Securing Location Aware Services over VANET using Geographical Secure Path Routing

Vivek Pathak
Department of Computer Science
Rutgers University
vpathak@cs.rutgers.edu

Danfeng Yao *
Department of Computer Science
Rutgers University
danfeng@cs.rutgers.edu

Liviu Iftode †
Department of Computer Science
Rutgers University
iftode@cs.rutgers.edu

***Abstract*— We propose to secure location aware services over VANET with our geographical secure path routing protocol (GSPR). GSPR is an infrastructure free geographic routing protocol which is resilient to disruptions caused by malicious or faulty nodes. Geographic locations of anonymous nodes are authenticated in order to provide location authentication and location privacy simultaneously. Our protocol also authenticates the routing paths taken by individual messages. This paper presents the design of the GSPR secure geographic routing protocol. The overhead of location authentication is investigated under various scenarios through network simulation. Results show that although the presence of malicious nodes increases the routing path length, a data delivery rate of larger than 80% is sustained even if 40% of the nodes are malicious.**

I. Introduction

Advancements in the availability and capability of vehicular computers have led to the creation of vehicular ad-hoc networks, or VANETs. Creation of location aware services is an important abstraction for providing useful services. Location aware services permit vehicular computers and their operators to make location oriented queries. Location aware services can help in navigating traffic, selecting convenient routes, and querying various infrastructure services like service stations along the travel route [1]. Although prior research has addressed location aware services in a benign environment, these services require location authentication in order to be credible in a hostile environment. Securing location aware services by authenticating node locations in a VANET is the focus of this work.

Geographic routing is an established protocol for routing in ad-hoc networks [2], [3], [4]. It relies on nodes knowing their geographic locations, and using their one hop neighbors for routing messages to target geographic destinations. Modern vehicles typically have an on-board GPS device. This makes geographic routing particularly suitable for routing in a VANET. Vehicles are typically used by a single person or family. The homing behavior of vehicles can therefore cause privacy violations. VANETs are also vulnerable to malicious nodes and other adversaries because of their ad-hoc networking model. The routing mechanism and the routed data both can be attacked. Securing location aware services in VANETs while simultaneously protecting location privacy makes our work interesting.

Securing ad-hoc routing is challenging because of the lack of pre-existing routing and security infrastructures. Nodes must create the routing infrastructure without using global knowledge. Lack of secure node identification is an additional challenge. Malicious or compromised nodes may pose as new nodes, or as known good nodes. Existing secure ad-hoc routing proposals have assumed various degrees of infrastructural support for addressing these challenges. In contrast, we use anonymous nodes and a cryptographic protocol for securing ad-hoc geographic routing in an infrastructure free manner. Our solution trades off security infrastructure for computational and messaging overhead. Superior resource provisioning of vehicular nodes makes our solution reasonable for VANETs.

We design an infrastructure-free secure geographic routing protocol. Our protocol protects location privacy and requires that nodes be able to determine their own geographic location. The protocol authenticates geographic locations thereby making it robust against malicious nodes. In contrast to existing location authentication research, our approach does not require out-of-band communication or shared secret initialization.

A. Our solution

We propose geographical secure path routing (GSPR) for securing ad-hoc routing against malicious nodes and passive adversaries. The routing protocol operates on location aware anonymous nodes to provide privacy preserving secure geographic routing for ad-hoc networks. The protocol has the following goals:

- Route messages to desired geographic locations in the presence of malicious nodes. Detect and avoid bad geographic regions containing malicious or faulty nodes.
- Authenticate self-generated public keys and geographic locations of nodes on the routing path.

This paper describes the GSPR protocol. The overhead introduced by the protocol is investigated in various operational and attack scenarios using the NS2 network simulator [5]. Due to space limitations, the correctness and attack resistance of the protocol are addressed in an extended paper [6].

The rest of this paper is organized as follows. An overview of geographic routing is presented in Section II. Section III

* This work has been supported in part by the Rutgers University Computing Coordination Council (CCC) Pervasive Computing Initiative Grant.

† This work has been supported in part by the NSF grant CNS-0520123.

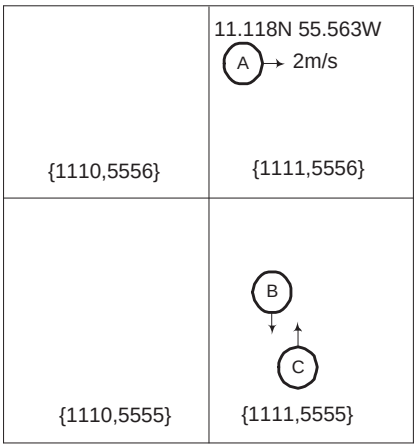

Fig. 1. An example of geographic location and integer co-ordinates. Co-ordinates are shown in curly brackets, e.g., {1110, 5556}. 11.118N, 55.563W is a geographic location.

discusses the geographic authentication model used for the proposed secure routing protocol. Section IV describes the GSPR protocol. The performance of GSPR in various operational and attack scenarios is investigated in Section V. Related work is discussed in Section VI, and Section VII presents the conclusions and future work.

II. Preliminaries

Geographic routing is a well researched approach for ad-hoc routing [4], [3], [7], [2]. Nodes are expected to know their own geographic locations, and to share it with their one-hop neighbors through periodic beacons. The periodic beacons allow each node to know the geographic locations of its one-hop neighbors.

Messages carry their target locations as they are routed though the ad-hoc network. Under the assumption of bidirectional connectivity, geographic routing can be efficiently implemented on a planar sub-graph of the one-hop connectivity graph. A number of planarization approaches have been developed, and each can lead to a different subset of neighbors to use for routing. Once the choice of neighbors to use for routing is made, the next hop can be determined by greedily selecting the next hop to minimize the remaining distance to the target geographical location. Greedy forwarding fails if there is no next hop among the neighbors which is closer to the destination. In this case, geographic routing makes progress by entering the perimeter mode, in which the next-hop is selected to traverse the perimeter of the region where greedy forwarding fails. Perimeter mode forwarding continues as long as there is no better greedy next hop neighbor and the initial location where the perimeter mode started is not revisited again. Geographic routing is simple and efficient. The state required at each node depends only on the node density.

III. Model

This section discusses the assumptions, the problems being solved, and the notation used in the paper. The participants in our geographical secure path routing protocol are referred to as nodes in the rest of this paper.

A. Assumptions

All the nodes are assumed to know their geographic locations. The protocol operates on integer co-ordinates which can be constructed by scaling the geographic co-ordinates with a global constant. For example, as shown in Figure 1, the integer co-ordinates $\{1111, 5556\}$ of node *A* can be derived from the geographic location 11.118N, 55.563W by applying a scaling factor of 100. Nodes generate their own public-private key pairs. We also make standard assumptions about the non-invertibility of popular cryptographic functions.

Nodes are identified through short-lived temporary pseudonyms. These pseudonyms are also used as the physical level node identifiers in order to prevent physical level identification attacks. Pseudonyms are constructed from a pseudo-random number generator. This ensures that pseudonyms can not be used to derive real node identity. Nodes are assumed to silence transmissions while changing pseudonyms. Nodes can also estimate the number of one-hop neighbors either by listening to local transmissions or by knowing the node density. These capabilities are sufficient for using mix zones. Mix zones are regions with sufficient node density which can provide large enough anonymity sets to support the desired level of anonymity [8]. Changing pseudonyms within mix zones prevents node tracking by making it impossible to associate a series of pseudonyms with a physical node. This protects the location privacy of participating nodes.

We assume that promiscuous mode reception is enabled on the networking adapters. Nodes overhear all transmissions in their one-hop neighborhood. This enables detection of malicious activity in the one-hop neighborhood. The underlying data link layer is expected to handle the hidden terminal problem, packet collision, and asymmetrical connectivity issues. Node connectivity is assumed to be symmetrical and resilient to packet losses caused by collisions or jamming.

Node connectivity is assumed to have a range limited one-hop neighborhood. The one-hop neighborhood is bounded by a maximum distance *R*. The maximum range distance is assumed to be a fixed global constant, which depends on the the link layer technology being used. For example, in an outdoor IEEE 802.11 peer-to-peer wireless network, the maximum range is limited to a few hundred meters. Violations of connectivity radius are detected, and result in the offending node being eliminated from the secure routing protocol. Limited connectivity radius is a reasonable assumption in the context of standardized wireless networking hardware.

B. Definitions

Consistent with traditional geographic routing [4], all the nodes are located on a plane. Every node p has a geographic location: $\text{Location}(p) \equiv (\mathbf{p_x}, \mathbf{p_y})$. The integer co-ordinates p_x and p_y of the node are computed by scaling the geographic location with the global scaling factor. Each node p has a

$f \circ g(x)$	Function composition of f and g, i.e. $f(g(x))$.
K_p	Public key of the node p.
K_p^{-1}	Private key of the node p.
$\{x\}_p$	A message consisting of string x signed by node p.
$(\mathbf{p_x}, \mathbf{p_y})$	The geographic location of node p, also represented as Location(p).
(p_x, p_y)	Integer co-ordinates of node p derived by scaling its geographic location.
$\mathcal{N}(p)$	The set of nodes in the one-hop neighborhood of node p.

TABLE I
NOTATION

Category	Specific Threat
Routing attack	Dropping messages
	Routing in the wrong direction
Data attack	Payload modification
	Control data modification
Malicious node	Reporting false location
	Directional transmission
	Transmission power changes
	Tracking node location

TABLE II
OVERVIEW OF THE THREATS HANDLED BY GSPR

public key K_p such that a message m can be secretly sent to p by encrypting with the well known public key as $K_p(m)$. The corresponding private key K_p^{-1} is only known to p, and can not be computed from K_p. All the messages transmitted from p are digitally signed with the private key K_p^{-1}. The notation used in this paper is summarized in Table I.

Our GSPR protocol resists attacks from malicious and faulty nodes. Malicious nodes may intentionally try to give incorrect responses while faulty nodes may be attacked and have incorrect inputs to offer. Honest nodes are distinguished from malicious or faulty nodes as follows:

DEFINITION 1 (Honest Node): An honest node knows its correct geographical location, follows the maximum range constraint, and executes the GSPR routing protocol correctly. Otherwise, the node is called malicious or faulty.

The ad-hoc network consists of honest and malicious nodes. These nodes may be placed at arbitrary geographical locations. Nodes become candidates for geographic routing depending on their geographic location. Routing paths consist of sequences of nodes. Each node on the routing path is responsible for forwarding the message towards the geographic destination. Honest nodes in the one-hop neighborhood of forwarding nodes are called honest witnesses. The presence of honest witnesses allows the protocol to secure geographic routing while forwarding messages.

Geographic routing allows packets to be routed to destination locations. The routing protocol will return a failure message if there are no nodes in the one-hop neighborhood of the target location. If the one-hop neighborhood of the target location has one or more nodes, then each of them is considered a valid destination. Our GSPR protocol operates only on honest nodes. The routed packets are expected to reach target locations by using secure routing paths only.

DEFINITION 2 (Secure Routing Path): A secure routing path consists of a sequence of honest nodes, each of which is the one-hop neighbor of its predecessor.

Participation in our protocol allows nodes to find a secure routing path to the destination. The source node requests for a node located near the desired geographic location. GSPR finds an honest node in the one-hop neighborhood of the location if possible. The returned response contains the public key of the discovered node and the secure routing path to it. Message integrity is guaranteed for both query and response messages. Next-hop nodes for the query messages are determined by geographic routing while the response message is source routed on the reverse path. The security properties of GSPR protocol are listed below:

DEFINITION 3 (Properties of GSPR protocol): Given a message $\mathfrak{M}$ starting at node s with the destination geographic location D, a geographic secure path routing (GSPR) protocol is secure if the following properties hold:

- If there is a secure routing path $\mathcal{S}(s,d)$ from the source node s to a node d located within one-hop distance of the destination location $\mathcal{D}$, then $\mathfrak{M}$ is routed to d.
- The destination node d receives the secure routing path $\mathcal{S}(s,d)$.
- On receiving the returned response, the source node s gets the correct public key K_d of the destination, and the secure routing path $\mathcal{S}(s,d)$ traversed by the message.

C. Threat model

A summary of the threats is given in Table II. Attacks can be mounted on the routing mechanism by dropping or incorrectly forwarding messages. Incorrect forwarding means forwarding messages towards incorrect directions, and includes classical attacks like routing loops and wormholes. Attacks on routed data could target the payload or the control data required for protocol operation. Control data susceptible to modification includes node identifiers, node locations, and other data fields governing the routing protocol.

Nodes are also vulnerable to attack in ad-hoc networking environment. Compromised nodes can be controlled by an attacker causing them to behave maliciously or incorrectly as defined in Definition 1. Malicious nodes may also collude to continuously track the location of a node thereby violating its location privacy. In case of nodes having home locations, location privacy violations also make anonymity violations more likely.

A number of low level attacks are possible against wireless ad-hoc networks. The jamming attack blocks radio transmissions in a given geographic region thereby preventing

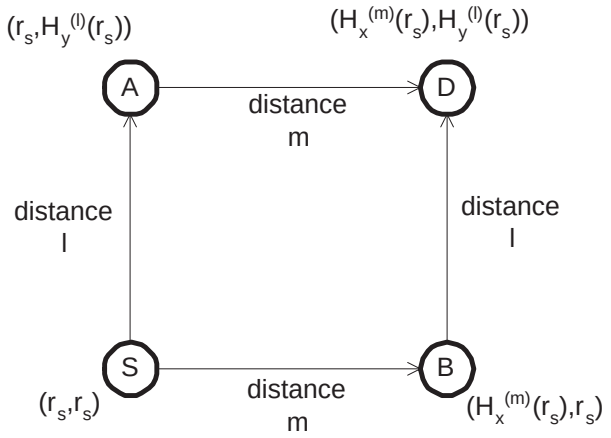

Fig. 2. Using associative one-way functions to create geographic hashes. H_x and H_y are associative one-way functions which are applied to maintain geographic hash values at different geographical locations. r_s is a random nonce published by node s and serves as the geographic hash of s at s.

the routing protocol from using that area. Jamming can be tackled with spread spectrum techniques [9]. Other low level attacks include transmission power changes and directional transmissions. These attacks are detected by the protocol and the responsible nodes classified as malicious.

IV. Geographical secure path routing

The details of our geographical secure path routing protocol are described in this section. Each of the building blocks: geographic hashes, periodic beacons, geographic routing, and malicious node detection are presented below.

A. Geographic hashes

We develop a novel method, called geographic hashes, for encoding relative geographic positions of two nodes. Our solution associates unforgeable transient geographic hashes to relative geographic positions as shown in Figure 2. Nodes maintain a set of integer tokens called geographic hashes, which associate a secret with a geographic location. The secret can not be determined without being in the vicinity of the location, but its knowledge can be verified remotely.

Geographic hashes are created through modular arithmetic. Consider a large prime p and a generating number a, such that the function $f(x) \equiv a^x \bmod p$ maps $\mathbb{Z}_p^\star = \{1,\ldots,p-1\}$ onto itself. Each integer $h \in \mathbb{Z}_p^\star$ can be used to represent a one-way function $H(x) \equiv (a^h)^x \bmod p$ since the discrete logarithm problem of finding x, given $y = a^x \bmod p$, is believed to be NP-hard.. Formally, geographic hash is defined next.

DEFINITION 4 (Geographic Hash): Each node A periodically publishes the following geographic hash parameters: a large prime p, a generator a for $\mathbb{Z}_p^\star = \{1,\ldots,p-1\}$, three integers $\zeta_A, \eta_A, \theta_A \in \mathbb{Z}_p^\star$, and a time interval Δ_A indicating the expiry time for a single version of the geographic hash.

The geographic hash of A is initialized to (r_A, r_A) at A, where r_A is a random nonce selected by A. Successive versions $r_A[i]$ and $r_A[i+1]$ of the random nonce satisfy:

$$r_A[i+1]a^{\theta_A} \bmod p = r_A[i]$$

Each node B in the neighborhood of A computes the geographic hash of A at B as follows:

$$\mathrm{GH}(A,B) = \left(r_A a^{\zeta_A \Delta x} \bmod p, r_A a^{\eta_A \Delta y} \bmod p\right)$$

where Δx and Δy are the differences in the integer coordinates related to the geographic location of A and B.

The geographic hash is a tuple of integers computed by repeated applications of one-way functions to the locally known geographic hash of a node. Since the construction of geographic hashes encodes the relative geographic locations of nodes, the computed geographic hash values would be identical across different calculation paths.

B. Beacon

Nodes are required to know one-hop neighbor locations for geographic routing. This is achieved by having the nodes transmit a periodic beacon containing the node identifier and location. Nodes continuously listen for beacons from neighboring nodes. The information gathered from the received beacons is stored in memory in order to support geographic routing. Our protocol extends the beacon to include the public key and the random nonce selected by the node. This ensures that public keys of nodes are well known in the one-hop neighborhood. The beacon also includes locations and geographic hashes of neighboring nodes. Beacon messages are digitally signed with the private key of the node and are broadcast to all the one-hop neighbors.

DEFINITION 5 (Periodic Beacon): Each node p periodically broadcasts a beacon message containing its geographic location, random nonce r_p, public key K_p, neighbor information list $\mathbb{Q}$, and a set of geographic hashes $\mathbb{G}$ known to it. The beacon message is broadcast to all nodes in the one-hop neighborhood $\mathcal{N}(p)$ of p as shown below:

$$p \quad \rightarrow \quad \mathcal{N}(p) \qquad \left\{\{p, \mathrm{Location}(p)\}_p, r_p, K_p, \mathbb{Q}, \mathbb{G}, \mathbb{M}\right\}_p$$

$$\text{where} \quad \mathbb{Q} \equiv \left\{\{q, \mathrm{Location}(q)\}_q \mid q \in \mathcal{N}(p)\right\}$$
$$\text{and} \quad \mathbb{G} \equiv \{\langle q, \mathrm{GH}(q,p)\rangle \mid \mathrm{Distance}(q,p) < 2R\}$$
$$\text{and} \quad \mathbb{M} \equiv \{\langle q,\ \mathrm{Evidence}\ \rangle \mid q \text{ is malicious }\}$$

The periodic beacon permits sharing the information which is used for validating routing actions. Sharing the neighbor information list $\mathbb{Q}$ helps in detecting false location attacks among the set of one-hop neighbors. The geographic hashes of nodes located within twice the maximum one-hop radius R are stored in memory. These geographic hashes are shared with neighboring nodes in order to detect malicious routing behavior beyond the one-hop neighbors.

C. Routing protocol

The GSPR protocol has a two step query-response messaging model. Payload and control data are sent towards the destination location as is done in traditional geographic routing protocols. The returning source routed acknowledgment completes the protocol. Figure 3 has a concise description of

- BEGIN FORWARDING
  The source node s begins routing the payload to the destination location $\mathcal{D}$. The message contains a location list $\mathbb{L}_s = [\text{Location}(s)]$ and a random nonce r_s selected by the source node.
  $$s \quad \rightarrow \quad p_0 \qquad \{\text{Forward}, \mathcal{D}, r_s, \mathbb{L}_s, \text{Payload}\}_s$$
- GEOGRAPHIC FORWARDING
  Each node p_i on the routing path forwards the message to the next hop p_{i+1} which is determined by geographic routing.
  $$p_i \quad \rightarrow \quad p_{i+1} \qquad \{\text{Forward}, \mathcal{D}, r_s, \mathbb{L}_{i-1} + [\text{Location}(p_i)], \text{Payload}\}_i$$
  The operation also causes a "local response" message to be returned to the previous hop node p_{i-1}. The local response contains public key, node identifier, and geographic hash information about the next-hop node:
  $$p_{i-1} \quad \leftarrow \quad p_i \qquad \{\text{Local Response}, r_s, K_{p_i}^{-1}(r_s), \langle K_{p_{i+1}}, p_{i+1}, \text{Location}(p_{i+1}), \text{GH}(p_{i+1}, p)\rangle\}_i$$
- END FORWARDING
  On verifying the integrity of the received message, the destination node d sends back the "recursive response" to the source. The recursive response contains the location list $\mathbb{L}_d$. It also contains a list of public keys of the nodes on the reverse routing path. $\mathbb{P}_d = [K_d]$.
  $$p_{k-1} \quad \leftarrow \quad d \qquad \{\text{Recursive Response}, \mathbb{L}_d, r_s, q_d \equiv K_d^{-1}(r_s), \mathbb{P}_d\}_d$$
- REVERSE RESPONSE FORWARDING
  The location list $\mathbb{L}_d$ is checked to ensure the current node is on the reverse route. If so, the following message is transmitted to the next node p_{i-1} on the reverse route:
  $$p_{i-1} \quad \leftarrow \quad p_i \qquad \{\text{Recursive Response}, \mathbb{L}_d, r_s, K_{p_i}^{-1}(q_{i+1}), \mathbb{P}_i\}_i$$
- VERIFICATION
  The verification operation permits to source node to verify the public keys along the path by checking:
  $$r_s = K_{p_0} \circ K_{p_1} \circ \ldots \circ K_d(q_0)$$

Fig. 3. Geographic Secure Routing Protocol

the protocol. Malicious node detection results in nodes being detected as malicious or the neighborhood being classified as a bad neighborhood. Malicious nodes are not used for routing and honest nodes do not forward messages if they are located in bad neighborhoods.

D. Malicious node detection

Malicious node detection is based on the broadcast nature of wireless communication and is modeled after the watchdog protocol [10]. Nodes listen to the transmissions of their neighbors in order to detect malicious nodes. Malicious nodes are not used for routing. Honest nodes witnessing malicious activity will warn neighboring nodes through the periodic beacon. Watchdog protocols are vulnerable to blacklisting attacks. Because the GSPR protocol uses temporary pseudonyms, a blacklisting attack can only have temporary effect. Avoiding temporary blacklisting is not a goal of the protocol, and is not considered further. Malicious nodes are detected both by checking for inconsistencies in the periodic beacons and by checking the correctness of geographically routed messages and their reverse source routed responses.

1) Beacon validation: Periodic beacons are received from neighboring nodes. Beacons are stored in memory and their contents used for detecting malicious nodes. Nodes launching false location attacks are detected by applying the range constraint R. Each node constructs a number of mappings from pseudonym to location, one received from each neighbor. Small inconsistencies in location are ignored as location errors. Larger inconsistencies permit the node to conclude that its one-hop neighborhood has malicious nodes and is a bad neighborhood. Nodes located in bad neighborhoods do not forward messages, but continue to transmit the beacon in order to propagate geographic hashes and malicious node information.

2) Forwarding validation: Operating in promiscuous mode permits overhearing wireless transmissions of one-hop neighbors. Incorrect forwarding is detected and the incorrectly forwarding nodes classified as malicious. Message forwarding operations of GSPR can be abstracted to a simple multi-hop ad-hoc forwarding protocol as follows: Let A, B, and C be successive hops on a routing path. All the transmissions made by B must be received by A because of the one-hop neighbor relationship. Therefore, A can detect if B fails to forward a message by listening for the next-hop transmission. In this case the honest previous hop node A can identify the malicious node B without the need for other honest witness nodes. Similarly, A can check the overheard next-hop transmission (from B to C) for malicious payload modification or control data modification. Data tampering by B is detectable as the message from B to C is digitally signed with the private key of B which can be verified by A because of being in the one-hop neighborhood. In case both B and C are malicious, B can forward the message correctly to C, and later collude by not reporting a malicious forwarding by

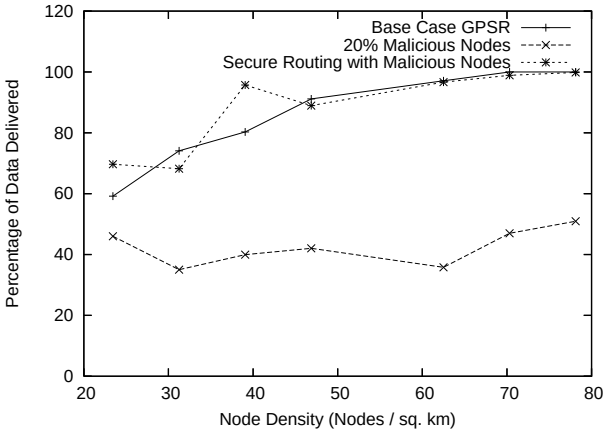

Fig. 4. Data delivery vs. node density on a square area with stationary nodes.

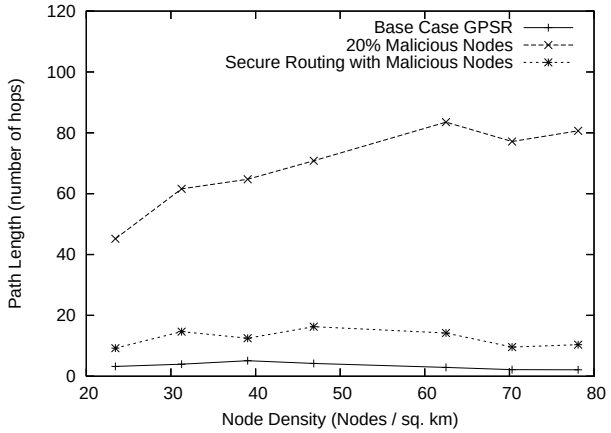

Fig. 5. Routing path length vs. node density on a square area with stationary nodes.

C. This collusion case is detected by the honest witness node *G* which is the one-hop neighbor of *A*, *B*, and *C*. Due to space limitations we refer the reader to [6] for a proof that such honest witnesses will exist given sufficient node density, and that the presence of the honest witness will permit *A* to recognize that *B* and *C* are malicious.

On encountering malicious nodes, the previous hop node *A* will find another route to the destination or send back a routing failure to the source. Nodes also check the integrity of messages and feasibility of routing paths by validating the geographic location list $\mathbb{L}$ for maximum transmission limit *R*. Messages violating the range constraint are dropped, and the previous hop node classified as malicious. The security of our protocol is shown in the extended version of this paper [6].

THEOREM 1: Let *R* be the node connectivity radius. If the honest node density is greater than $\frac{18}{R^2}$, then the geographical secure path routing protocol provides the following security properties:

- Messages are routed through secure routing paths.
- The destination node receives the secure routing path.
- The source node receives the secure routing path and the correct public key of the destination node.

V. Performance analysis

Performance of geographic secure path routing is analyzed by the NS2 network simulator [5]. Although NS2 has support for wireless and mobile ad-hoc network simulation, geographic routing is not available in the standard NS2 code. Therefore, we use the patch provided by Kiess [11], [12] that maintains Karp's original implementation of GPSR [4]. The patch simulates IEEE 802.11 MAC layer with a node range of 500m. It provides support for mobility through the random way point model [13].

Geographic routing takes all routing decisions based on the local one hop neighborhood. Since secure routing detects and avoids malicious nodes, the changes to routing performance can be evaluated by changing the routing behavior to avoid malicious nodes. The remaining protocol operations just authenticate node locations and public keys without affecting routing. Therefore, we change only the local routing behavior of geographic routing in order to assess the performance of GSPR. We modified the simulator code to keep track of malicious or honest nodes. The routing mechanism was also modified to operate in the base mode or to avoid malicious nodes for routing.

The objective of this evaluation is to compare the routing performance and attack resilience of traditional insecure GPSR protocol against our proposed geographical secure path routing protocol. We select percentage rate of data delivery and the routing path length as the indicators of routing performance. The comparative evaluation of the two routing protocols is done for various combinations of node density, mobility, and the presence of malicious nodes.

A. Node density

GPSR and secure routing were simulated on NS2 using one constant bit rate source per node. Simulation is done with stationary nodes randomly placed on a square area with side 800m. The number of nodes was varied from 20 to 50 to create a number of node density scenarios. The simulation was run for 300 seconds of simulated time. An average of 10 runs was used in the following observations. The delivery rate was calculated by counting the number of application packets sent and received. Path length was computed by tracing GPSR packets through forwardings. As shown in Figure 4, the baseline GPSR protocol attains a high percentage of packet delivery. This is consistent with Karp's observations in [4]. It can be observed that introducing 20% malicious nodes severely impacts the effectiveness of GPSR as demonstrated by the reduction in delivery rate. Geographic secure path routing is resilient to the malicious nodes. Its delivery rate closely replicates the rate achieved by GPSR in a benign environment.

The impact of incorrect routing introduced by malicious nodes is shown in Figure 5. Given the node range of 500m, and the 0.64km^2 node placement area, we expect nodes to be about 2 hops away in the greedy routing case. This is consistent with our average reading of 3.38 hops for GPSR in

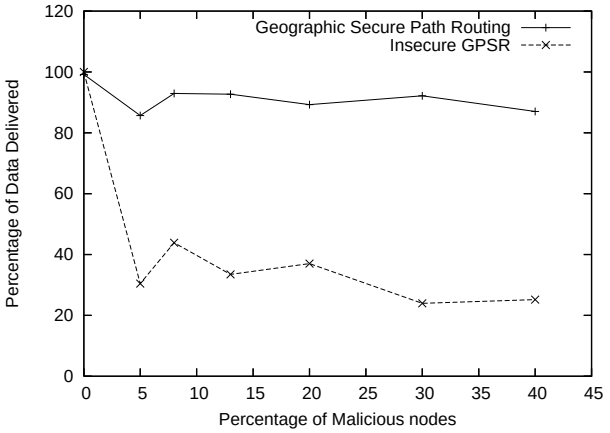

Fig. 6. Effect of malicious nodes on data delivery.

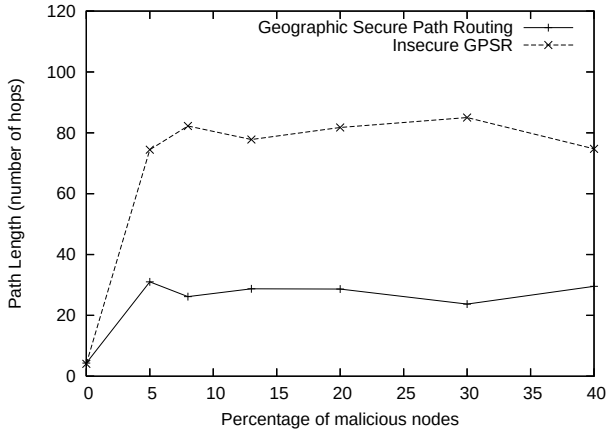

Fig. 7. Effect of malicious nodes on path length.

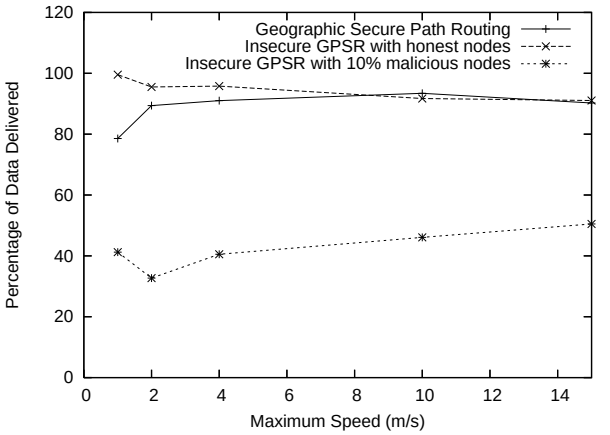

Fig. 8. Effect of mobility on data delivery for baseline and with 10% malicious nodes.

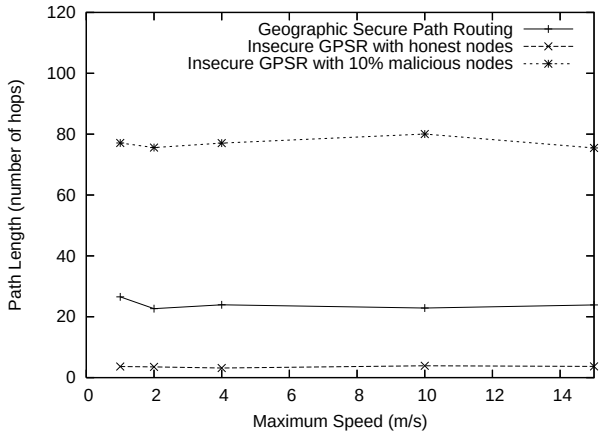

Fig. 9. Effect of mobility on forwarding path length for baseline and with 10% malicious nodes.

benign environment. We also note that secure routing incurs an overhead on the routing path length by routing packets around malicious nodes. The average number of hops for geographic secure path routing with 20% malicious nodes is 10.37, an increase of about three times over GPSR in benign environment.

B. Effect of malicious nodes

Malicious nodes publish an incorrect location on their beacons. We modify the published location of malicious node to appear as the closest neighbor for the packet being forwarded. This allows malicious nodes to falsely become the next hop neighbors. Malicious nodes also forward the packet to a random neighbor in violation of geographic routing rules. The effect of malicious nodes on GPSR and geographic secure path routing is studied by running an NS2 simulation with varying proportions of malicious nodes on a universe of 42 stationary nodes. These nodes are placed randomly in a rectangular area of 1.5km by 0.5km. Observations are collected by averaging the collected over 10 runs of 300 seconds simulated time each.

The data delivery achieved in presence of malicious nodes is shown in Figure 6. It can be observed that insecure geographic routing is very sensitive to malicious nodes. The delivery rate falls rapidly even with a small percentage of malicious nodes. The simulation also indicates that malicious nodes do not affect the delivery rate of our secure routing. The secure delivery rate falls from close to 100% in a benign environment to about 90% when 40% of the nodes are malicious. The effect on forwarding path length is shown in Figure 7. The path length shown by insecure geographic routing grows by orders of magnitude as the malicious nodes force the insecure protocol to route in incorrect directions. Geographic secure path routing incurs a more modest overhead by rejecting malicious nodes for routing.

C. Effects of mobility

The effect of mobility on data delivery rate is given in Figure 8. Mobility improves the data delivery rate of secure routing because mobility allows nodes to discover new honest nodes. This effect is also found in the insecure geographic routing to a smaller degree. Geographic routing in insecure environment becomes less effective with increasing mobility because of increasing chance of inaccuracy in one hop node locations. We also note that mobility reduces the routing path length of secure routing as shown in Figure 9. This happens because mobility increases the chance of discovering new honest nodes in the one hop neighborhood.

VI. Related work

Secure routing in ad-hoc networks has been investigated by a number of prior works. Hu and Perrig propose the Ariadne protocol for securing on-demand and source routing protocols in ad-hoc networks [14]. Similarly, the SEAD protocol [15] secures distance vector routing in ad-hoc networks. Both the protocols requires a secure cryptographic initialization phase, but use highly efficient symmetric key cryptography. While our approach uses expensive modular arithmetic, it does not require secure initialization.

The privacy threat posed by location aware devices has also been a topic of intense research. Existing research has taken two approaches for protecting user privacy: the first is to fudge the locations of identifiable nodes as in [16], [17]. The second is to use pseudonyms for temporary identification of nodes as proposed in [18], [19], [20], [21], [8]. We use the latter approach because secure geographic routing requires authenticated locations. Using temporary pseudonyms for location aware nodes allows us to provide location privacy and location authentication simultaneously.

A multi-hop anonymous challenge mechanism has been used by Mahajan et. al. [22] for detection of free riders in ad-hoc wireless network. Their mechanism requires two-hop transmission of challenge messages. Our recursive challenge response is related to their mechanism in the sense that remote nodes are used for security processing. Unlike in our work, the authors do not study the effects of mobility.

A scheme based on secure cryptographic initialization is presented by Liu et. al. [23] to provide robust location estimation for sensor nodes in a hostile environment. Our approach is distinguished in the sense that we do not require any secure initialization. However, our approach is more expensive, requiring asymmetric cryptography. We discuss this as a future work in Section VII.

VII. Conclusion and future work

We design and evaluate geographical secure path routing, a privacy preserving ad-hoc routing protocol, which geographically routes messages through anonymous nodes to destination locations. The secure routing protocol also authenticates the public key and the geographic location of destination nodes.

Geographical secure path routing protocol requires associative cryptographic one-way hash functions for security. These hash functions are derived from the discrete logarithm problem which uses expensive modular arithmetic. Superior resource provisioning of vehicular networks makes our solution suitable for securing location aware services on VANET.

Geographical secure path routing protocol was evaluated with the NS2 network simulator for various values of node density, node mobility, and the proportion of malicious nodes. Evaluation results show that the protocol tolerates malicious nodes with an increased routing path length. The geographical secure path routing protocol is also able to maintain a low loss rate even when the majority of nodes are malicious.

References

[1] M. D. Dikaiakos, A. Florides, T. Nadeem, and L. Iftode, "Location-aware services over vehicular ad-hoc networks using car-to-car communication," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 8, pp. 1590–1602, 2007.

[2] P. Bose, P. Morin, I. Stojmenovic;, and J. Urrutia, "Routing with guaranteed delivery in ad hoc wireless networks," *Wirel. Netw.*, vol. 7, no. 6, pp. 609–616, 2001.

[3] G. G. Finn, "Routing and addressing problems in large metropolitan-scale internetworks," Information Sciences Institute, Research Report 180, March 1987.

[4] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM Press, August 2000, pp. 243–254.

[5] "The Network Simulator ns-2 (v2.1b8a)," http://www.isi.edu/nsnam/ns/, October 2001.

[6] V. Pathak, L. Iftode, and D. Yao, "Securing geographical routing in mobile ad-hoc networks," Department of Computer Science, Rutgers University, Tech. Rep. 638, July 2008.

[7] Y.-J. Kim, R. Govindan, B. Karp, and S. Shenker, "Geographic routing made practical," in *Proceedings of 2nd Symposium on Networked Systems Design and Implementation*. USENIX, 2005, pp. 217–230.

[8] S. Capkun, J.-P. Hubaux, and M. Jakobsson, "Secure and privacy-preserving communication in hybrid ad hoc networks," Ecole Polytechnique Fédérale de Lausanne, I&C Research Report 200444, May 2004.

[9] W. Xu, K. Ma, W. Trappe, and Y. Zhang, "Jamming sensor networks: attack and defense strategies." *IEEE Network*, vol. 20, no. 3, pp. 41–47, May-June 2006.

[10] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *MOBICOM*, 2000, pp. 255–265.

[11] W. Kiess, H. Füßler, J. Widmer, and M. Mauve, "Hierarchical Location Service for Mobile Ad-Hoc Networks," *ACM SIGMOBILE Mobile Computing and Communications Review (MC2R)*, vol. 8, no. 4, pp. 47–58, October 2004.

[12] W. Kiess, "Hierarchical Location Service for Mobile Ad-Hoc Networks," Master's thesis, Department of Mathematics and Computer Science, University of Mannheim, 2003.

[13] C. Bettstetter, "Mobility modeling in wireless networks: categorization, smooth movement, and border effects," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 5, no. 3, pp. 55–66, 2001.

[14] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks." *Wireless Networks*, vol. 11, no. 1-2, pp. 21–38, 2005.

[15] Y.-C. Hu, D. B. Johnson, and A. Perrig, "SEAD: secure efficient distance vector routing for mobile wireless ad hoc networks." *Ad Hoc Networks*, vol. 1, no. 1, pp. 175–192, 2003.

[16] J. Lundberg, "Routing security in ad hoc networks," Helsinki University of Technology, Tech. Rep. Tik110. 501, 2000.

[17] A. Grlach, W. W. Terpstra, and A. Heinemann, "Survey on Location Privacy in Pervasive Computing," in *Proceedings of The First Workshop on Security and Privacy at the Conference on Pervasive Computing (SPPC)*, April 2004.

[18] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing," *IEEE Pervasive Computing*, vol. 2, no. 1, pp. 46–55, 2003.

[19] X. Wu and B. K. Bhargava, "Ao2p: Ad hoc on-demand position-based private routing protocol." *IEEE Trans. Mob. Comput.*, vol. 4, no. 4, pp. 335–348, 2005.

[20] S. M. M. Rahman, M. Mambo, A. Inomata, and E. Okamoto, "An anonymous on-demand position-based routing in mobile ad hoc networks." in *SAINT*. IEEE Computer Society, 2006, pp. 300–306.

[21] B. Zhu, Z. Wan, M. S. Kankanhalli, F. Bao, and R. H. Deng, "Anonymous secure routing in mobile ad-hoc networks." in *LCN*. IEEE Computer Society, 2004, pp. 102–108.

[22] R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan, "Sustaining cooperation in multi-hop wireless networks," in *Proceedings of 2nd Symposium on Networked Systems Design and Implementation*. USENIX, 2005, pp. 231–244.

[23] D. Liu, P. Ning, and W. K. Du, "Attack-resistant location estimation in sensor networks," in *IPSN '05: Proceedings of the 4th international symposium on Information processing in sensor networks*. Piscataway, NJ, USA: IEEE Press, 2005.