ACHIEVING CONFIGURABLE LOCATION PRIVACY IN LOCATION BASED ROUTING FOR MANET

Weichao Wang, Cheng Cui
Department of Software and Information Systems
University of North Carolina at Charlotte, NC USA
Email: weichaowang@uncc.edu and ccui@uncc.edu

***Abstract*—Research shows that location based routing can improve the performance and efficiency of communication in mobile ad hoc networks. From another point of view, disclosure of location information can cause a serious privacy risk, especially in environments where different groups of nodes cannot fully trust each other. In this paper, we propose a protocol through which a wireless node can achieve configurable location privacy by distributing location information with different levels of perturbations to different groups of nodes. To achieve this goal, polynomial based personal keys are deployed for group based location information access. A modified location based routing protocol with privacy awareness features is introduced. Authentication mechanisms are designed to protect the genuineness of location information and prevent impersonation attacks. The efficiency and safety of the proposed approach are investigated.**

I. INTRODUCTION

With the proliferation of positioning devices such as GPS, location-based routing (LBR) algorithms and services have been proposed. Since most location based routing protocols avoid the pre-establishment of forwarding paths, investigation has shown that they are more efficient than proactive and on-demand routing protocols.

However, disclosure of location information without safeguard endangers users' location privacy and exhibits a significant potential of abuse. Since the location information in data packets can be read by any node in the network, the privacy concerns must be highlighted. At the same time, malicious nodes can conduct different attacks on location information by impersonating a victim node or generating fake positions. Therefore, an efficient mechanism must be developed to protect location privacy of wireless nodes and defend against various attacks while still enabling them to enjoy the advantages of location based routing.

In this paper, we focus on the environments in which wireless nodes in a MANET want to achieve a configurable tradeoff between location privacy and the efficiency of LBR. Before presenting the details of our approach, we use an example to illustrate the potential applications.

The research is supported in part by NSF award 0754592 and NSA Capacity Building in Information Assurance Project.

We assume that a mobile ad hoc network is formed by military personnel, and LBR is adopted to support communications among them. The network members can be divided into multiple groups based on their ranks and security clearance levels. This classification also determines the accuracy of location information of other nodes that a member can access. For example, a Captain can know the accurate position of a Private but only a rough position of a General. Location information at different accuracy levels provides protection to users' privacy. At the same time, routing protocols must be adjusted to preserve the efficiency of LBR.

Enforcing security and privacy in these environments puts new challenges to researchers. First, it is different from traditional multicast problem. In the studied environments, a wireless node is a member of only one group but it needs to distribute location information at different accuracy levels to different groups. Second, since wireless nodes introduce perturbations to their location information to protect the privacy, routing protocols must be adjusted to accomplish data delivery after the packets are transmitted to only an approximate position. Finally, mechanisms must be designed to defend against various attacks. For example, both location servers and position requesters must be able to verify the origin of the location information.

A straightforward solution to this problem is to deploy a different public/private key pair for each group. For example, a Private may encrypt her/his most accurate position with $Pub_{General}$ so that only Generals can read this information. This approach, although simple, has several disadvantages. First, traditional asymmetric encryption, which usually involves exponential computation, is not efficient for a wireless node when its limited power and computation capabilities are considered. Second, when a group change happens, it will cause a large amount of computation overhead to generate new public-private key pairs. Finally, since public keys are known to every node, we cannot verify the sender of the location information unless additional authentication methods are adopted. In this way, we cannot defend against malicious attacks such as node impersonation and modifications to positions.

In this paper, we propose a new mechanism that inte-

grates perturbation based privacy preservation, polynomial based key distribution, and a modified location-based routing protocol to solve this problem. While a wireless node may get its accurate position through GPS, it will introduce different levels of noises into the information when it is distributed to different node groups. Achieved location privacy can be jointly determined by multiple parameters such as node density, movement patterns, and perturbation levels. Location information is encrypted by symmetric personal keys so that only members of the target group can read the data. The encryption keys are jointly determined by polynomials and node IDs so that it is more difficult for malicious nodes to conduct impersonation attacks. After packets are transmitted to an approximate position, we adopt a second stage routing based on anonymous routing for MANET [1], [2] to accomplish data delivery.

The proposed approach has the following advantages. First, we achieve a tradeoff between configurable location privacy and network performance by allowing wireless nodes to determine perturbation levels by themselves. It is different from previous mechanisms that depend on a centralized, trustworthy location broker. Second, symmetric encryption is more efficient than asymmetric methods and suits wireless networks better. Third, we integrate polynomial based key distribution and hash chains to defend against impersonation attacks and protect the authenticity of location information.

The remainder of this paper is organized as follows. In Section II, we review the previous efforts that contribute to our research. Section III presents the details of our approach. In Section IV, we investigate the overhead, safety, and achieved privacy of our approach. Finally, Section V concludes the paper and discusses future extensions.

II. Related work

Location based routing

Location based routing for MANETs can be divided into two phases. During the location distribution phase, a wireless node sends its position to one or several location servers. During the forwarding strategy phase, an efficient routing method must be adopted to deliver packets to the final destination. Several mechanisms [3] depend on localized broadcast to distribute location information. In Giordano et al. [4], every node is mapped to a Virtual Home Region (VHR) by a hash function. The nodes in VHR are responsible for maintaining location information of the node. Location servers can also be mapped to multiple groups of nodes in a distributed manner to reduce the communication overhead during position queries.

In both Location Aided Routing (LAR) [5] and DREAM [3] packets are forwarded in a zone area. Several mechanisms [6], [7] based on face traversal have been designed to help packets recover from the local minimum. Fang et al. [8] proposed an algorithm to identify local minima and void regions in the network based on the geometric properties of the topology. Survey papers with more details on this problem can be found in [9]–[11].

Polynomial based key management

Polynomial interpolation was first used to implement threshold secret sharing [12]. Staddon et al. [13] proposed a self-healing key distribution mechanism with revocation capability. The group manager uses a bivariate polynomial as a masking function to privately transmit information to group members. Liu et al. [14] proposed a more efficient self-healing group key distribution scheme. Wang et al. [15] proposed a stateless key management mechanism to support both intra and inter group multicast.

Symmetric key based authentication

In [16], the author proposed three authentication methods using symmetric keys. The first one is using diversified keys with challenge and response procedures. This protocol needs a large database of information stored at the verifier side. The second one is using a one time password scheme for authentication and the third one is using one-time signatures. A group of mechanisms based on hash chains and synchronized clocks [17], [18] have been proposed.

Location privacy enforcement

In [19], a middleware architecture and algorithms are designed to enable a centralized broker to adjust the resolution of location information along spatial or temporal dimensions to enforce location privacy. The core idea is input data perturbation or data cloaking. In [20], a secure solution for position aided ad hoc routing is provided based on asymmetric key management. In [21], an onion structure routing protocol combined with one time public-private keys was proposed for anonymous routing. In our approach, personal key shares for symmetric encryption are adopted to protect the location information.

III. Location Based Routing with Configurable Location Privacy

A. Assumptions and model of attackers

We assume that links among wireless nodes in the studied networks are bidirectional. Every node has a permanent ID that is known by all other nodes in the network. We assume that every node in the network is equipped with GPS. Therefore, all nodes know their accurate positions and they have synchronized clocks. All operations described in the protocol will take place in a finite field F_q, where q is a prime number with a large enough value.

We assume that wireless nodes can be divided into multiple groups based on their security clearance levels. To protect location privacy, they have different privileges

to access location information with different perturbation levels. Mechanisms to determine the degree of perturbation and assess achieved privacy will be discussed in Section IV. Secret keys will be deployed to control access to location information. Since a node may change its group during network lifetime, secret keys must be updated to preserve forward and backward secrecy. The key update operations during group changes will be investigated in Section IV.

Threats to the proposed approach may come from both external and internal attackers. We assume that external attackers can eavesdrop on traffic in the network. However, they cannot directly compromise encryption keys or reverse a hash function. Some internal attackers are curious and they try to get access to location information that is beyond their security levels. Other internal attackers are malicious and they try to impersonate other nodes by generating fake location information. Therefore, both confidentiality and authenticity of location information must be protected.

B. Predistributed information

In this part, we introduce the information that is distributed to wireless nodes during the network initiation procedure. We assume that there are n nodes in the network and every node has a permanent ID $s \in (1 \cdots n)$. We assume that every node has a function that can securely generate fake identities for itself. Without losing generality, we assume that the nodes are divided into three groups G_1, G_2, and G_3. We also assume that node i is in G_1.

To protect the confidentiality of location information and control access to it, we use t-degree polynomials $h(x)$ to determine the personal key shares that are used to encrypt the perturbed position information. As a member of G_1, node i must be able to recover the position information that is sent to its group. Therefore, it should be aware of three such functions, $h_{1,1}(x)$, $h_{1,2}(x)$, and $h_{1,3}(x)$. Here the first and second indexes represent the destination and source groups of the location packets, respectively. For example, $h_{1,2}(x)$ is the polynomial to determine the personal key shares of the nodes in G_2 to send location information to G_1. A node u in G_2 will get its personal key share $h_{1,2}(u)$ during the network initiation procedure. When it sends out its location information with the accuracy level for G_1, it will send out (node u for G_1, $E_{h_{1,2}(u)}(node\ u, position, timestamp)$). When node i receives this packet, it can apply u to $h_{1,2}(x)$ to calculate the encryption key and decrypt the packet. To enable node i to distribute its location information to other groups, it will get the personal key shares $h_{1,1}(i)$, $h_{2,1}(i)$, and $h_{3,1}(i)$.

We adopt a variation of TESLA [17] to help wireless nodes authenticate the origin of location information and prevent impersonation attacks. Every node has its own hash chain and it will disclose the entries in the reverse order at a regular time interval. Before a hash entry is disclosed, the knowledge of that entry can be used to verify the identity of a node. We assume that every node has a certificate to prove its ownership of the hash chain. The public key to verify the certificate is given to every node during the initiation procedure. Since the hash entries are disclosed at a regular time interval, a node may have to temporarily buffer a packet before it can verify the sender. Since GPS devices provide synchronized clocks to wireless nodes, we do not need a separate synchronization protocol.

When a group change happens, new polynomials and personal keys must be distributed to preserve forward and backward secrecy. We assume that a special node *Group Manager* (GM) in the network will accomplish this task. Mechanisms to generate a GM in a MANET will be discussed in Section IV. The authenticity of GM's messages will also be protected by a hash chain whose entries are disclosed at a regular time interval.

We assume that during the network lifetime, every group can have at most m times group member changes. We also assume that for any single group, the total number of nodes that change their group membership is smaller than l. We adopt Logical Key Hierarchy (LKH) [22] to distribute new polynomials during group changes. We treat wireless nodes in a group as leaf entries and use them to form a balanced binary tree. Every entry in the tree is assigned a symmetric key. Every node has the keys corresponding to the path between its leaf entry and the root of the tree. We also generate m polynomials with the degree $(t + l)$ for each group. These functions are represented as $f_{w,j}(x)$, $w = 1 \cdots 3$, $j = 1 \cdots m$. Every node will receive $2m$ personal values determined by these functions of other groups. For example, node i in G_1 will receive $f_{2,j}(i)$ and $f_{3,j}(i)$, $j = 1 \cdots m$ during the initiation procedure. These values are used to distribute new personal keys during group changes and the details will be discussed in Section IV.

Table I summarizes the information that is distributed to wireless nodes during the initiation procedure. We use node i as an example and we assume that node i is in G_1.

C. Distributing perturbed location information

Since every node is equipped with GPS, it can get its accurate position in real time. We represent the accurate position of node i as P_i. The node will then add different levels of noises to P_i to generate the perturbed positions for different groups. Following the previous example, we represent them as $P_{i,1}$, $P_{i,2}$, and $P_{i,3}$ respectively.

We adopt a variation of the Virtual Home Region (VHR) [4] to determine the location servers of a node. Every node has a function $vhr(x)$ that will map a node identity to a specific position in the network. Therefore, the virtual home region of node i is a circle area in the network with

TABLE I
PARAMETERS HELD BY NODE i AND THEIR USAGE.

Parameters	Domain	Usage
Location information encryption keys		
$h_{1,1}(x)$	t-degree polynomial in $F_q[x]$	Polynomial to calculate keys for decrypting location information for a node in G_1
$h_{1,2}(x)$	t-degree polynomial in $F_q[x]$	Polynomial to calculate keys for decrypting location information for a node in G_2
$h_{1,3}(x)$	t-degree polynomial in $F_q[x]$	Polynomial to calculate keys for decrypting location information for a node in G_3
$h_{1,1}(i)$	F_q	personal key share to encrypt location information sent to the members of G_1
$h_{2,1}(i)$	F_q	personal key share to encrypt location information sent to the members of G_2
$h_{3,1}(i)$	F_q	personal key share to encrypt location information sent to the members of G_3
Hash chain for node authentication		
hash chain of node i	F_q	allow other nodes to verify packets from i
Key encryption keys for group changes		
Keys in logical key hierarchy	F_q	recover new polynomials
$f_{2,j}(i), f_{3,j}(i), j = 1 \cdots m$	F_q	recover new personal keys

the center (x_i, y_i)=$vhr(i)$ and a radius r. All nodes in the VHR will play the role of location servers of node i and buffer its position information.

We assume that h^1 and h^s are the first and last entry of the hash chain of node i and $h^{j+1} = hash(h^j)$. Node i will disclose the hash entries in the reverse order at a time interval T. Without losing generality, we assume that i will update its position information every $10 \times T$. We use the hash entries to calculate message authentication codes (MAC) of packets to allow other nodes to verify the sender. Therefore, node i will estimate the transmission delay between itself and the servers so that the position update packets will arrive at the servers before the corresponding hash entry is disclosed. For example, the following packet should arrive at the servers before h^{1000} is disclosed.

$$
\begin{aligned}
&((x_i, y_i), \textit{node } i, Q_{i,1} = E_{h_{1,1}(i)}(i, i_1, P_{i,1}, hash(i_1)),\\
&Q_{i,2} = E_{h_{2,1}(i)}(i, i_2, P_{i,2}, hash(i_2)),\\
&Q_{i,3} = E_{h_{3,1}(i)}(i, i_3, P_{i,3}, hash(i_3)),\\
&a^{991} = hash(h^{991}, Q_{i,1}, Q_{i,2}, Q_{i,3}, h^{991}),\\
&a^{992} = hash(h^{992}, Q_{i,1}, Q_{i,2}, Q_{i,3}, h^{992}),\\
&\vdots\\
&a^{999} = hash(h^{999}, Q_{i,1}, Q_{i,2}, Q_{i,3}, h^{999}),\\
&hash(h^{1000}, Q_{i,1}, Q_{i,2}, Q_{i,3}, a^{991}, \cdots, a^{999}, h^{1000})\\
&\textit{timestamp, hash chain certificate for node } i).
\end{aligned}
$$

Here (x_i, y_i) represents the position of VHR$_i$; $Q_{i,1}$ to $Q_{i,3}$ represent the encrypted positions for different groups; i_1 to i_3 are fake identities used later; a^{991} to a^{999} are message authentication code for position requesters; and the next entry is the authentication code for servers.

The packet will be delivered to nodes in VHR$_i$ through location based routing. When a location server of i receives the packet, it will first check its clock to make sure that h^{1000} has not been disclosed. It will then temporarily buffer the packet until it receives the hash entry. When it gets h^{1000}, it will verify its authenticity based on the certificate of node i. This will also allow it to verify that $Q_{i,1}$ to $Q_{i,3}$ and a^{991} to a^{999} are all sent by node i. If the packet passes all examinations, the server will use the new position record to replace the old entry of node i. During the verification procedure, the server does not need to decrypt the perturbed positions.

D. Acquiring position of destination

When node u in G_2 wants to send a packet to node i, it needs to get i's position first. u can use the function $vhr(x)$ to calculate the virtual home region of i and send out a position request to i's location servers at (x_i, y_i). The request packet will contain:

$$\textit{Position request} : ((x_i, y_i), i, (x_u, y_u), \textit{sequence})$$

Here (x_u, y_u) represents the position of node u. Since in the packet u does not disclose its identity, its location privacy will not be violated. This information is also used for location based routing when the reply is sent. The sequence number is used by u to uniquely label a request.

When any node in VHR$_i$ receives this request, it will search in its database to locate the latest position record of the node. It will also estimate the transmission delay between itself and the requester and attach an authentication code. For example, if the reply will reach at u before h^{993} is disclosed, the packet will contain:

$$\textit{Position reply} : ((x_u, y_u), \textit{sequence}, i, Q_{i,1}, Q_{i,2}, Q_{i,3}, a^{993}, \textit{hash chain certificate for } i).$$

(x_u, y_u) will guide the packet to node u and the sequence number will help it link the reply to a request.

When node u receives this packet, it will first check its clock to make sure that the hash entry of i has not been disclosed. It will buffer the packet until it gets a copy of h^{993}. Using this hash value, the authenticity of the encrypted positions can be verified. u will then calculate the personal key share $h_{2,1}(i)$ so that it can gain access to the perturbed position $P_{i,2}$ and fake ID i_2 for G_2.

E. Routing the data packet

After acquiring the position of the destination node i, u can adopt those already-developed location based routing

protocols to deliver data packets to $P_{i,2}$. The data packets will have the format ($P_{i,2}$, i_2, *data message*). Since i_2 is a fake ID randomly generated by node i, external attackers and nodes in other groups cannot link it to the real identity. Our approach does not depend on any specific location based routing protocols and it can smoothly switch among them when a more efficient method is designed.

One problem that we must handle carefully to preserve the efficiency of location based routing is the perturbed position. Since node i has added noises to its real position to construct $P_{i,2}$, we can deliver data packets to only the neighborhood of the destination. A 'phase-two' routing must be conducted to forward the packets from $P_{i,2}$ to node i. We propose to use either localized broadcast or a variation of the anonymous routing protocols for MANET [1], [2] to achieve this goal. The fake ID i_2 will be used by node i to identify those packets sent to it. In most anonymous routing protocols, an intermediate node will only know the temporary IDs of its upper link and down link neighbors along the path but not who the final destination is or how far it is from the destination. Therefore, the location privacy of node i will be preserved.

IV. Discussion and Analysis

In Section III we use an example to illustrate the position distribution, position query, and routing procedures of our approach. In this part, we investigate the key update operations during group changes and the safety, overhead, and achieved privacy of the mechanism.

A. Key updates during group changes

When a group change happens, secret keys must be updated to enforce forward and backward secrecy. We assume that every group can have at most m times group changes and in each time multiple nodes can switch their groups. All group changes can be decomposed into two atomic operations: leaving and joining. Below we use an example to illustrate the leaving event. Operations during a joining event will be very similar.

Distributing new polynomials and LKH

We assume that node u leaves G_2 in the jth group change of G_2. Since u is no longer a member of the group, it should not gain access to the position information for G_2. Therefore, the polynomials $h_{2,1}(x)$, $h_{2,2}(x)$, and $h_{2,3}(x)$ must be updated. As described in Section III, we assume that nodes in G_2 form a logical key hierarchy with the height of H_2. Node u has the secret keys corresponding to the entries on the path between its leaf node and the tree root. We represent these keys as $k_{u,1}$ to k_{u,H_2}. Since every entry on the path between u and the tree root has a sibling node, we represent the secret keys corresponding to these sibling nodes as $\overline{k_{u,1}}$ to $\overline{k_{u,H_2-1}}$ (the tree root does not have a sibling). We call these keys as **complementary keys for** u. Based on the definition of LKH, every remaining node in G_2 will have at least one key in this group. Therefore, $\overline{k_{u,1}}$ to $\overline{k_{u,H_2-1}}$ can be used to distribute new polynomials.

The group manager GM will generate a random key K_2, the new polynomials $h'_{2,1}(x)$, $h'_{2,2}(x)$, and $h'_{2,3}(x)$, and the new keys in LKH $k'_{u,1}$ to k'_{u,H_2}. It will then send out the following packet.

($\mathit{polynomial\ update\ for}\ G_2$,
$E_{\overline{k_{u,1}}}(K_2), E_{\overline{k_{u,2}}}(K_2), \cdots, E_{\overline{k_{u,H_2-1}}}(K_2)$,
$E_{K_2}(h'_{2,1}(x), h'_{2,2}(x), h'_{2,3}(x))$,
$E_{K_2}(E_{k_{u,1}}(k'_{u,1}), E_{k_{u,2}}(k'_{u,2}), \cdots, E_{k_{u,H_2}}(k'_{u,H_2}))$,
authentication code.)

In the first line, GM states the purpose of the packet. In the second line, we use the complementary keys for u to encrypt the secret K_2. Every node in G_2 but u will be able to recover the secret. Using K_2, they can recover the new polynomials $h'_{2,1}(x)$, $h'_{2,2}(x)$, and $h'_{2,3}(x)$ through decrypting the third line. Finally, the new secrets in the LKH are double encrypted by K_2 and the old secrets. Therefore, only the remaining nodes in G_2 that have the old secrets can recover the new keys. The authenticity of the packet is protected by the authentication code. This packet can establish the new polynomials and LKH for G_2.

Distributing new personal keys

Since G_2 has established its new polynomials, nodes in G_1 and G_3 must update their personal key shares. Below we illustrate the procedure of node i in G_1 to get its new personal key $h'_{2,1}(i)$. We assume that w nodes have left G_1 and their identities can be represented as r_1 to r_w. Based on the assumption in Section III, we know that $w < l$. The group manager will send out the following packet.

($\mathit{personal\ key\ update\ for}\ G_1$,
$R_{2,j}(x) = g_1(x) \cdot h'_{2,1}(x) + f_{2,j}(x), r_1, r_2, \cdots, r_w$,
authentication code).

where $g_1(x) = (x - r_1)(x - r_2)\cdots(x - r_w)$.

Now let us consider the operations of node i. It can calculate $R_{2,j}(i)$ and $g_1(i)$ by applying its identity to the polynomials. Since it has received $f_{2,j}(i)$ during the initiation procedure, it can calculate its new personal key $h'_{2,1}(i)=\frac{R_{2,j}(i)-f_{2,j}(i)}{g_1(i)}$. For a different node i' in G_1, since it does not have the value of $f_{2,j}(i)$, it cannot calculate i's personal key. For a node $r_s, s = 1 \cdots w$ that has left G_1, since $g_1(r_s) = 0$, it will not be able to recover its new personal key. In this way, nodes in G_1 and G_3 will get their new personal secrets.

During the key update procedures, the authenticity of the polynomials and secrets is protected by the hash chain

of the group manager. The key update packets can be distributed in the network through true broadcast. Our approach does not require the group changes to be monotonic.

B. Safety of the approach

In the proposed mechanism, we use polynomials to determine personal key shares so that perturbed positions can be encrypted. For an external eavesdropper, since it does not know the polynomials, it cannot get the encryption keys. At the same time, since the fake IDs are randomly generated by the wireless nodes, an eavesdropper cannot link a real ID to a fake ID or any two fake IDs together.

An inside node cannot gain access to perturbed positions for other groups since it does not know their polynomials. Since a malicious node has to collect at least $(t + 1)$ different values to reconstruct a t-degree polynomial, we can control the safety of the mechanism by adjusting this parameter. The security issues during key updates have been investigated in Section IV.A.

We adopt two mechanisms to prevent impersonation attacks. First, the distribution of personal key shares prevents an internal attacker from impersonating another node in the same group. For example, for node u and v in G_2, since they have different personal key shares $h_{1,2}(u)$ and $h_{1,2}(v)$ for G_1, node v cannot send false position information to impersonate node u. Second, we use a variation of the TESLA approach [17] to allow both location servers and position requesters to authenticate the origin of perturbed positions. Since hash functions have the one-way property, knowledge of a hash entry proves its identity before the hash value is disclosed. To reduce communication overhead caused by hash value expiration, we can attach timestamps to position distribution and query packets to improve the estimation accuracy of the end-to-end delay.

Location servers and position requesters have to temporarily buffer the position information before the corresponding hash value is disclosed. Malicious nodes may use this short period of time to conduct DoS attacks by sending a large number of fake position packets. Mechanisms to defend against such attacks are under investigation.

The group manager plays an important role in the proposed approach and different mechanisms can be used to generate it. If a pre-distributed infrastructure exists, the manager generation procedure can take advantage of those special nodes. For example, in a cellular–ad hoc integrated system, the base stations can manage the membership and generate new keys for every group. In a self-organized environment, a variation of the secure leader election algorithms for ad hoc networks [23] can be adopted.

C. Overhead analysis

In this part, we will investigate the storage, computation, and communication overhead of our approach.

As illustrated in Table I, the proposed mechanism will cause a very small amount of storage overhead at each node. If we assume that there are d groups in the network, a wireless node needs to store d t-degree polynomials to calculate personal key shares of other nodes, d personal keys to encrypt perturbed positions, and the first entry of its hash chain. To enable key updates during group changes, it needs to store up to $log(n)$ values in the logical key hierarchy, and $(d-1)m$ values for personal key updates. Therefore, the node needs to store $(d(t + 2) + log(n) + (d-1)m)$ values in F_q for its own operations.

In addition to storing secrets for its own operations, every node has a certain probability to become location servers of other nodes. If we assume that wireless nodes are randomly and evenly distributed in the whole network area S, and the radius of a VHR is r, on average every node will be a location server for $\frac{\pi r^2}{S}n$ nodes. It has to buffer their perturbed positions and respond to location queries.

The description in Section III has shown that our approach avoids those complex operations such as exponential computation. On the contrary, the light weight operations such as symmetric encryption and hash calculation are widely used. There are two operations that we need to pay a close attention. First, a node may need to verify a digital signature when it receives a hash chain certificate for the first time. It can then use the last entry in the hash chain to authenticate other entries. Second, when a new personal key is distributed, a node may need to evaluate several polynomials. We can use the method proposed in [24] to reduce the computation overhead.

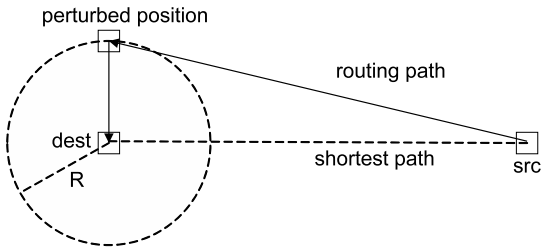

Fig. 1. Route through perturbed position.

The communication overhead at different phases of our approach must be investigated respectively. During the position information distribution and query procedures, we need to transmit more data than traditional LBR protocols since positions with different perturbations must be sent. At the same time, multiple message authentication codes must be sent to allow both location servers and position requesters to authenticate the sender. During the data delivery procedures, more communication overhead will be introduced by the protocol since the data packets will first be transmitted to a perturbed position. If we assume that the perturbed position can be as far as R from the real destination, the data packets will travel up to $2R$ longer than the traditional LBR protocols, as shown in Figure

1. We can adjust the perturbation level R to balance the communication overhead and achieved privacy. A majority of the broadcast traffic during key updates comes from the distribution of polynomials and logical key hierarchy. The choices of the parameters t and l must be jointly considered with the safety of our approach. We can use CKDS [25] to replace LKH to further reduce communication overhead.

D. Achieved privacy

Although our approach does not depend on any specific perturbation methods, in the following analysis we assume the adoption of spatial cloaking and k-anonymity. During the perturbation procedure, we assume that the real position of the node can be replaced by any point within R. If the nodes are evenly and randomly distributed in the network, on average there will be $\frac{\pi R^2}{S}n$ nodes in the perturbation area. If we represent the perturbation area of node u as $pert(u)$, the set of nodes T_u whose perturbation areas overlap with $pert(u)$ can be represented as $T_u = \{v|pert(v)\ overlap\ pert(u)\}$. Therefore, u can generate fake positions to hide itself in T_u. A wireless node can passively estimate the node density and determine the size of its perturbation area to control the size of T.

V. Summary and Future Work

In this paper we propose a mechanism to achieve configurable location privacy in position based routing for MANET. A wireless node can control the perturbation level of its positions and the nodes that can gain access. We present the details of position distribution, position query, and data delivery procedures. We use polynomials and personal keys to control access to position information. Hash chain based authentication is adopted to prevent impersonation attacks. We also present the details of key update operations during group changes. The safety, overhead, and achieved privacy of the approach are investigated.

We propose to extend our approach from the following aspects. First, malicious nodes can still trace the data packets if they can eavesdrop on the whole network and conduct traffic analysis attacks. We propose to study this vulnerability and design prevention methods. We will use the entropy-based measurement to investigate how node movement will impact their location privacy. Finally, we will conduct systematic simulation to investigate the communication overhead and achieved privacy.

References

[1] Y. Zhang, W. Liu, and W. Lou, "Anonymous communications in mobile ad hoc networks," in *IEEE INFOCOM*, 2005.

[2] D. Sy, R. Chen, and L. Bao, "Odar: On-demand anonymous routing in ad hoc networks," in *The Third IEEE International Conference on Mobile Ad-hoc and Sensot Systems (MASS)*, 2006.

[3] S. Basagni, I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward, "A distance routing effect algorithm for mobility (DREAM)," in *MobiCom*, 1998, pp. 76–84.

[4] S. Giordano and M. Hamdi, "Mobility management: The virtual home region," Ecole. Polytechnique Federale de Lausanne, Switzerland, Tech. Rep., 1999.

[5] Y. Ko and N. Vaidya, "Location-aided routing (LAR) in mobile ad hoc networks," *Wirel. Netw.*, vol. 6, no. 4, pp. 307–321, 2000.

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

[7] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *MobiCom*, 2000, pp. 243–254.

[8] Q. Fang, J. Gao, and L. J. Guibas, "Locating and bypassing routing holes in sensor networks," in *Proceedings of INFOCOM*, 2004.

[9] M. Mauve, J. Widmer, and H. Hartenstein, "A survey on position-based routing in mobile ad hoc networks," *IEEE Network Magazine*, vol. 15, no. 6, pp. 30–39, 2001.

[10] S. Giordano and I. Stojmenovic, "Position based routing algorithms for ad hoc networks: A taxonomy," in *Ad Hoc Wireless Networking*, X. Cheng, Ed. Kluwer, 2004, pp. 103–136.

[11] S. Rührup, "Position-based routing strategies," Ph.D. dissertation, University of Paderborn, Germany, 2006.

[12] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[13] J. Staddon, S. Miner, M. Franklin, D. Balfanz, M. Malkin, and D. Dean, "Self-healing key distribution with revocation," in *Security and Privacy. Proc. IEEE Symposium on*, 2002, pp. 241–257.

[14] D. Liu, P. Ning, and K. Sun, "Efficient self-healing group key distribution with revocation capability," in *Proc. of ACM Conf. on Computer and communications security*, 2003, pp. 231–240.

[15] W. Wang and T. Stransky, "Stateless key distribution for secure intra and inter-group multicast in mobile wireless network," *Comput. Netw.*, vol. 51, no. 15, pp. 4303–4321, 2007.

[16] J.-H. Hoepman, "Symmetric key authentication using verification in public," University of Twente, Tech. Rep., 2001.

[17] A. Perrig, R. Canetti, D. Song, and J. D. Tygar, "Efficient and secure source authentication for multicast," in *Network and Distributed System Security Symposium*, 2001, pp. 35–46.

[18] D. Liu and P. Ning, "Multilevel μtesla: Broadcast authentication for distributed sensor networks," *ACM Trans. Embedded Comput. Syst.*, vol. 3, no. 4, pp. 800–836, 2004.

[19] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *Inter. Conf. on Mobile Sys., App., and Services*, 2003, pp. 31–42.

[20] S. Carter and A. Yasinsac, "Secure Position Aided Ad hoc Routing Protocol," in *Proc. of Inter. Conf. on Communications and Computer Networks*, 2002, pp. 329–334.

[21] S. Seys and B. Preneel, "Arm: Anonymous routing protocol for mobile ad hoc networks," in *Advanced Information Networking and Applications, AINA Inter. Conf.*, 2006, pp. 133–137.

[22] C. K. Wong, M. Gouda, and S. Lam, "Secure group communications using key graphs," *Networking, IEEE/ACM Transactions on*, vol. 8, no. 1, pp. 16–30, 2000.

[23] S. Vasudevan, B. DeCleene, N. Immerman, J. Kurose, and D. Towsley, "Secure leader election algorithms for wireless ad hoc networks," in *Proc. of DISCEX*, 2003.

[24] W. Du, J. Deng, Y. S. Han, and P. Varshney, "A pairwise key predistribution scheme for wireless sensor networks," in *Proceedings of ACM CCS*, 2003, pp. 42–51.

[25] M. Moharrum, R. Mukkamala, and M. Eltoweissy, "CKDS: an efficient combinatorial key distribution scheme for wireless ad-hoc networks," in *Proc. of IEEE ICPCC*, 2004, pp. 631–636.