attacks: One type of passive attacks is a global eavesdropper. As discussed in the previous section, it is impossible for an eavesdropper to obtain the identity information about the source or destination node in any communication session in AASR.

Another type of passive attack is the silent dropping, which means the adversaries or selfish nodes silently refuse to perform the requested functions in the protocol. In normal routing protocols, the watchdog model can be used to detect such actions. However, in the anonymous mobile communication, it is hard to recognize the misbehavior of adversaries or selfish nodes. In AASR, this can be improved by introducing a node trust model [24].

2) Impersonation Attacks: Impersonation attacks can be launched by the inside attackers. For example, the RREQ packets may be read and modified in some anonymous routing protocols. While in AASR, any node without the group key cannot join the communications. Because the forgery of a group signature is computational infeasible, it is impossible for an adversary to modify the packets. Since the group signature is traceable, if a group manager is available in the network, the singer of the fake routing packet can be identified by the group manager with the group's master key.

3) DoS Attacks: DoS attacks aim to deplete the nodes' resources. If the attacks are launched by the outside adversaries not having the keys, the packets can pass the packet verification. Such DoS attacks have little threat on our protocol. If the attacks are launched by the inside adversaries, more damage will be caused. However, once an inside adversary does so, its behavior of sending a large amount of route requests can be detected by other nodes in its neighborhood. Such abnormal behavior will be reported to the group manager. Then the attacker will be identified by tracing its signature.

C. Cryptographic Overhead Analysis

In this section, we analyze the extra time in processing the routing packets due to the use of security mechanisms.

We assume that the shared key algorithm uses a keyed-hashed MAC such as MD5. We use the following symbols to denote the time consumed in security mechanisms:

- $R_{encrypt}$: Group public key encryption;
- $R_{decrypt}$: Group public key decryption;
- R_{sign}: Group private key signature;
- R_{verify}: Group private key verification;
- $E_{symm.}$: Symmetric key encryption or decryption;
- O_{nion}: One-layer onion construction or destruction.

The operation costs for different packet types in AASR can be summarized in Table III.

We use the cryptographic benchmarks on 1GHz Pentium III according to [25], [26], where $R_{encrypt} = 22ms$, $R_{decrypt} = 17ms$, $R_{verify} = 26ms$, and $R_{sign} = 26ms$. For the shared key operations such as MD5, it processes packets at a speed of $29.2Mbps$. For a typical message of $1KB$, it takes $E_{symm.} = 0.28ms$ to perform a hash operation. We assume that the data packet and onion packet are $4KB$, their processing time is $4E_{symm.}$. Thus the cryptographic operation time of the RREQ, RREP, and DATA for the source node in AASR protocol are $49.68ms$, $1.40ms$, and $1.12ms$, respectively.

TABLE III
CRYPTOGRAPHIC OPERATIONS PER PACKET AT EACH NODE

Type	Src. node	Inter. nodes	Dest. node
RREQ	R_{sign}, $R_{encrypt}$, $2E_{symm.}$, O_{nion}	R_{sign}, $R_{encrypt}$, O_{nion}	R_{verify}, $R_{decrypt}$, $2E_{symm.}$, O_{nion}
RREP	$E_{symm.}$, O_{nion}	$E_{symm.}$, O_{nion}	$E_{symm.}$, O_{nion}
DATA	$4E_{symm.}$	0	$4E_{symm.}$

VI. PERFORMANCE SIMULATION

We implement the proposed AASR protocol in ns-2 by extending the AODV module to support the cryptographic operations. We compare the performances of AASR to those of ANODR and AODV in various mobility and adversary scenarios.

A. Network Configurations

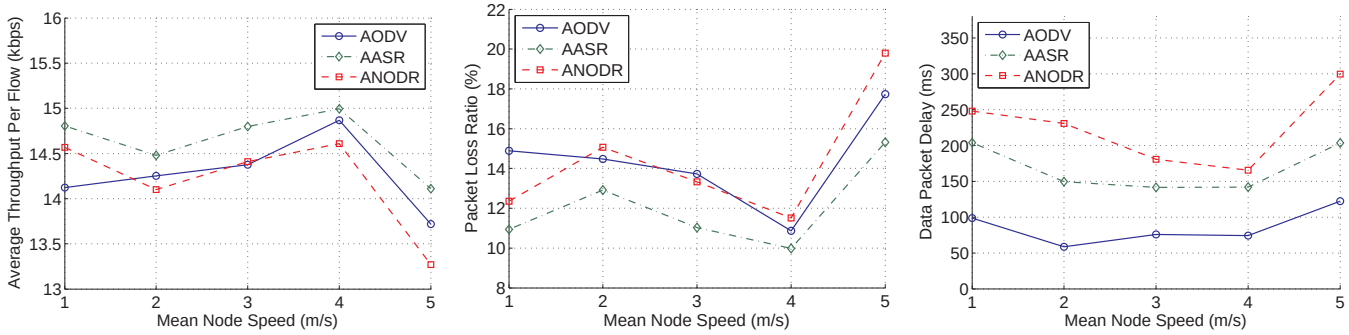
1) Topology and Traffic: In our simulations, the network area is $1200m \times 300m$ with 60 nodes initially and uniformly distributed. The distributed coordination function (DCF) of IEEE 802.11 is used as the MAC layer. The radio uses the two-ray ground reflection propagation model. The channel capacity is $2Mbps$. The transmission range is $150m$. The Random Way Point (RWP) model is used to model the nodal mobility. In our simulation, the mobility is controlled in such a way that the speed varies in the range of the minimum and maximum speeds. A total of 15 UDP based CBR sessions are used to generate the network traffic. For each session, the data packets are generated with the size of $512byte$ in the rate of $16Kbps$. The source-destination pairs are chosen randomly from all the nodes.

2) Attack Models: We assume that only the intermediate nodes along a route may become malicious. A malicious node will randomly drop routing packets. The packet dropping probability is varied from 0.1 to 0.5. The number of malicious nodes varies from 0 to 9. Such a random dropping attack is designed to simulate the effects of malicious attacks in different levels. For example, when being attacked by the fake routing packets in the impersonation attack, due to the lack of authentication, ANODR and AODV will suffer more packets losses than AASR. The attacks are simulated in the following way: AODV takes no action against the attack; ANODR acts in its routing maintenance procedures; AASR can detect the malicious node via the group signature, and get rid of the attackers in the routing tables.

B. Simulation Results

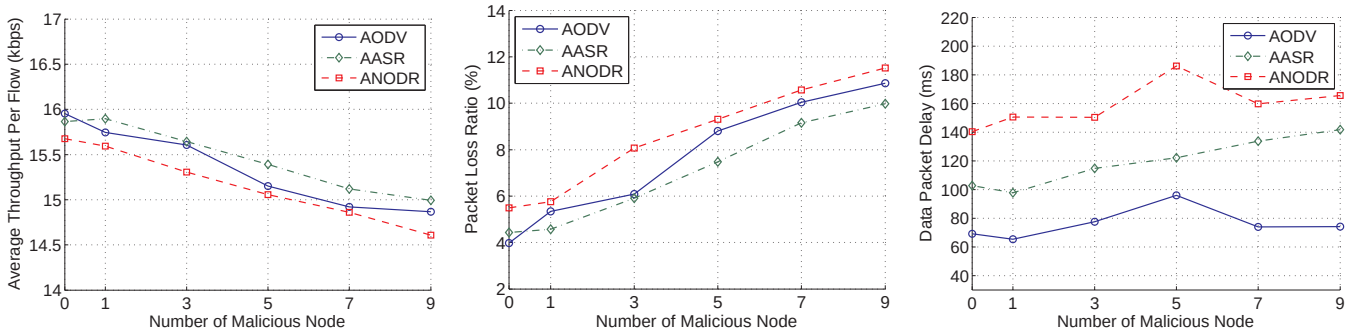
We present two groups of simulation results. The first one is to compare the routing performances of AODV, ANODR, and AASR under different mobility scenarios. The second one is to compare their behaviors under the packet dropping attacks with different levels. We perform five simulation runs

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2313180, IEEE Transactions on Vehicular Technology



(a) Per flow throughput (b) Packet loss ratio (c) End-to-end delay

Fig. 2. Performance comparison under different mobility settings



(a) Per flow throughput (b) Packet loss ratio (c) End-to-end delay

Fig. 3. Performance comparison in the presence of different numbers of malicious nodes

for each configuration, and record the per-flow performances, including throughput, packet loss ratio, and end-to-end delay. The average performance of different runs is presented as follows.

1) Group 1: The Effects of Mobile Scenario: To simulate the adversarial environments, we set 15% of the total nodes, i.e., 9 nodes, as malicious nodes. We change the network mobility from 1 to 5 m/s and record the performance results of the three protocols.

As shown in Fig. 2(a), when the average nodal speed increases, the throughput of the three protocols varies. Because the nodes move randomly, the throughput of CBR flow connections may be improved or degraded in different mobile topologies. Despite the performance variation, AASR always achieves the highest throughput. This can be explained by its ability in defending the packet dropping attack. Observing Figs. 2(a) and 2(b), we can see that the throughput and loss ratio achieved by ANODR and AODV are similar. Once being attacked, ANODR requires more cryptographic processing delays than the normal AODV protocol. As a result, sometimes ANODR performs worse than AODV, e.g., in the "slow" movement scenarios. The curves of the end-to-end delay are shown in Fig. 2(c). Due to the additional security processing time in RREQ flooding, ANODR and AASR have longer delays than AODV, while AASR has $50ms$ less of delay than ANODR in average.

2) Group 2: The Effects of Malicious Attacks: We configure the mobile network with an average speed of $4m/s$. We change the number of malicious nodes from 0 to 9. The results are recorded and plotted in the following figures.

As shown in Fig. 3(a), when the number of malicious nodes increases, the average throughput of three protocols decreases obviously. Since AASR has the ability to detect the packet dropping attack, it outperforms ANODR and AODV. Observing Fig. 3(b), we can see similar results. AASR achieves 2% less loss ratio than ANODR in average. The end-to-end delays are shown in Fig. 3(c). Since AODV is blind to the malicious attacks and takes no additional actions, its delay does not vary in the presence of different numbers of malicious nodes. Since AASR and ANODR spend time in the security processing in their route discovery, their delays are much higher than AODV. If ANODR is under a heavy attack, it will launch new route discoveries for the broken routes, which introduce more delays in average. Compared to the attacked ANODR, AASR reduces the need of re-routing, resulting in $30ms$ less of delay in average.

VII. Conclusion

In this paper, we design an authenticated and anonymous routing protocol for MANETs in adversarial environments. The route request packets are authenticated by group signatures, which can defend the potential active anonymous attacks without unveiling the node identities. The key-encrypted onion

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2014.2313180, IEEE Transactions on Vehicular Technology

routing with a route secret verification message is designed to not only record the anonymous routes but also prevent the intermediate nodes from inferring the real destination. Compared to ANODR, AASR provides higher throughput and lower packets loss ratio in different mobile scenarios in the presence of adversary attacks. It also provides better support for the secure communications that are sensitive to packet loss ratio.

In our future work, we will improve AASR to reduce the packet delay. A possible method is to combine it with a trust-based routing [24]. With the help of the trust model, the routing protocols will be more active in detecting link failures, caused either by the mobility or adversary attacks.

ACKNOWLEDGEMENT

The authors would like to thank the editors and reviewers for their constructional comments and suggestions that help improving the quality of the paper.

REFERENCES

[1] D. Kelly, R. Raines, R. Baldwin, B. Mullins, and M. Grimaila, "Towards a taxonomy of wired and wireless anonymous Networks," in *Proc. IEEE WCNC'09*, Apr. 2009.

[2] C. Perkins, E. Belding-Royer, S. Das, *et al.*, "RFC 3561 - Ad hoc On-Demand Distance Vector (AODV) Routing," *Internet RFCs*, 2003.

[3] D. Johnson, Y. Hu, and D. Maltz, "RFC 4728 - The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4," *Internet RFCs*, 2007.

[4] J. Kong and X. Hong, "ANODR: ANonymous On Demand Routing with Untraceable Routes for Mobile Ad hoc networks," in *Proc. ACM MobiHoc'03*, Jun. 2003, pp. 291–302.

[5] J. Kong, X. Hong, and M. Gerla, "ANODR: An identity-free and on-demand routing scheme against anonymity threats in mobile ad hoc networks," *IEEE Trans. on Mobile Computing*, vol. 6, no. 8, pp. 888–902, Aug. 2007.

[6] A. Boukerche, K. El-Khatib, L. Xu, and L. Korba, "SDAR: a Secure Distributed Anonymous Routing Protocol for Wireless and Mobile Ad hoc Networks," in *Proc. IEEE Int'l Conf. Local Computer Networks (LCN'04)*, Nov. 2004, pp. 618–624.

[7] R. Song, L. Korba, and G. Yee, "AnonDSR: efficient anonymous dynamic source routing for mobile ad hoc networks," in *Proc. ACM Workshop Security of Ad Hoc and Sensor Networks (SASN'05)*, Nov. 2005.

[8] Y. Zhang, W. Liu, and W. Lou, "Anonymous communications in mobile ad hoc networks," in *Proc. IEEE INFOCOM 2005*, vol. 3, Mar. 2005, pp. 1940–1951.

[9] Y. Zhang, W. Liu, W. Lou, and Y. G. Fang, "MASK: Anonymous On-Demand Routing in Mobile Ad hoc Networks," *IEEE Trans. on Wireless Comms.*, vol. 5, no. 9, pp. 2376–2386, Sept. 2006.

[10] L. Yang, M. Jakobsson, and S. Wetzel, "Discount anonymous on demand routing for mobile ad hoc networks," in *Proc. Int. Conf. on SECURECOMM'06*, Aug. 2006.

[11] J. Paik, B. Kim, and D. Lee, "A3RP: Anonymous and Authenticated Ad hoc Routing protocol," in *Proc. International Conf. on Information Security and Assurance (ISA'08)*, Apr. 2008.

[12] S. William and W. Stallings, *Cryptography and Network Security, 4th Edition*. Pearson Education India, 2006.

[13] M. G. Reed, P. F. Syverson, and D. M. Goldschlag, "Anonymous Connections and Onion Routing," *IEEE Journal on Selcted Area in Comm.*, vol. 16, no. 4, pp. 482–494, May 1998.

[14] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," in *Proc. Int. Cryptology Conf. (CRYPTO'04)*, Aug. 2004.

[15] K. E. Defrawy and G. Tsudik, "ALARM: Anonymous Location-Aided Routing in Suspicious MANETs," *IEEE Trans. on Mobile Computing*, vol. 10, no. 9, pp. 1345–1358, Sept. 2011.

[16] S. Seys and B. Preneel, "ARM: Anonymous Routing protocol for mobile ad hoc networks," *Int. Journal of Wireless and Mobile Computing*, vol. 3, no. 3, pp. 145–155, Oct. 2009.

[17] R. Song and L. Korba, "A robust anonymous ad hoc on-demand routing," in *Proc. IEEE MILCOM'09*, Oct. 2009.

[18] Z. Wan, K. Ren, and M. Gu, "USOR: An Unobservable Secure On-Demand Routing Protocol for Mobile Ad Hoc Networks," *IEEE Trans. on Wireless Communication*, vol. 11, no. 5, pp. 1922–1932, May. 2012.

[19] X. Wu and B. Bhargava, "AO2P: Ad Hoc On-Demand Position-Based Private Routing Protocol," *IEEE Trans. Mobile Computing*, vol. 4, no. 4, pp. 335–348, July/Aug. 2005.

[20] K. E. Defrawy and G. Tsudik, "Privacy-Preserving Location-Based On-Demand Routing in MANETs," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 10, pp. 1926–1934, Dec. 2011.

[21] H. Shen and L. Zhao, "ALERT: An Anonymous Location-based Efficient Routing Protocol in MANETs," *IEEE Trans. on Mobile Computing*, vol. 12, no. 6, pp. 1079–1093, 2013.

[22] M. Yu, M. C. Zhou, and W. Su, "A secure routing protocol against Byzantine attacks for MANETs in adversarial environment," *IEEE Trans. on Vehicular Tech.*, vol. 58, no. 1, pp. 449–460, Jan. 2009.

[23] X. Hong, J. Kong, Q. Zheng, N. Hu, and P. Bradford, "A Hierarchical Anonymous Routing Scheme for Mobile Ad-Hoc Networks," in *Proc. IEEE MILCOM'06*, Oct. 2006.

[24] M. Yu and K. Leung, "A Trustworthiness-based QoS routing protocol for ad hoc networks," *IEEE Trans. on Wireless Comms.*, vol. 8, no. 4, pp. 1888–1898, Apr. 2009.

[25] M. Brown, D. Hankerson, J. López, and A. Menezes, *Software implementation of the NIST elliptic curves over prime fields*. Springer, 2001.

[26] M. Scott, "Miracl – multiprecision integer and rational arithmetic c/c++ library," *Shamus Software Ltd, Dublin, Ireland*, 2003.

Wei Liu received the B.S. degree in Telecommunication Engineering in 1999 and Ph.D. in Electronics and Information Engineering in 2004, both from Huazhong University of Science and Technology, Wuhan 430074, China.

He is currently an associate professor with the Department of Electronics and Information Engineering, Huazhong University of Science and Technology. His research interests include ad hoc networks, content centric networks, and software defined networks.

Ming Yu received his Ph.D. from Rutgers University, New Brunswick, NJ, in 2002, and Doctor of Engineering from Tsinghua University, Beijing, in 1994, all in Electrical and Computer Engineering.

He joined AT&T in July 1997 as a Senior Technical Staff Member. During 2003 and 2006, he was with the Department of Electrical and Computer Engineering, State University of New York (SUNY) at Binghamton, NY. From August 2006, he joined the Department of Electrical and Computer Engineering, Florida State University, Tallahassee, FL. Currently he is an associate professor. His research interests include secure and reliable communications in various cyber-physical systems (CPS), from backbone/wireless networks, intelligent transportation systems (ITS) to smart grids. He was awarded an IEEE Third Millennium Medal by IEEE USA on May 2000.