

A Framework of Secure Location Service for Position-based Ad hoc Routing

Joo-Han Song, Vincent W.S. Wong, and Victor C.M. Leung

Department of Electrical and Computer Engineering

The University of British Columbia

2356 Main Mall, Vancouver, BC, Canada V6T 1Z4

e-mail: {jooahans, vincentw, vleung}@ece.ubc.ca

ABSTRACT

In large and dense mobile ad hoc networks, position-based routing protocols can offer significant performance improvement over topology-based routing protocols by using location information to make forwarding decisions. However, so far security issues in position-based routing protocols has not been widely considered. In this paper, we identify several security problems of position-based routing protocols in mobile ad hoc networks. To avoid these problems, we propose the Secure Grid Location Service (SGLS), which enhances the original GLS protocol with secure features. Countermeasures employed by SGLS against feasible attacks use both a broadcast authentication protocol and a reputation system for monitoring. Simulation results showed that SGLS can detect and isolate message dropping attackers efficiently.

Categories and Subject Descriptors

C.2.2 [Computer-Communication Networks]: Network Protocols – Routing Protocols

General Terms

Algorithms, Security

Keywords

Ad hoc Wireless Networks, Grid Location Service, Position-based Routing Protocol, Security

1. INTRODUCTION

Current research on Mobile Ad hoc NETWORK (MANET) has mainly focused on *topology-based* routing protocols, including both proactive and reactive (on-demand) approaches [1]. When the topology of the network changes frequently or the size of network increases, some of these protocols may incur a significant amount of routing control overhead. Recent research has shown that position-based routing protocols are good alternatives to topology-based routing protocols in large and dense MANETs [2]. Position-based routing protocols avoid the flooding of control traffic by using location information. For an intermediate node to

make a packet forwarding decision, it only needs to know its own position and the positions of its neighboring nodes to forward a packet to a neighbor geographically closest to the position of the packet's destination. To implement a position-based routing protocol, information about the physical location of each destination must be available. Each node can determine its own position using the Global Positioning System (GPS) [3]. In addition, a *location service* is used by the sender of a packet to determine the location of the destination. Each node may have a *location table* to store the position information of other nodes.

Most of the proposed ad hoc routing protocols assume that there is an implicit *trust-your-neighbor* relationship in which all the neighboring nodes behave properly. However, it is an undeniable fact that attackers do exist in real networks, and they may try to paralyze a MANET by manipulating the messages. Thus, a secure routing protocol is crucial. Recently, several secure ad hoc routing protocols have been proposed in the literature [4]-[6] with the aim of preventing various possible attacks. However, to the best of our knowledge, proposals on *secure position-based* routing protocols with location service are lacking. While position-based routing protocols do not store routing tables in nodes that are kept up-to-date via message exchanges, there are significant privacy and security concerns regarding their location service that integrates tracking and navigation capabilities.

In general, the attackers can be divided into two types: *malicious users* and *compromised users*. A malicious user is an outside attacker with no valid shared cryptographic key information. A compromised user, on the other hand, is an inside attacker who is behaving maliciously but is authenticated by the network and is being trusted by other users. A compromised user is capable of launching many kinds of attacks without being detected by other entities. Either malicious or compromised user can launch several attacks against the position-based routing protocols in MANETs. Some of these attacks include:

- A1. *Message tampering attack*: altering the content in any packet, e.g., (i) impersonating other nodes, and (ii) relaying or generating packets with altered contents.
- A2. *Message dropping attack*: intentionally dropping some (or all) control or data packets. Since nodes within a MANET function as both end hosts and routers, message dropping attacks can paralyze the network completely as the number of attackers increases.
- A3. *Message replay attack* (e.g., *wormhole* [7]): eavesdropping on the packets and replaying (or retransmitting) those packets again later.
- A4. *Location table tampering attack*: changing the information stored in the location table. This attack includes the physical

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

PE-WASUN'04, October 7, 2004, Venezia, Italy.

Copyright 2004 ACM 1-58113-959-4/04/0010...\$5.00.

deletion, alteration or falsification of information stored in location tables in a node.

- A5. *Blackmail attack*: causing the false identification of a good node as a bad one. This attack can occur in networks where the feedback of negative reputation is possible.

The location table tampering attack is feasible against any position-based routing protocols. Since compromised users are authenticated by the network and are being trusted by other users, it is possible for knowledgeable insiders [8] to modify the location tables without violating security associations within the MANET. As far as the other nodes are concerned, the location service is functioning normally. Therefore, this attack is difficult to detect. By manipulating the location information, attackers can prevent some of the existing routes from being used.

The objective of this paper is to provide security mechanisms for both data and control packets of position-based routing protocols in MANETs. The main contributions of this paper are as follows:

- (1) We propose the *Secure Grid Location Service (SGLS)*, which is a novel security extension to the original GLS protocol [9]. Our proposed SGLS has the capability of preventing message tampering and dropping attacks.
- (2) We present simulation results to show that in the presence of data and control packet dropping attackers, SGLS with greedy packet forwarding [10] continues to maintain a high packet delivery ratio at the expense of a slightly higher average end-to-end delay and routing overhead when compared to the original GLS protocol.

This paper is organized as follows. Section 2 provides some background. In Section 3, we present our proposed SGLS. The reputation system is described in Section 4. Section 5 explains the rationales of the assumptions we make in our framework. The performance comparisons between the original GLS and SGLS are presented in Section 6. Conclusions are given in Section 7.

2. BACKGROUND

In this section, we first provide an overview of GLS [9]. We then explain the basic concept of broadcast authentication, which is used for hop-by-hop authentication of messages. We also describe the use of a reputation system to detect and isolate message-dropping attackers.

2.1 Grid Location Service (GLS)

To find the current location of a specific node, a location service is necessary. Figure 1 shows the three primary location service entities in a general architecture:

- (1) *Location Generator (LG)* sends updates of its location to the location servers whenever necessary.
- (2) *Location Server (LS)* receives location update messages from LGs. It may also receive location queries from location recipients and relay them to the LGs.
- (3) *Location Recipient (LR)* sends a location query to an LS and receives location reply from an LG.

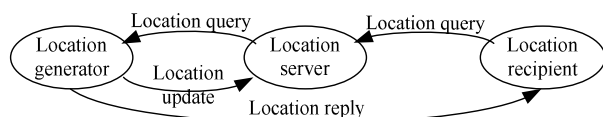


Figure 1. Location service entities

GLS [9] is a distributed location service that tracks the locations of destination nodes in a MANET. It divides the area into a hierarchy of squares. Four order- n squares make up an order- $(n+1)$ square, and so on. Each node periodically broadcasts a list of all neighbors using a HELLO message. Each entry in the table includes the node's unique ID, location, speed, and a timestamp.

A node maintains its current location in a number of LSs distributed throughout the network. Each node recruits three LSs in each level of the grid hierarchy. Node A 's strategy is to recruit nodes with IDs "close" to its own ID, over a mod 2^m ID space, to serve as its LSs (i.e., least ID greater than A). GLS makes greedy forwarding [10] decisions using information about both immediate neighbors and packets' destinations in the network.

To perform a location query, node B sends a request using geographic forwarding to the node with least ID greater than A for which B has location information. If this least ID node has no location information about A , it forwards the query in the same way, and so on. Eventually, the query will reach a LS of A , which forwards the query to A itself. Since the query contains B 's location, A can respond directly by using geographic forwarding. A node updates its order- i location servers every time it moves a distance $2^{i-2}d$ after the last update, where d is the threshold distance. When the route to a destination cannot be found, a location error message will be sent to the source node using greedy forwarding. The source node may then re-initiate the location query for that destination if needed.

2.2 Broadcast Authentication Protocol

Since malicious packet injection is possible in broadcast networks, a broadcast authentication protocol is required to enable the receivers to verify that the broadcast packets they received were actually sent by the claimed sender.

TESLA (Timed Efficient Stream Loss-tolerant Authentication) [11] applies the MAC¹ (Message Authentication Code) [12] to a message for broadcast authentication. Using MAC can provide secure authentication in point-to-point communication. For broadcast communications, however, multiple receivers need to know the MAC key for verification. Any receiver with the secret MAC key can forge data and impersonate the sender. TESLA provides key secrecy by using clock synchronization and delayed key disclosure. Each node chooses a random initial key K_n and generates a one-way key chain by repeatedly computing a one-way hash function h [13]. A node can compute any previous key K_j from a key K_i where $j < i$, by $K_j = h^{i-j}[K_i]$. To authenticate any received value on the one-way chain, a node applies this equation to the received value to determine if the computed value matches a previously known authentic key on the chain. Each node pre-determines a schedule at which it discloses each key of its one-way key chain, in the reverse order from generation.

TESLA with Instant Key disclosure (TIK) is an extension of TESLA, designed for use in MANETs [7]. For example, TIK can authenticate packets within the computation capacity of currently available pocket PCs [7]. The sender can disclose the key even in the same packet if all nodes have tightly synchronized clocks (e.g.,

¹ The acronym "MAC" refers to the Message Authentication Code. To avoid confusion, the term "Medium Access Control" is written out in full.

within 100 ns). This level of time synchronization can be achieved with off-the-shelf hardware based on GPS [3]. To schedule the key disclosure time within a packet's transmission, a minimum payload length is defined according to both the transmission rate and range of the physical layer. Moreover, when IEEE 802.11 Distributed Coordinated Function (DCF) with RTS/CTS [14] is used, the minimum packet size can be reduced by piggybacking the MAC in the RTS (Request-To-Send) frames.

2.3 Reputation System with Monitoring

The goal of a reputation system is to ensure that all participating nodes in a MANET are behaving properly. Each node is assumed to be able to listen to the transmissions of its neighbors in promiscuous mode [6]. Reports are exchanged between nodes. Misbehaving nodes are detected and punished.

The CONFIDANT (Cooperation Of Nodes: Fairness In Dynamic Ad hoc NeTworks) protocol [15] works as an extension to the reactive source-routing protocol to find the selfish and/or misbehaved nodes and to isolate them. Each node has a *monitor* for observations, a *reputation manager* for first-hand and trusted second-hand observations about routing and forwarding behavior of other nodes, a *trust manager* to control trust given to received warnings, and a *path manager* to adapt their behavior according to reputation and to take action against misbehaving nodes.

Each node estimates the probability of another node acting maliciously by inference from data obtained by direct or indirect observations. It uses the Bayesian approach [23] for reputation and trust representation, updates, and integration. Node i maintains a record of *first hand information* about node j in the form of $F_{ij} = (\alpha, \beta) = (\# \text{ of good behavior}, \# \text{ of bad behavior})$ and is initially set to $(1, 1)$. It gives less weight to observations in the past for reputation fading. The first-hand reputation rating is represented in the form of $FR_{ij} = \alpha/(\alpha + \beta)$ [23], which is initially set to 0.5. When node i receives the reported first-hand reputation information FR_{kj} about node j from node k , node i updates the *reputation rating* R_{ij} as follows: $R_{ij} = R_{ij} + \omega \cdot FR_{kj}$ where ω is a small constant. The trust threshold λ and the deviation threshold μ [23] are introduced to consider the trustworthiness of a reputation report. For example, when node i receives the reported first-hand information FR_{kj} from node k , node i updates its reputation rating R_{ij} only when either the trust value of node k is larger than the threshold λ , or the difference between FR_{ij} and FR_{kj} is less than the deviation threshold μ . Based on this deviation test, the trust rating of each node is updated in the same way as the reputation rating described above.

3. SECURE GRID LOCATION SERVICE (SGLS)

In this section, we describe our proposed SGLS protocol, which includes several security mechanisms. Our goal is to design efficient mechanisms that are robust against both malicious and compromised users. These mechanisms shall be sufficiently general to be applicable to a wide range of position-based routing protocols with location service.

3.1 Secure Geographic Forwarding

We use the following notations in this paper:

- (1) K_A^T denotes the public TESLA key of node A , and K_{AB} denotes the shared secret key between nodes A and B .

- (2) $MAC_{K_A^T}(M)$ denotes the computation of the MAC of message M with the TESLA key K_A^T using the keyed Hash-based MAC (HMAC) algorithm.²

We assume that source S and destination D share a secret key K_{SD} for message authentication. To convince the destination of the legitimacy of each field in a forwarded data packet, the sender simply includes a MAC computed over the non-mutable parts (e.g., timestamp or location information) with key K_{SD} . Therefore, the destination can easily verify the authenticity and freshness of the received data.

Since the intermediate nodes have not shared any secret key with the source, they cannot verify the non-mutable parts of the packet. However, each intermediate node needs to authenticate the previous node to avoid message tampering and replay attacks. We propose to use the TIK protocol for hop-by-hop authentication. We assume that a node can obtain a TIK public key of any other node, for example K_A^T , each receiver knows a key expiration interval I , and time is synchronized up to a maximum time synchronization error Δ . We also assume that every node has its own TESLA one-way key chain.

We now describe the geographic forwarding using TIK with MAC in detail. When a source node S forwards a message via its neighbor N to a destination D , the sender S sends the following message: $\langle MAC_{K_S^T}(M + MAC_{K_{SD}}(M)), MAC_{K_{SD}}(M), M, K_S^T \rangle$ where $MAC_{K_{SD}}(M)$ (or $MAC_{K_S^T}(M + MAC_{K_{SD}}(M))$) represents the MAC computed over message M (or $M + MAC_{K_{SD}}(M)$) computed using key K_{SD} (or K_S^T). The sender discloses the key K_S^T at the end of the same packet.

Figure 2 shows the timelines of sending and receiving a TIK packet between two neighbors, which are time synchronized within a maximum time synchronization error Δ . Time t_s indicates

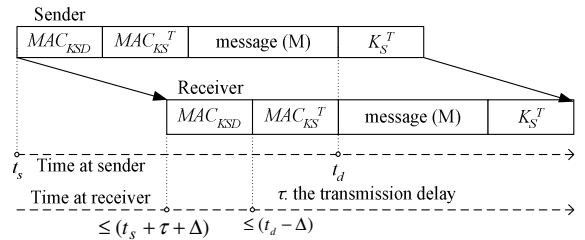


Figure 2. Geographic forwarding with TIK.

the time when sender S starts transmission of the packet, and time t_d is the disclosure time for key K_S^T . When the neighbor N receives the $MAC_{K_S^T}(M + MAC_{K_{SD}}(M))$, it verifies that the sender did not start

sending the corresponding key K_S^T yet, based on the timestamp t_i in message M and the synchronized clocks. If the sender has not yet started sending K_S^T , the receiver will verify that the key K_S^T at the end of the packet is authentic. When destination D receives this message, it can verify the authenticity of the message by comparing the received MAC value to the MAC value that is

² HMAC is a mechanism for message authentication using cryptographic hash functions, which can be used with any iterative cryptographic hash function, e.g., MD5 [13], in combination with a secret shared key.

computed for itself over the received message M with the secret key K_{SD} it shares with sender S .

Because of the time synchronization, the receiver can verify after receiving the packet that the key K_S^T used to compute the authentication has not yet been disclosed. As the receiver knows the expiration time for each key and the sender only discloses the key after it expires, no attacker can know K_S^T . Therefore, if the packet authentication verifies correctly once the receiver later receives the authentic key K_S^T , the packet must have originated from the claimed sender. Since only the sender knew the key K_S^T , at the time when the receiver received the packet, other nodes cannot forge a new message with a correct MAC. Therefore, SGLS can now prevent message tampering (A1) attack on unicast messages (i.e., data and control messages).

When a packet reaches a node that does not know about any node closer than it to the ultimate destination, this node is indicated as a dead-end [10]. The dead-end node then sends an error message to the packet's source node. This error message can be protected by the TIK protocol in the same way. However, there is no way to verify the authenticity of the error message at the source node. Therefore, a partial message tampering attack (A1-(ii)) is still possible on error messages. This attack can only be partially detected by the reputation system (see Section 4).

3.2 Secure Location Query & Reply Messages

The secure geographic forwarding concept developed in Section 3.1 can generally be applied to any unicast message if a source and a destination share the same secret key. Therefore, it can be applied to end-to-end unicast location control messages such as location queries and replies.

A location query can fail when the location information at the LS is out of date. When an LG moves from one square s_1 to another square s_2 in the grid system, the LSs (particularly those that are far away) will think that the LG still remains in the square s_1 . To cope with this, the LG leaves a *forwarding pointer* in s_1 indicating that it has moved to s_2 [9], and the node with the forwarding pointer can simply relay location queries using the TIK protocol for hop-by-hop authentication. Non-mutable parts can still be protected using the shared secret K_{SD} between source and destination.

3.3 Secure Location Update Messages

Any LS can be compromised by location table tampering attackers, who can manipulate the location update mechanism by changing any field of an LG's table entry at the LS. One natural solution is to use digital signatures or MAC.

In this paper, instead of cryptographic solutions, we propose to use a reputation system with timeout value. An LG updates its LSs based on its movement as described in Section 2.1. Thus, the LG sends the update messages at a rate proportional to its speed v and the distance to the LS. Therefore, in a location update packet, the LG can include a timeout value during which it has moved at least distance d since the last update, T_{update} . In this way, any mobile node can maintain a location cache of recently overheard update packets by setting a timer with the timeout value, $d/v + T_{update}$, for location information of the LG. If a node with this information receives or overhears a location query and finds the location information to be incorrect and $T_{current} < d/v + T_{update} - 2\Delta$, it determines that the corresponding LS is misbehaving and will

invoke the reputation system to indicate that the LS intentionally has changed the location information of LG and will accordingly drop this packet. Using this method, possible location table tampering attacks (A4) can be avoided.

3.4 Secure Exchange of HELLO Messages

In GLS, each node maintains a table of its immediate neighbors as well as each neighbor's neighbors. The neighbor's location information can be verified by using secure location verification techniques [16]. However, the location information about a neighbor's neighbors cannot be verified by using these techniques since they are out of the transmission range of the verifier.

In this paper, we propose to use the TESLA broadcast authentication method. Node A includes two additional fields in each HELLO packet: $\langle time\ interval, MAC \rangle$, where *time interval* is the pessimistic expected arrival time of the HELLO message at all two-hop neighbors. Note that a HELLO packet is broadcast periodically with interval t_h . As described in Section 3.1, node A 's local time is synchronized up to a maximum time synchronization error Δ with any other node's local time. Therefore, *time interval* can be set to any time interval for which the key is not released within the next $t_h + \tau + 2\Delta$ given τ , the maximum transit time between two-hop nodes. The MAC of the location information is generated by using the TESLA key K_i where i is the index for the time interval specified in the HELLO message.

When a neighbor node B receives a HELLO message, it checks whether the time interval in the HELLO message is valid (i.e., the TESLA key has not been disclosed yet). If the time interval is not valid, the node discards this message. Otherwise, the node modifies its next HELLO message by appending the location information with *time interval* and *MAC* from its neighbors.

When the location information is received by a two-hop neighbor C , node C checks the validity of the HELLO message by determining that the keys from the time interval specified have not yet been disclosed. Node A waits until it is able to disclose its key from the time interval specified; it then appends its key from that time interval to the next HELLO message. When node C receives this new HELLO message, it can verify the previous HELLO message from A . If this verification process fails, the reputation system is called upon to report the fact that neighbor B intentionally changes location information of its neighbor A . Therefore, we can detect possible location table tampering attacks (A4) in SGLS. One limitation of this scheme is that two-hop neighbors' location information can only be verified correctly after time $t_h + \tau + 2\Delta$ at a maximum.

4. REPUTATION SYSTEM

In this section, we describe modifications of the CONFIDANT reputation system for position-based routing protocols [15]. Our modified reputation system consists of the following components: *the monitor*, *the reputation manager*, and *the trust manager*. Every component is present in each node. Unlike CONFIDANT, we do not need the path manager because position-based routing protocols in general do not maintain a specific path from source to destination.

4.1 First-Hand Reputation Rating

When node i makes an observation about node j , $F_{ij} = (\alpha, \beta) = (\# \text{ of good behavior}, \# \text{ of bad behavior})$ is updated. For example, if

the observation is classified as misbehavior, the value of β is increased by one. When the reporting timer expires, the first-hand reputation information FR_{ij} about node j from node i is updated as follows: $FR_{ij} = v \cdot FR_{ij_old} + (1 - v) \cdot FR_{ij_new}$ where v is a weight value. On the other hand, during inactivity periods, we periodically decay the value as follows: $FR_{ij} = v \cdot FR_{ij_old} + (1 - v) \cdot FR_{initial}$.

4.2 Reputation Reporting and Rating

A node's reputation information is also sent to its neighbors by piggybacking on a HELLO message when the reporting timer expires. Assume node i receives the reported first-hand reputation information FR_{kj} about node j from node k , node i updates the reputation rating R_{ij} as follows: $R_{ij} = (1 - \omega) \cdot FR_{ij} + \omega \cdot FR_{kj}$ where ω is a small positive real number. This process is performed for all j being reported. Based on this reputation rating, node i classified node j as a good node if $R_{ij} \geq r$; or as a bad node if $R_{ij} < r$. The value of r is set to 0.5 in our simulations. To avoid blackmail attack, our reputation system also takes into account the trust rating of each node. For simplicity, we assume that false reports do not occur in our simulations (i.e., $\mu = 1$).

4.3 Countermeasures for Dropping and Tampering

There are two attacks that can be partially defended by our reputation system: *message dropping attack* and *HELLO message tampering attack*. Suppose there is a path from node S to D through intermediate nodes A , B , and C . Node A cannot transmit directly to C , but it can listen in on B 's traffic. Thus, when A transmits a packet for B to forward to C , A can find out whether B relays the packet or not. The monitor module maintains a buffer of recently sent packets and compares each overheard packet with the packet in its buffer to see if there is a match. If a packet has remained in the buffer for longer than a timeout interval, the reputation manager is called. The reputation manager then decreases the reputation value for the node responsible for forwarding the packet. As we explained in Section 4.2, if the reporting timer expires, the node sends its first-hand reputation information in its HELLO message to warn its neighbors of malicious nodes.

The monitor module of node A can overhear a packet generated from node B where the next hop address field is matched with one of its neighbors' addresses, and is not the final destination. If this packet remains in the buffer for longer than a certain timeout, the reputation manager is called. The same operations are applied to packets that node A itself has sent. In this way, our reputation system can detect possible message dropping attacks (A2).

When an attacker A changes or erases the location information of its neighbor B in its HELLO message, node B , or nodes C and D , which are the neighbors of both nodes A and B at the same time, can also detect this attack by investigating location information of B in this HELLO message. Therefore, the SGLS protocol can also partially prevent attackers from launching HELLO message tampering attack (A1).

4.4 Limitations of Reputation System

In general, it may be difficult to distinguish misbehaving nodes from transmission failures and other kind of failures [6] in wireless channels. A reputation system only provides probabilistic

Table 1. Simulation Parameters.

Secure GLS parameters	
HELLO message interval	2 seconds
Reputation reporting interval	10 seconds
Weight value v	0.9
Weight values ω	0 or 0.1
Threshold r	0.5

guarantees of the detection of misbehaving nodes. Although the reputation system with both positive and negative feedbacks can force misbehaving nodes to behave correctly up to the certain threshold level, it is impossible to avoid blackmail attack (A5) completely. For example, an attacker can first participate in the routing and data forwarding operations properly in order to increase its reputation and trust ratings to exceed certain threshold levels. After that, it can send the falsified reputation messages to the network. One feasible solution is the use of Tamper Resistant Module (TRM) to protect the routing modules [6]. The use of TRM can be justified due to the seriousness of blackmail attacks in hostile environments such as military battlefields.

5. DISCUSSIONS

In this section, we explain the rationale of the assumptions on which our proposed SGLS is based.

5.1 Network Assumptions

We do not consider possible attacks in both physical and medium access control layers. We assume that network links are bidirectional. That is, if node A is able to transmit to node B , then B is also able to transmit to A . Many wireless medium access control protocols require bidirectional links to exchange link-layer frames between a source and destination to avoid collisions. In addition, we assume wireless interfaces that support promiscuous mode operations. That is, if A is within range of B , it can overhear communications from B even if those communications do not directly involve A . Since there is no flooding message in GLS, we do not explicitly protect against packet injection in this paper. In other words, DoS attack in network layer is infeasible in GLS.

5.2 Node Assumptions

When SGLS uses TIK [7] for authentication of all messages, we assume that all nodes have tightly synchronized clocks, such that the difference between any two nodes' clocks does not exceed Δ . The value of Δ must be known by all nodes in the network. Since most position-based routing protocols require each node to have GPS to get its own location information, accurate time synchronization can be maintained with embedded GPS. While it is conceivable for an attacker to manipulate clock skews to affect the network performance, the result will be the same with that of a packet dropping attack. Therefore, the attacker will be identified by the reputation system.

5.3 Security Assumptions and Key Setup

The safety of SGLS relies on the keys stored in each node. Without relying on secure location service, we can set up any keys by choosing one of the following options:

- (1) Pair-wise shared secret keys (i.e., for symmetric keys: $n(n-1)/2$ keys in a network with n nodes).
- (2) Asymmetric keys (i.e., n public keys).

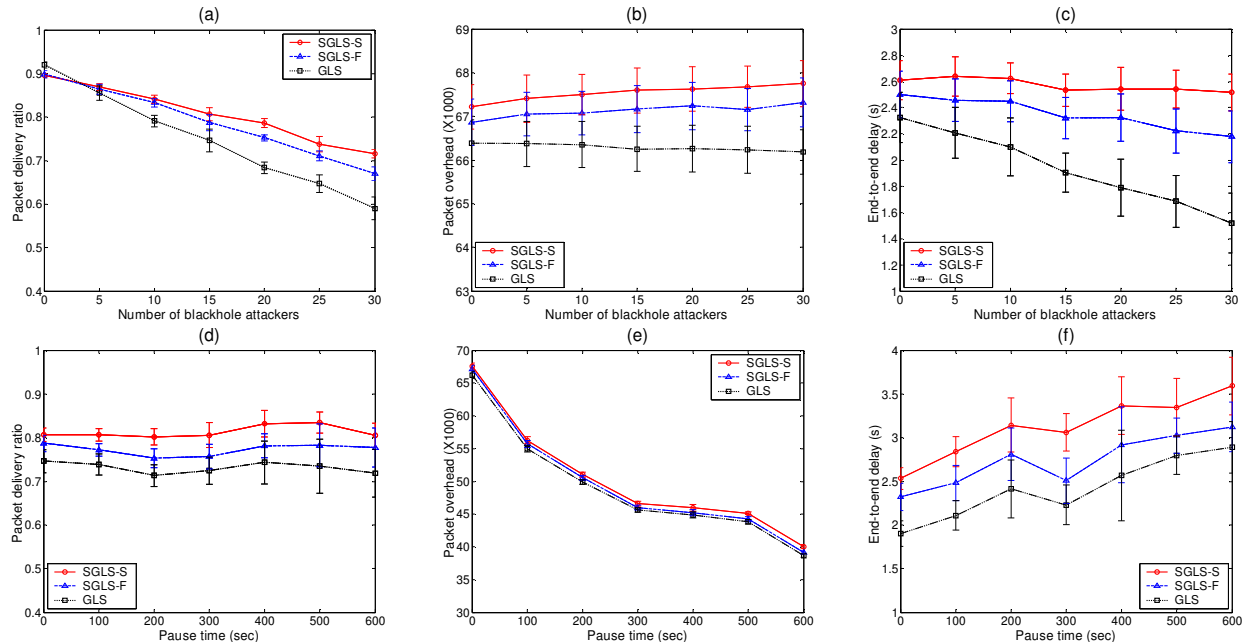


Figure 3. Performance comparisons between SGLSs and GLS with data blackhole attackers

- (3) Public TESLA key (i.e., n public keys) and shared secret keys between source and destination.

There are several key generation mechanisms with shared secret, for example, Kerberos [17]. However, to set up shared secret keys, we require authenticity and confidentiality, which are difficult in MANETs. And it may be expensive to generate the shared session keys between the initiator and all the nodes on the path from source to destination. To set up authentic public keys, we can assume a PKI and embed the trusted certification authority's public key in each node and then use that key to authenticate the public keys of other nodes. However, asymmetric operations such as digital signature are expensive and three to four orders of magnitude slower than symmetric operations such as MAC [7]. In this paper, we assume a mechanism that obtains an authentic public TESLA key for each node. This key distribution can be done through either non-cryptographic approach [18] or various distributed key management schemes [19][20] without a trusted third party. There are also several offline methods to exchange the secret information [21]. Therefore, any source and destination pair can set up the shared secret key based on these schemes.

6. PERFORMANCE COMPARISONS

We performed simulation experiments to evaluate the performance of our proposed SGLS protocol in the presence of dropping (i.e., blackhole) attackers.

6.1 Simulation Environment

We consider a network topology with 100 nodes randomly placed over a 1000×1000 (m^2) flat-grid. The size of an order-1 grid is 250×250 (m^2). We assume that 50 of these nodes are constant bit-rate (CBR) data sources, each sending fixed size 128-byte packets at 4 packets/s for 200 s. Each simulation run takes 600 simulated seconds. The characteristics of each mobile node's radio interface approximate the Lucent WaveLAN, operating as a shared-medium radio with a nominal bit rate of 2 Mb/s and a nominal radio range

of 250 m. For the medium access control layer, the IEEE 802.11 DCF is used. The propagation model combines both a free space propagation model and a two-ray ground reflection model. Table 1 provides a summary of other simulation parameters. A random waypoint model [1] is used for the mobility model. Each node moves in a straight line towards the destination at a speed that is uniformly distributed from 0 to 10 m/s. For fair comparisons, identical mobility and traffic scenarios are applied to both GLS and SGLS protocols. Results are averaged over 9 simulation runs; the error bars represent the 95% confidence intervals about the means.

To evaluate the proposed SGLS with reputation system as presented in Section 4, we model SGLS by modifying the ns-2 grid package [22] and implementing both the reputation system and blackhole attackers. In the following results, SGLS-S refers to SGLS using both first and second-hand reputation information (i.e., $\omega = 0.1$), and SGLS-F refers to SGLS using only first-hand reputation information (i.e., $\omega = 0$). We compare both SGLS-S and SGLS-F with the original GLS. The performance metrics for evaluations are packet delivery fraction, average end-to-end delay of transferred data packets, and routing overhead (i.e., the number of hop-by-hop transmissions of control packets).

6.2 Blackholes Dropping Data Packets

Figure 3(a)-(c) shows the simulation results with varying number of blackhole attackers who drop data packets in the network, and zero pause time for node mobility.

Figure 3(a) shows the packet delivery ratio as a function of the number of blackhole attackers. Both SGLS-S and SGLS-F yield a higher packet delivery ratio than GLS as the number of blackhole attackers increases. This shows that our proposed reputation system can effectively isolate blackhole attackers. As it uses also second-hand reputation information, SGLS-S works slightly better than SGLS-F with faster detection. One interesting point here is

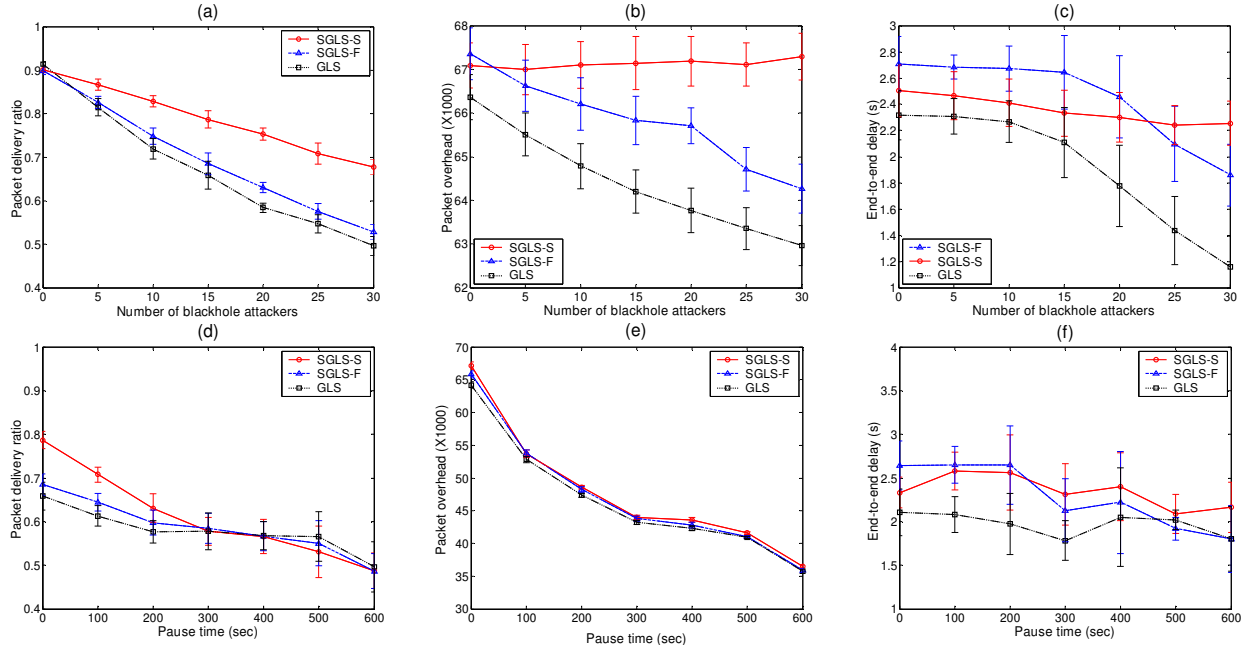


Figure 4. Performance comparisons between SGLSs and GLS with both data and control blackhole attackers.

that SGLS has a lower delivery ratio than GLS when the number of blackhole attackers is small. This is because the reputation system cannot distinguish between malicious dropping and other droppings as mentioned in Section 4.4. Figure 3(b) shows that GLS incurs a lower routing control overhead than SGLS. This is due to the fact that SGLS can detect the blackholes and re-initiate the location query (or detour) to avoid these nodes. These additional route discoveries increase the routing control overhead. Figure 3(c) indicates that GLS has a lower average end-to-end delay when compared with SGLS. Since SGLS incurs more routing control packets, the average end-to-end delay for data packets increases. Note that the average end-to-end delay of GLS decreases as the number of blackhole attackers increases. Since blackhole attackers drop data packets at the intermediate nodes and the dropped packets are not counted in the end-to-end delay calculation, the average end-to-end delay is decreased.

Figures 3(d)-(f) show the performance comparison with varying pause time (i.e., mobility), while keeping the number of blackhole attackers at 15 out of 100.

Figure 3(d) shows that the delivery ratio does not increase or decrease remarkably in all protocols. This is due to the fact that our simulation network is not very congested. Therefore, the increase of control overhead due to high mobility does not affect the delivery fraction. Figure 3(e) shows that the routing overhead of all protocols decreases as pause time increases (i.e., mobility decreases). That is because each node updates its closest location servers every time it moves a particular threshold distance d (100 meters in this paper) since sending the last update. This indicates that a node sends out updates at a rate proportional to its mobility. Figure 3(f) shows the average end-to-end delay for all three protocols increases as mobility decreases. We can find the reason from the fact that on average a longer path is obtained as mobility decreases in a MANET [6]. In our experiment, the average number of hops increases from 2.4 to 3.4 as pause time increases

from 0 to 600 seconds. Note that in a highly congested network, the end-to-end delay may instead decrease as mobility decreases.

6.3 Blackholes Dropping Data and Control Packets

In MANETs employing topology-based ad hoc routing protocols, such as AODV [3], control packet dropping attacks have no benefit for the attackers since they will not be able to join (or attack) the communication session as an intermediate node. However, since location query and location reply pass through different paths, control packet dropping attacker can still join the communication session as an intermediate node in position-based routing. In this set of simulations, blackhole attackers can drop not only data packets but also control (i.e., location query) packets to disrupt a routing protocol.

Figure 4 shows the simulation results with varying number of blackhole attackers, who drop both data and control packets. In Figures 4(a)-(c), the simulations employ a zero pause time. In Figures 4(d)-(f), the number of blackhole attackers is fixed at 15.

Figure 4(a) shows the packet delivery ratio as a function of the number of blackhole attackers. SGLS yields a higher packet delivery ratio than GLS as the number of blackhole attackers increases. This shows that our proposed reputation system can still isolate blackhole attackers even if they drop location query and reply packets. Due to the use of second-hand reputation information, SGLS-S works better than SGLS-F. One interesting point here is that the performance difference in delivery ratio between SGLS-S and SGLS-F is remarkable. This indicates that the inaccurate and slow detection of reputation information in SGLS-F reduces the delivery ratio when attackers drop control packets. Figure 4(b) shows that both GLS and SGLS-F incur a much lower routing control overhead than SGLS-S. That is due to the detection of blackholes and the restart of location discovery. This result shows that first-hand reputation information is not

enough when blackhole attackers drop both control and data packets. Results in Figure 4(c) indicate that both GLS and SGLS-F have a lower average end-to-end delay when compared with SGLS-S. This is due to the larger number of routing control packets incurred by SGLS-S to overcome blackhole attacks.

Figure 4(d) shows that the delivery ratio decreases quickly in all three protocols as pause time increases. This indicates that our reputation system cannot work efficiently when attackers drop both data and control packets as mobility decreases. That is because of the limitation of selecting and querying LSs in GLS. Since a small subset of deterministic nodes work as LSs, some nodes cannot find specific destination location information if the LS is malicious. As mobility decreases, the LS is changed less frequently, thus making the situation worse. Figure 4(e) shows that the routing overhead of all protocols decreases as pause time increases (i.e., mobility decreases). That is because each node updates its closest LSs every time it moves a particular threshold distance d after sending the last update. Figure 4(f) shows the average end-to-end delay for all three protocols decreases as mobility decreases. Although on average longer paths result as mobility decreases in a MANET, the average end-to-end delay of all three protocols decreases. Since blackhole attackers drop control packets at the intermediate nodes, frequently the path to the destination cannot be found, thus reducing the average end-to-end delay.

7. CONCLUSION

In this paper, we have proposed SGLS, which is a security enhancement to the original GLS protocol. The security mechanisms added to GLS include (a) TESLA with MAC and (b) a reputation system for monitoring. SGLS has the capability of preventing message tampering, dropping, and table tampering attacks by either malicious or compromised users. To the best of our knowledge, this may be the first paper to address security issues in position-based routing protocols.

Simulation results showed that in the presence of data and control packet dropping attacks, the proposed mechanisms maintain a high packet delivery ratio at the expense of a higher average end-to-end delay and routing overhead in general. We are investigating the computational complexity of SGLS through simulations. For future work, we plan to implement our algorithm on mobile devices, and study it in real world scenarios. Moreover, countermeasures against blackmail attacks will be investigated.

8. ACKNOWLEDGMENTS

This work was supported by a University of British Columbia Graduate Fellowship, and by the Canadian Natural Sciences and Engineering Research Council under grants RGPIN 262604-03 and 44286-00.

9. REFERENCES

- [1] J. Broch, D. Maltz, D. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proc. of ACM Mobicom*, Dallas, TX, Oct. 1998.
- [2] M. Mauve, J. Widmer, and H. Hartenstein, "A survey on position-based routing in mobile ad hoc networks," *IEEE Network*, vol. 15, pp. 30-39, Nov./Dec. 2001.
- [3] T. Imielinski and J.C. Navas, "GPS-based geographic addressing, routing, and resource discovery," *Commun. ACM*, vol. 42, pp. 86-92, 1999.
- [4] M. Zapata and N. Asokan, "Securing ad hoc routing protocols," in *Proc. of ACM Workshop on Wireless Security (WiSe)*, Atlanta, GA, Sept. 2002.
- [5] Y.-C. Hu, A. Perrig, and D.B. Johnson, "Ariadne: a secure on-demand routing protocol for ad hoc networks," in *Proc. of ACM Mobicom*, Atlanta, GA, Sept. 2002.
- [6] J.-H. Song, V. Wong, V. Leung, and Y. Kawamoto, "Secure routing with tamper resistant module for mobile ad hoc networks," *ACM Mobile Computing and Communications Review*, vol. 7, issue 3, July 2003.
- [7] Y.-C. Hu, A. Perrig, and D.B. Johnson, "Packet leashes: a defense against wormhole attacks in wireless network," in *Proc. of IEEE Infocom*, San Francisco, CA, Mar./Apr. 2003.
- [8] R. Anderson and M. Kuhn, "Tamper resistance—a cautionary note," in *Proc. USENIX Workshop on E-Commerce*, Oakland, CA, Nov. 1996.
- [9] J. Li, J. Jannotti, D.S.J. De Couto, D.R. Karger, and R. Morris, "A scalable location service for geographic ad hoc routing," in *Proc. of ACM Mobicom*, Boston, MA, Aug. 2000.
- [10] B. Karp and H.T. Kung, "GPSR: greedy perimeter stateless routing for wireless network," in *Proc. of ACM Mobicom*, Boston, MA, Aug. 2000.
- [11] A. Perrig, R. Canetti, J.D. Tygar, and D. Song, "The TESLA broadcast authentication protocol," *RSA Laboratories CryptoBytes Technical Newsletter*, vol. 5, no. 2, Summer/Fall 2002.
- [12] H. Krawczyk, M. Bellare, and R. Canetti, "HMAC: keyed-hashing for message authentication," *IETF RFC 2104*, Feb. 1997.
- [13] C. Madson and R. Glenn, "The use of HMAC-MD5-96 within ESP and AH," *IETF RFC 2403*, Nov. 1998.
- [14] IEEE Computer Society LAN/MAN Standards Committee, "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications, IEEE Std 802.11," Sept. 1999.
- [15] S. Buchegger and J.-Y.L. Boudec, "Performance analysis of the CONFIDANT protocol," in *Proc. of ACM Mobihoc*, Lausanne, Switzerland, June 2002.
- [16] N. Priyantah, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *Proc. of ACM Mobicom*, Boston, MA, Aug. 2000.
- [17] J. Kohl and B.C. Neuman, "The kerberos network authentication service (V5)," *IETF RFC 1510*, Sept. 1993.
- [18] F. Stajano and R. Anderson, "The resurrecting ducking: security issues for ad hoc wireless networks," in *Security Protocols 7th International Workshop*, Springer-Verlag, Berlin, Germany, 1999.
- [19] H. Yang, X. Meng, and S. Lu, "Self-organized network-layer security in mobile ad hoc networks," in *Proc. of ACM Workshop on Wireless Security (WiSe)*, Atlanta, GA, Sept. 2002.
- [20] P. Papadimitratos and Z.J. Haas, "Secure routing for mobile ad hoc networks," in *Proc. of SCS CNDS*, San Antonio, TX, Jan. 2002.
- [21] B. Schneier, *Applied cryptography second edition: protocols, algorithms, and source code in C*, John Wiley & Sons Inc., 1996.
- [22] NS-2 for grid. www.pdos.lcs.mit.edu/grid/sim/index.html.
- [23] S. Buchegger and J.-Y. Le Boudec, "A robust reputation system for mobile ad-hoc networks," *EPFL Technical report No. IC/2003/50*, July 2003.