PRISM: Privacy-friendly Routing In Suspicious MANETs (and VANETs)

Karim El Defrawy and Gene Tsudik*
Donald Bren School of Information and Computer Science
University of California, Irvine
{*keldefra,gts*}*@ics.uci.edu*

***Abstract*— Mobile Ad-Hoc Networks (MANETs) are particularly useful and well-suited for critical scenarios, including military, law enforcement as well as emergency rescue and disaster recovery. When operating in hostile or suspicious settings, MANETs require communication security and privacy, especially, in underlying routing protocols. This paper focuses on privacy aspects of mobility. Unlike most networks, where communication is based on long-term identities (addresses), we argue that the location-centric communication paradigm is better-suited for privacy in suspicious MANETs. To this end, we construct an on-demand location-based anonymous MANET routing protocol (PRISM) that achieves privacy and security against both outsider and insider adversaries. We analyze security, privacy and performance of PRISM and compare it to alternative techniques. Results show that PRISM is more computationally efficient and offers better privacy than prior work.**

I. Introduction

Mobile Ad-Hoc Networks (MANETs) play an increasingly important role in many environments and applications, especially, in critical settings that lack fixed network infrastructure, such as: emergency rescue, humanitarian aid, as well as military and law enforcement [1]. Since most MANETs are multi-hop in nature, agile and resilient routing is a crucial function with requirements appreciably distinct from those in fixed networks. To this end, a number of MANET routing protocols have been proposed, ranging widely in assumptions, efficiency and functionality. At the same time, many MANET deployment scenarios involve operation in *hostile* environments, meaning that attacks are either expected or, at least, possible. Moreover, threats can originate from both outside and inside the network. The research community responded to the security challenge with various techniques that mitigate (by prevention and/or detection) a number of potential threats and attacks. Such techniques yielded several security-fortified MANET routing protocols. (See [2] for a comprehensive survey.)

While most prior work in secure MANET routing focused on security issues, less attention has been devoted to privacy. Note that, in this context, privacy does not mean confidentiality of communication (i.e., data) among MANET nodes. The latter is a fundamental part of secure MANET operation; it is easily attained by encryption, assuming that appropriate key management solutions are used to set up or distribute cryptographic keys. What we mean by *privacy* is resistance to tracking. We believe that this narrow interpretation of privacy is well-justified. Since mobility is the only distinctive MANET feature, the sequence of movements by a given MANET node can represent sensitive private information. This is clearly not always the case, i.e., some MANETs do not require privacy of this type. Whereas, any setting where tracking of MANET nodes is undesirable or dangerous would benefit greatly from hiding node movements and movement patterns.

Application Examples: As mentioned above, military and law-enforcement MANETs are compelling examples of settings where privacy, in addition to security, is very important. Zooming in on the military example, one can imagine a battlefield MANET composed of different types of nodes, e.g., infantry soldiers, vehicles, aircrafts as well as other types of personnel and equipment. If the adversary can track nodes' movements, it can easily deduce node types. For example, one that moves 50 miles within 10 minutes is most likely, an aircraft. Whereas, one moving only 5 miles within the same interval is probably a vehicle. Another example in the same setting is an adversary aiming to track specific nodes. If the adversary knows that a certain node corresponds to a commander, it could wait until this node moves within reach of sniper fire, with obvious consequences.

A very different domain where privacy is very important is Vehicular Ad Hoc Networks (VANETs) [3]. Such networks are an emerging phenomenon, increasingly considered in the research literature. While there is no single globally accepted view of VANET characteristics, the common model envisions vehicle-to-vehicle communication for the purposes of sharing traffic, road and weather conditions. In this peer environment tracking-resistance is a natural feature: if tracking of vehicles is possible, there would be considerably less incentive for drivers to share information.

With the focus on privacy, our central goal is to design

*This research was supported in part by an award from the US Army Research Office (ARO) under contract W911NF0410280.

1-4244-2507-5/08/$20.00 ©2008 IEEE

tracking-resistant techniques for MANETs. As discussed below, such techniques can not offer a privacy panacea, since they depend on certain environmental factors, such as sufficient network size and pervasive mobility.

If nodes do not move, tracking-resistance is clearly impossible. This is because an adversary observing successive snapshots of the topology can easily see that certian nodes remain at the exact same positions.

Furthermore, tracking-resistance requires us to reexamine the very basics of MANET communication, e.g., how nodes refer to each other and why they communicate in the first place.

Contributions: This paper makes two contributions. First, it shows how to obtain privacy-friendly on-demand location-centric MANET routing. (By "privacy-friendly" we mean *resistant to node tracking by both outsider and insider adversaries*). Moreover, this is achieved without sacrificing security. Second, it demonstrates – via simulation – that the proposed PRISM protocol offers better privacy and better efficiency than prior results.

Organization: The rest of this paper is organized as follows. We first discuss certain key features of the envisaged MANET setting and justify certain choices in our design in Section II. We present our assumptions and adversary model in Section III. We then describe the details of PRISM and analyze its security and privacy in Section IV. An overview of related work is presented in Section V and PRISM's efficiency is compared through simulation to prior work in Section VI. We summarize our conclusions and discuss future work in Section VII.

II. Design Elements and Choices

A. Goals

Our work has three main goals:

(1) *Privacy:* maximize tracking-resistance of individual nodes, by outsider and insider adversaries.

(2) *Security:* provide protection against active and passive outsider and insider attacks.

(3) *Efficiency:* attain the above two goals with reasonably efficient solutions.

B. Long-Term Identities Considered Harmful

The need for comprehensive addressing is fundamental in most networks. Some form of a unique address (or name) is usually a pre-requisite for one node to communicate with another. However, we argue that in a privacy-conscious MANET setting, using long-term or persistent identifiers can be harmful. The first threat comes from outsiders: tracking nodes based on their identifiers is possible by eavesdropping on routing information exchanged. This can be easily remedied by having all MANET nodes share a network-wide key and encrypting all routing information. The second threat comes from malicious insiders, i.e., MANET nodes that aim to track their peers. This threat is much harder to address, since a typical (even secure) MANET routing protocol is designed to provide routing information based on a destination address.

Of course, MANET routing protocols vary widely in terms of how much topological information is made available. Link-state protocols (e.g., OLSR [4]) reveal the entire topology, whereas, some distance-vector protocols provide no information beyond the hop-count and the next hop for a given destination (e.g., AODV [5]). However, even when minimal information is made available, some tracking is still possible. (Consider that, even in AODV, a node can, at the very least, always discover its immediate neighbors.)

One alternative that offers perfect privacy is to expose no routing information at all and use the simplest form of routing – flooding – for all communication. This would allow the use of long-term identities for addressing while, protecting nodes from being tracked. (Each message is flooded and only the intended destination receives/processes it.) However, this approach is very inefficient since flooding consumes a lot of bandwidth.

C. One-Time Pseudonyms

A natural alternative to long-term persistent identities is short-term (or even one-time) pseudonyms. This general approach has been used in many application domains. However, pseudonyms work well only with proactive protocols. For example, in a link-state protocol such as OLSR, each node propagates its immediate neighborhood information to all other nodes. Thus, if each node has a collection of unrelated pseudonyms, it uses them, one at a time, to avoid being tracked. In a distance-vector protocol, such as DSDV, a node can also periodically switch to a new pseudonym and shed its previous identity. The same does not apply to reactive (on-demand) protocols, such as AODV or DSR, since they predicate route discovery upon knowledge of the destination's identity.

However, using pseudonyms is problematic even in proactive protocols. This is because, any identity (short- or long-term) must be securely bound to some unique cryptographic material, e.g., a public key contained in a public key certificate (PKC) signed by some trusted certification authority (CA). If no such binding exists, privacy is easily attained at the expense of security, since, without individual keys, only security against outsiders is possible. Whereas, if both privacy and security are necessary (as stated in our goals), each pseudonym must be individually certified and bound to a unique public key. Consequently, each node must be issued a large-enough set of one-time pseudonyms and corresponding certificates, which is a viable but an unscalable approach.

Another issue with pseudonyms is that they form a poor basis for communication. Normally, communication is initiated on the basis of long-term identity, i.e., node A decides to communicate with node B. If identities are random and used only once (to maximize privacy), two problems arise. First, how would a given node learn current one-time identities of other nodes? Second, why would a node want to communicate to another node if the latter's identity is a random-looking value indistinguishably from any other node's current identity?

D. Communication Paradigm

As discussed above, our privacy goal dictates that long-term identities can only be used in conjunction with flooding (which is inefficient). Whereas, random short-term (one-time) identities are not meaningful as the sole basis for communication. This leads us to consider a fundamental question:
Is communication identity-centric or location-centric?
The term *identity-centric* means that one node decides to communicate with another based on the long-term identity, regardless of the latter's location, current MANET topology or other ephemeral factors. Location-centric communication means that communication decisions are made largely on the basis of current topology or some other related criteria, e.g., nodes' physical coordinates. We observe that many critical MANET as well as VANET scenarios are not inherently identity-centric. For example, in a disaster relief setting, current node location might be much more important than node identity. The same holds in VANETs where nodes (vehicles) query, collect and disseminate information based on their current locations.[1]

In the rest of this paper, we restrict the scope of our work to MANETs where communication decisions are location-centric.

E. Topology Exposure

Another important privacy issue is topology exposure: *to what degree should the routing protocol advertise the current topology?*[2] Generally, since less information means better privacy, we can conclude that the best approach is to use a reactive (on-demand) routing protocol that hides MANET topology. AODV is a good example – it reveals only the hop-count for a given destination.

Another extreme is a link-state protocol such as ALARM [6] which advertises the entire topology but uses one-time pseudonyms (based on current location and group signatures). Unfortunately, revealing the entire MANET topology can cause a real privacy problem, even if no long-term identities are used. The problem occurs in periods of low mobility (i.e, only a few nodes move between successive topology updates) which makes tracking trivial: the (insider) adversary compares two topology snapshots and – assuming that the few nodes which moved were not clustered – can track all movements by comparing relative displacements between nodes in both snapshots. It is impossible for certain nodes in the first snapshots to have moved to other positions that appear in the second snapshot, due to the movement speed and the time between snapshots.

Since maximizing tracking-resistance is one of the main goals, our present work attempts to minimize topology exposure. However, we acknowledge that this limits applicability since, in some MANET scenarios, communication decisions are made on the basis of both location and topology. In other words, absolute location (current coordinates) and relative location (current position within the network) might both be needed for a source node to pick a destination. We anticipate that some military, law enforcement and emergency rescue scenarios would fall into this category. For example, an emergency rescue MANET set up after an earthquake would want to expose the entire topology to its nodes to make sure that no node is ever alone or is poorly connected to the rest of the network. The situation is very different in other scenarios, such as VANETs. In a VANET, a node (vehicle) would be naturally interested in a certain location (e.g., to inquire about road conditions) and much less concerned with the topology. Another reason not to expose VANET topology is to prevent easy node tracking: vehicles do not move arbitrarily but follow restricted routes corresponding to existing roadways. Finally, a VANET, especially in a city, might be very large, spanning many thousands of nodes at a time; hence, propagating topology information to end-nodes might be simply unscalable.

F. Is Anybody Out There?

If the current MANET topology is unknown and there are no long-term node identities, how do nodes communicate? One possibility is to use a *hit-and-miss* approach, which we adopt in this paper. In it, a node picks a geographical location (coordinates), draws a certain perimeter around it (e.g., by specifying a radius or points

[1] The divide between identity and location-centric communication is not clear-cut. There might be scenarios that require both location and long-term identity for nodes to make communication decisions.

[2] In this context, "advertise" applies to genuine MANET nodes, i.e., we assume that outsiders are unable to obtain topology information.

of a polygon) uses the resulting area as the destination address. The message (route request) addressed in such a way propagates through the network (via flooding, as in AODV) and either fails to find any nodes in the specified area or reaches one or more. Destination node(s) then reply (if they want to) using state along the reverse route, with intermediate nodes using information cached during route request processing.

This simple location-based technique is effective as it guarantees that, as long as the network is connected, all destinations within the specified area are reached. However, it complicates operation since the specified area might be empty. In this case, the source needs to either expand the perimeter or try a different area altogether. Another potential problem is that the destination area might include too many nodes thus resulting in too many route replies. This is a QoS-related issue: in some cases, the source intends to reach all destinations, whereas, it may also wish to reach one or some fixed number.

G. Privacy with Security

As mentioned in Section II-A, we need to find a balance between privacy and security: an ideal solution would be tracking-resistant, immune to insider and outsider attacks (and, hopefully, efficient).

Security and privacy with respect to outsiders is relatively easy to obtain with standard cryptographic techniques: encryption and authentication of routing information and subsequent data packets. Privacy with respect to insiders is much harder to obtain because it runs counter to security: malicious behavior by insiders must be traceable, however, traceability can violate privacy. One way to side-step this conflict is to adopt conditional or escrowed privacy:

> As long as a node behaves correctly, its privacy is assured, i.e., its movements can not be tracked. However, if misbehavior is detected, the identity of the offending node can be later discovered by some trusted off-line authority.

We note that the same approach is taken in the ALARM protocol [6] which utilizes group signatures as a means of obtaining escrowed anonymity. We adopt the same approach, except that, unlike ALARM, we do not mandate the use of group signatures, due to their excessive cost and difficulty of defending against Sybil attacks [7]. (More on this below in Section III-C.)

III. Environment Features

This section describes the essential features of the envisaged MANET environment, including network assumptions, the adversarial model and security infrastructure details.

A. Network Assumptions

Our MANET assumptions are as follows:

- A node has no public identity. There might be a private long-term identity (or address) for each node but this information is assumed to remain private between each node and a trusted off-line authority (see Section III-C).
- All communication is hit-and-miss and location-centric: a source node selects a destination location (area) and attempts to communicate to a destination node (or nodes) at that location. If the specified location is empty, the source node times out. (The source can also optionally specify the maximum number of destinations it wants a response from.)
- The MANET environment is suspicious, meaning that even genuine nodes can not be trusted. (See the next section).
- Each node has a means of determining its location with reasonable accuracy, e.g., a GPS device.
- Nodes are loosely time synchronized; (this feature is "free" with GPS).
- Nodes have uniform transmission range.
- Nodes are capable of generating good-quality random numbers and performing basic public key operations (e.g., encryption and signatures).

B. Adversarial Model

As stated earlier, we are concerned with both outsider and insider adversaries. The outsider can be passive or active and has no fear of detection. Its goal is to violate privacy, security, or both. Outsiders can eavesdrop on all communication in the network, but, as mentioned earlier, simple link encryption prevents that, whereas, appropriate textbook authentication measures can prevent any modification and fabrication of messages.

Therefore, we are mainly concerned with insider threats. The insider is passive and follows the so-called "honest-but-curious" model. This model is well-known in the security literature [8]. Such an adversary outwardly behaves correctly by following all rules and protocols. In other words, it sends no fraudulent messages, does not attempt to impersonate other nodes and does not delete or modify other nodes' traffic. Doing otherwise would bring attention and could result in its eventual detection and exposure. However, the honest-but-curious insider is not assumed to be *silent*, i.e., its communication patterns are not significantly different from those of other non-malicious nodes.

Disclaimer: Our model does not cover an adversary who physically tracks nodes in the field, e.g., visually or by using physical-layer signal finger-printing to identify network interfaces. Furthermore, it does not include adversaries who mount denial-of-service (DoS) attacks by creating sinkholes, wormholes and other topological abnormalities. We also do not consider security against insiders who lie about their current location.

C. Security Infrastructure

We make several assumptions about the MANET security infrastructure. First, we assume the existence of an **off-line** Trusted Third Party (TTP). The TTP performs the functions of a Certification Authority (CA) in addition to other tasks, such as forensic auditing of security logs and after-the-fact tracing of potential misbehavior by MANET nodes (insiders). Second, we assume that, prior to deployment, each MANET node is properly registered with the TTP and is issued sufficient credentials, such as public key (or group signature) certificates. Another TTP responsibility is the creation and distribution of a MANET-wide secret key used for all traffic encryption. This is needed to protect against passive outsiders attempting to eavesdrop on intra-MANET communication. As implied by the above, the TTP is also the only party aware of each node's long-term identity.

IV. PRISM Protocol

This section describes PRISM: Privacy-friendly Routing in Suspicious MANETs. PRISM is an anonymous location-based on-demand routing protocol based on three main building blocks: (1) the well-known AODV routing protocol, (2) any secure group signature scheme, and (3) location information. Location information, as mentioned in Section III-A, is assumed to be available to each node, e.g., via GPS.

A. Why AODV?

AODV [5] presents an attractive foundation for PRISM, for several reasons:

- AODV is on-demand (reactive) and thus does not *propagate* topology information, in contrast with proactive protocols, such as OLSR.
- AODV is distance-vector; it does not return source routes (which reveal partial topology), unlike source-routing-based protocols, such as DSR.
- AODV is robust since it uses flooding for route discovery; thus, it does not require mobility to be synchronized.

We do not describe AODV in detail, since, as an established routing protocol, it is well-known and has been extensively studied.

B. Why Group Signatures?

Group signatures, described in Appendix A, are an appealing building block for anonymous MANET routing, mainly because it satisfies the conditional privacy property mentioned in Section II-G above. Our use of group signatures in the context of anonymous MANET routing is not new, e.g., they were used in ALARM [6] for a very similar purpose. Referring to Section A in the appendix, it is easy to imagine a group signature scheme deployed in a MANET setting, where each node corresponds to a group member and the off-line TTP (see Section III-C) corresponds to a Group Manager (GM).

C. Protocol Description

PRISM is designed with the following features in mind:

- The source authenticates the destination and vice versa. Node authentication means that the node is genuine and can be later identified in the event of misbehavior or disputes.
- Intermediate nodes do not learn current location of the source or the *exact* current location of the destination(s). [3]
- Intermediate nodes are not authenticated. Route length (hop count) is not verified. Albeit, it can be lower-bounded using time, assuming no wormhole attacks.
- After route discovery, all communication between source and destination is encrypted and authenticated using a one-time (session-specific) secret key.
- The TTP (group manager) can later learn claimed locations of all nodes that engage in direct communication, i.e., serve as either sources or destinations.[4] The TTP is thus capable of identifying suspicious or malicious behavior by nodes that generate too many route discoveries or move along implausible trajectories (i.e., lie about their location). This is enabled by having all nodes record all route requests and route replies they process (as source, destination or intermediate nodes) and later off-load the accumulated information to the TTP.

The basic operation of PRISM is similar to AODV. Note that PRISM allows the source to specify a destination area and simultaneously discover multiple destination nodes. However, to keep the description simple and due to space limitations, we assume that at most one node exists within the destination area.

[3] Note that intermediate nodes learn the area requested by the source, thus any destination must be within that area; however, the exact location is not revealed to anyone but the source.

[4] Interim locations of nodes that do not engage in direct communication are not traceable by anyone.

1) The source broadcasts a route request (RREQ) which contains the destination location, in the form of coordinates and a radius – DST-AREA.[5] RREQ also contains a temporary public key PK_{TMP}, a time-stamp TS_{SRC} and a group signature, $GSIG_{SRC}$ computed over all fields.
2) Upon receiving a RREQ, each node first checks if TS_{SRC} is valid. If not, the RREQ is dropped. Next, the node checks whether it has previously processed the same RREQ. This is done by computing a hash of the new RREQ ($hRREQ$) and looking it up in the local cache where all recently handled RREQ hashes are stored. Then, the node checks whether it is within DST-AREA.
   (A) If not, the intermediate node caches $hRREQ$ and re-broadcasts the RREQ. Note that no RREQ fields are changed.
   (B) If the node is within the destination area, it verifies $GSIG_{SRC}$. If invalid, the RREQ is discarded. Otherwise, it stores the entire RREQ (including $GSIG_{SRC}$). This is needed for forensic analysis, in order to identify and track misbehavior. The destination then composes a route reply (RREP) which contains: (1) $hRREQ$, (2) a new random session key K_S and (3) the exact destination location. Both (2) and (3) are encrypted under PK_{TMP} obtained from the RREQ. The RREP also includes the group signature – $GSIG_{DST}$ of all fields. Finally, the destination broadcasts RREP.[6]
3) Upon receiving a RREP, each node checks whether it has cached the corresponding $hRREQ$. If not, the RREP is dropped since this node was not on the forward route. If $hRREQ$ is already cached, the node checks if the same RREP has been processed. (If so, the RREP is dropped.) The intermediate node now creates a new entry in its active routes table and re-broadcasts the RREP. Each active table entry contains: $hRREQ$, $hRREP$ and timestamp of entry creation.
4) When the RREP is received, the source first verifies the group signature. If invalid, the RREP is discarded. Next, the source decrypts the session key and location supplied by the destination. (This key is subsequently used for message encryption and/or authentication.) Next, the source stores the entire RREP for forensic purposes. This completes the route set-up process.

Once the route is established, each source-destination data message specifies the tuple $< hRREQ, hRREP >$ as a unique route identifier. In the opposite direction, $< hRREP, hRREQ >$ is used as a route identifier. If the route breaks, a route error (RERR) message similar to that in AODV is generated.

D. Security Analysis

Passive Attacks: PRISM is immune to passive outsiders, since simple link encryption using a common MANET-wide key prevents eavesdropping. As mentioned in Section III-C, we assume that the TTP sets this up before deployment. Attacks by passive insiders are more worrysome. A passive insider can observe RREQ-s and corresponding RREP-s, which reveal several things:

(1) The timestamp of the RREQ source TS_{SRC} may inform the insider about the distance away from the source, even though the direction is unknown. However, this is easily prevented by using coarsely-granular timestamps, i.e., TS_{SRC} can be expressed in minutes or sufficiently granular units of time such that timing analysis is obviated. It is also possible to get rid of the timestamp altogether. In that case, we would need to introduce link-by-link timestamping and re-encryption, which is cumbersome.

(2) The DST-AREA field in RREQ is *visible* and thus betrays the source's interests. There seems to be no practical way to address this issue, since the content of DST-AREA is precisely what enables routing in PRISM.

(3) The mere existence of a RREP tells the insider that at least one node exists in the destination area specified in RREQ. (Multiple RREP-s provide even more knowledge.) This leaks information about the current topology to passive intermediate nodes. However, recall that the destination's precise location is encrypted and is visible only to the source.

At the same time, a passive insider can not link two requests from the same source. This is due to the basic property of group signatures which makes it infeasible to decide whether two (or more) valid group signatures are generated by the same signer. Moreover, each RREQ includes a unique PK_{TMP} and, once established, each route uses a distinct K_S for traffic encryption and authentication.

Active Outsiders: Since all traffic within the MANET is protected by a group-wide secret key, an active outsider is unable to modify, replay or introduce messages. Specifically, replays are prevented since each RREQ is timestamped and each RREP must correspond to a previous RREQ. (Spurious RREQ-s are simply discarded.)

[5] Without loss of generality, we use a circular area in the description. In practice, any reasonably simple polygon can be used.

[6] Note that, unlike some other anonymous protocols, PRISM does not require the destination to re-broadcast the RREQ or to delay sending the RREP, since any insider "overhearing" the RREP already knows that the the destination is within the area specified in RREQ.

Consequently, the attacker can obtain the group-wide secret key only by compromising a genuine node, which transforms it into an *active insider*.

Active Insiders: PRISM is not secure against active insiders. An active insider can lie about its location and reply to RREQ even though it is not within DST-AREA. This misbehavior might remain undetected, either in real time or later. However, it does not create any loss of privacy. Also, a malicious insider, who is within DST-AREA, can refuse to reply to a RREQ. Again, this does not influence privacy. A malicious intermediate node can also drop route setup messages or otherwise interfere with setup or subsequent data communication. This misbehavior corresponds to DoS attacks which are out of scope of our work, especially, since it has no bearing on privacy. Furthermore, group signatures make Sybil attacks very easy: nothing prevents a malicious insider from responding to numerous RREPs with different false location information; this is due to the unlinkability property of group signatures. However, we stress that this is a security, and not a privacy, issue and can be detected by the TTP.

One real threat to privacy stems from malicious insiders. An insider can continuously and/or excessively probe the topology by generating a multitude of RREQ-s, in an effort to monitor node movements and topology fluctuations. In PRISM, such attacks can not be detected in real time since group signatures are unlinkable. However, off-line, the TTP (Group Manager) can open all group signatures logged by each node and determine the exact long-term identity of each node which generated every RREQ or RREP. Recall that PRISM route setup requires each source and destination to log each valid RREP and RREQ (respectively) and each such message contains a group signature. Every so often (e.g., whenever the period of field deployment ends), each MANET node is expected to off-load all of its accumulated RREP/RREQ pairs to the TTP. The TTP, in turn, performs forensic analysis and identifies proven and suspected misbehavior.

V. Related Work

We now briefly survey related work. This section is particularly terse due to submission space restrictions.

Secure MANET routing has been extensively studied in both security and networking research communities. A comprehensive survey of this work can be found in [2]. Well-known on-demand protocols include: SRDP [9], Ariadne [10], SEAD [11] and endairA [12]. All of them focus on security of route discovery, route maintenance and defending against modification and fabrication of routing information. Privacy – especially, tracking-resistance – is not one of the goals.

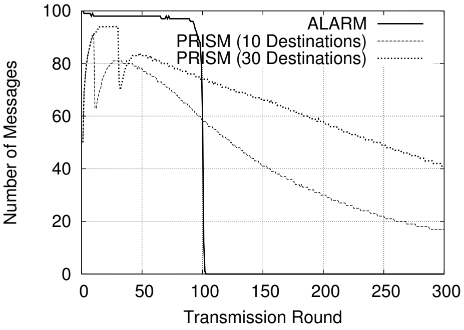

Fig. 1. Number of Routing Messages vs Time (RWM)

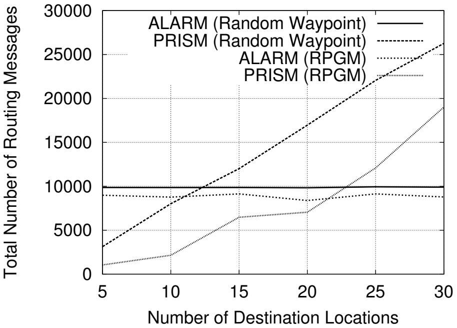

Fig. 2. Total Number of Routing Messages vs Number of Destinations

A more relevant body of research tackles anonymous on-demand MANET routing, e.g., SPAAR [13], AO2P [14], ASR [15], MASK [16], ANODR [17], D-ANODR [18], ARM [19], ASRP [20] and ODAR [21]. The only proactive anonymous MANET routing protocol is the link-state ALARM [6].

Of the on-demand protocols, SPAAR [13] and AO2P [14] require on-line location servers. ASR [15] and ARM [19] assume that each authorized source-destination pair pre-shares a unique symmetric key. ASRP [20] assumes that each source-destination pair shares some secret information, which could be the public key of the destination or a symmetric key. ANODR [17] assumes that the source shares some secret with the destination for the construction of a trapdoor, for example the destination's TESLA secret key. SDAR [22] assumes that the source knows the public key of the destination, obtained from a certification authority (CA), and ODAR [21] requires an on-line public key distribution server. MASK [16] and D-ANODR [18] contain the final destination in the clear in each RREQ message.

PRISM is fundamentally different from all prior anonymous on-demand MANET routing protocols on two accounts:

(1) PRISM uses a location-centric, instead of an identity-centric, communication paradigm. Therefore, it does not assume any knowledge of long-term node identifiers or public keys.

(2) PRISM requires neither pre-distributed pairwise shared secrets nor on-line servers of any kind.

As an on-demand protocol, PRISM is also very different from ALARM [6], even though the latter uses group signatures and is also location-centric. First, ALARM is a link-state protocol and exposes the entire topology to all insiders. Second, ALARM assumes a restrictive and arguably unrealistic *leapfrog* mobility model, whereby synchronized periods of mobility are alternated with (also synchronized) periods of rest. In contrast, PRISM makes no assumption about the mobility model and does not expose network topology. However, one advantage of ALARM is that, unlike PRISM, it uses the exact destination address due to global knowledge of current topology. Recall that PRISM uses the hit-and-miss approach to destination selection.

VI. Performance and Simulations Results

We simulate PRISM and compare it with ALARM. We did not compare PRISM with other anonymous on-demand protocols since most of them are identity-centric. Other location-centric on-demand routing prtocols might exhibit similar performance, but with lower security and privacy. ALARM, on the other hand, is the only other anonymous location-centric MANET routing protocol.

The goal of the simulations is two-fold: (1) determine which communication patterns best suit PRISM if privacy is not the main concern, and (2) determine how much of the network topology is leaked by PRISM.

We use two mobility models in our simulations: (1) random waypoint model (RWM) [23] and (2) reference point group mobility model (RPGM) [24]. Next, we describe simulations results for each model.

1) Random Waypoint: We simulate a 100-node MANET within a $100m^2$ area. All nodes have a uniform transmission range of $15m$. These parameters ensure that each node on average reaches over 90% of other nodes.[7] Nodes move according to the RWM model. We define the destination area as a circle with a center (x, y) and a radius of 2m. We simulate two cases, both with 50 sources. Each source communicates with 10 random destinations in the first case, and, with 30 in the second.

Figure 1 shows the number of routing messages. This includes location announcement messages (LAM-s) in ALARM and RREQ/RREP messages in PRISM. The number of messages in ALARM is almost constant but decreases rapidly around the 100-th transmission round when nodes have transmitted the received LAM-s. Due to the protocol's link-state nature, this number of messages in ALARM is independent of the number of actual communicating node-pairs. In PRISM, the number of routing messages depends on the number of communicating node-pairs. If 50% of the nodes (i.e., 50) communicate to 10% (or 30%) the peak number of routing messages is smaller than in ALARM, it peaks at the 10th (and 30th) transmission round and then decays slowly. We conclude that, while ALARM generates a flurry of routing messages, it does so within short periods of time and the load is independent of the number of communicating node-pairs. Whereas, PRISM has a lower peak load, but takes longer even when only half the nodes each communicate to only 10 destination areas. ALARM is thus better-suited for scenarios where the network is connected and each node communicates with a large fraction of the other nodes. If each node needs to communicate only with a few others, PRISM is a better choice in terms of traffic load and better privacy (since it reveals less topology information than ALARM).

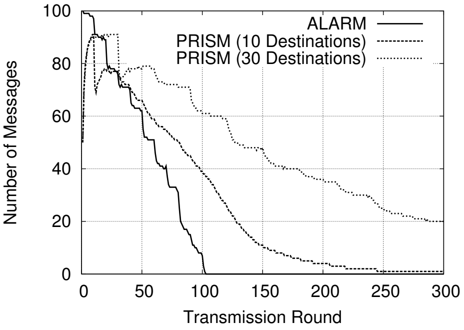

Fig. 3. Number of Routing Messages vs Time (RPGM)

Next, we varied the number of destination areas in PRISM between 5 and 30, in increments of 5. The purpose is to determine the communication patterns (if any) for which PRISM generates less routing overhead than ALARM. We measure the total number of routing messages in both protocols. The result in Figure 2 shows that, in PRISM, if a source communicates to less than 12 destinations, the total number of messages is lower than in ALARM (e.g., for 5 destinations per source, PRISM generates less than half the number of messages in ALARM).

2) Group Mobility: We repeat the same set of simulations with the RPGM model. Nodes are divided into groups, each with a logical center that defines movement

[7] This is determined from simulating how the connectivity of the network varies with different transmission ranges and number of nodes.

for the entire group, i.e., speed, acceleration and direction. We have 10 groups, each with a maximum radius of 50m. Figure 3 shows results for two cases for 10 and 30 destinations per source, respectively. Depending on how close the groups are to each other, some groups receive each other's transmissions and forward them. However, as results show, the number of messages drops gradually. PRISM's route discovery takes longer and requires more messages, although, the peak number of messages sent simultaneously is lower than in ALARM. Compared to the random waypoint simulation, ALARM dips below PRISM earlier (in terms number of simultaneous number of messages) because the network is less connected under the RPGM.

Next, as before, we vary the number of destinations in PRISM between 5 and 30, in increments of 5. The total number of routing messages in both protocols is shown in Figure 2. Results indicate that, in PRISM, when sources communicate to less than 22 destinations each, the total number of messages is less than in ALARM. In particular, for 10 destinations per source, PRISM generates less than 20% of the number in ALARM.

3) Privacy: PRISM generally offers a higher level of privacy than ALARM. This is because a passive insider in ALARM obtains the entire topology. In PRISM, a node only obtains a partial view of the network. The degree of exact topology exposure for a passive insider in PRISM depends on two factors: (1) how many RREQ/RREP pairs it forwards, and (2) how many RREQ-s it generates. Note that even unsuccessful RREP-s divulge information since the source discovers that a given destination area is empty or disconnected.

We show the fraction of topology exposure to insiders with different number of destinations per source. Figure 4 shows simulation results with the same settings as above. In ALARM, on average, each node always knows 97% of the topology in the RWM case (50% in PRGM because the network is fragmented). In PRISM, if each node only communicates to 5 random destinations, it knows, on average, only 16% of the entire topology in the RWM case (less than 1% in PRGM). If we increase the number of destinations to 15, the toplogy exposure jumps to 50% in the RWM case (less than 5% in PRGM). Increasing it further to 35 leads to about 90% topology exposure in the RWM case (17% in PRGM) [8].

VII. Conclusions and Future Work

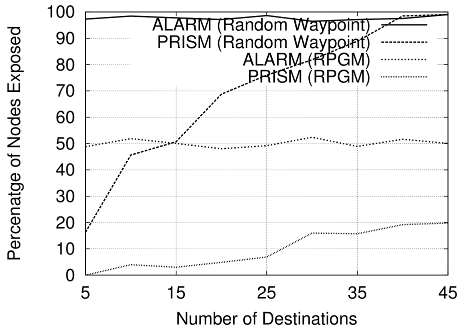

Fig. 4. Percentage of Nodes Exposed vs Different Destinations

This paper describes an on-demand anonymous MANET routing protocol (PRISM) with strong privacy and security features. PRISM is resistant to node tracking by both outsider and insider adversaries. Unlike prior results, it requires no on-line servers, no long-term node identities and has no mobility restrictions. PRISM also does not unduly expose the network topology. Our simulation results compare PRISM with an alternative location-centric link-state approach and show that PRISM generally achieves better performance under reasonable communication assumptions.

A number of items remain for future work. First, we need to specify protocol extensions for the cases of multiple receivers within the destination area. Also, we need to conduct additional simulations with larger-scale parameters (more nodes, greater area of movement) and compare PRISM to other on-demand anonymous MANET routing protocols.

Acknowledgment

The authors are grateful to Claude Castelluccia for comments on the draft of this paper.

References

[1] B. Hartzog and T. Brown, "Wimax- potential commercial off-the-shelf solution for tactical mobile mesh communications," *Milcom*, 2006.
[2] H. Yih-Chun and A. Perrig, "A survey of secure wireless ad hoc routing," *IEEE Security & Privacy*, 2004.
[3] M. Raya, P. Papadimitratos, and J. Hubaux, "Securing vehicular communications," *IEEE Wireless Comm. Magazine*, vol. 13, no. 5, pp. 8–15, 2006.
[4] T. Clausen and P. Jacquet, "Optimized link state routing protocol (olsr)," *IETF RFC 3626*, 2003.
[5] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," *IEEE WMCSA*, 1999.
[6] K. El Defrawy and G. Tsudik, "Alarm: Anonymous location-aided routing in suspicious manets," *IEEE ICNP*, 2007.
[7] J. R. Douceur, "The sybil attack," *IPTPS*, 2001.
[8] L. Kissner and D. Song, "Privacy-preserving set operations," *CRYPTO*, 2005.
[9] J. Kim and G. Tsudik, "Srdp: Securing route discovery in dsr," *MobiQuitous*, 2005.
[10] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Ariadne: A secure on-demand routing protocol for ad hoc networks," *Wirel. Netw.*, vol. 11, no. 1-2, pp. 21–38, 2005.

[8] Note that the results depend on the radius of the destination area specified in a RREQ. For practical reasons, we expect this radius to be much less than the transmission range.

[11] Y.-C. Hu, D. Johnson, and A. Perrig, "Sead: secure efficient distance vector routing for mobile wireless ad hoc networks," *IEEE Workshop on Mobile Computing Systems and Applications*, 2002.

[12] G. Acs, L. Buttyan, and I. Vajda, "Provably secure on-demand source routing in mobile ad hoc networks," *IEEE Transactions on Mobile Computing*, 2006.

[13] S. Carter and A. Yasinsac, "Secure position-aided ad hoc routing," *CCN*, 2002.

[14] X. Wu and B. Bhargava, "Ao2p: ad hoc on-demand position-based private routing protocol," *IEEE Transactions on Mobile Computing*, 2005.

[15] B. Zhu and Z. W. et al., "Anonymous secure routing in mobile ad-hoc networks," *IEEE International Conference on Local Computer Networks*, 2004.

[16] Y. Zhang and W. L. et al., "Mask: anonymous on-demand routing in mobile ad hoc networks," *IEEE Transactions on Wireless Communications*, 2006.

[17] J. Kong and X. Hong, "Anodr: anonymous on demand routing with untraceable routes for mobile ad-hoc networks," *MobiHoc*, 2003.

[18] M. J. L. Yang and S. Wetzel, "Discount anonymous on demand routing for mobile ad hoc networks," *SECURECOMM*, 2006.

[19] S. Seys and B. Preneel, "Arm: Anonymous routing protocol for mobile ad hoc networks," *International Conference on Advanced Information Networking and Applications*, 2006.

[20] Y. Cheng and D. Agrawal, "Distributed anonymous secure routing protocol in wireless mobile ad hoc networks," *OPNETWORK*, 2005.

[21] D. Sy, R. Chen, and L. Bao, "Odar: On-demand anonymous routing in ad hoc networks," *IEEE MASS*, 2006.

[22] A. Boukerche and K. E.-K. et al., "An efficient secure distributed anonymous routing protocol for mobile and wireless ad hoc networks," *Elsevier Computer Communications*, 2005.

[23] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communications & Mobile Computing (WCMC)*, no. 5, pp. 483–502.

[24] X. Hong, M. Gerla, G. Pei, and C. Chinag, "A group mobility model for ad hoc wireless networks," *ACM/IEEE MSWiM*, 1999.

[25] D. Chaum and E. V. Hejst, "Group signatures," *EUROCRYPT*, 1991.

[26] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," *CRYPTO*, 2004.

[27] J. Camenisch and A. Lysyanskaya, "Signature schemes and anonymous credentials from bilinear maps," *CRYPTO*, 2004.

[28] X. Ding, G. Tsudik, and S. Xu, "Leak-free group signatures with immediate revocation," *IEEE ICDCS*, 2004.

[29] J. Furukawa and H. Imai, "An efficient group signature scheme from bilinear maps," *IEICE TFECCS*, 2005.

APPENDIX

A. Group Signatures: Overview

In this section we briefly describe group signatures for the sake of completeness; those familiar with group signatures may wish to skip this section with no loss of continuity.

Group signatures can be viewed as traditional public key signatures with additional privacy features. In a group signature scheme, any member of a (potentially large and dynamic) group can sign a message, producing a *group signature*. A valid group signature can be verified by anyone who has a copy of a constant-length group public key. A valid group signature implies that the signer is a *genuine* group member. However, given two valid group signatures, it is computationally infeasible to decide whether they are generated by the same or different group member(s). However, if a dispute arises over a group signature, a special off-line entity – called a Group Manager – can "open" a group signature and identify the actual signer. This important feature is referred to as *Escrowed Anonymity* or, equivalently, *Conditional Anonymity*.

Group signatures were first introduced by Chaum and Van Hejst [25] and a number of schemes (e.g., [26], [27], [28]) varying in assumptions, complexity and features have been proposed since. Any group signature scheme distinguishes among (at least) three types of entities:

- Group Manager (GM): entity responsible for administering the group: initializing the group as well as handling member joins and leaves (revocations). GM is also responsible for de-anonymizing a signature in case of a dispute.[9]
- Group Members: users/members that represent the current set of authorized signers. In our case, a signer/member is a legitimate MANET node. Each member must have the common group public key and a unique private key that allows it to sign on behalf of the group.
- Outsiders: any other user/entity external to the group. Outsiders are assumed to possess the group public key and are thus able to verify group signatures.

Each group member must have a secret long-term identity which is tied to the group and to the member's unique private key. However, only the GM knows the relationship between the group members and their long-term identities.

A group signature scheme consists of the following:

- SETUP: A probabilistic polynomial-time algorithm, run by the GM, that outputs a cryptographic specification for the group, including the group manager's public and private keys.
- JOIN: A protocol between the GM and a new user that results in the latter becoming a group member. The output of this protocol includes some private output for the new member, including her secret membership key.
- SIGN: An algorithm, executed by any group member, that, on input of: a message, a group public key and a member's private input, outputs a group signature.
- VERIFY: An algorithm, run by anyone, which, on input of: a message, a group public key and a group signature, outputs a binary flag indicating the validity of the said group signature.
- OPEN: An algorithm, run by the GM, that on input of: a message, a group signature, a group public key and GM's secret key, verifies the group signature and returns the signer's identity along with a proof that allows anyone to verify the group identity of the actual signer.
- REVOKE: An algorithm, performed by the GM, to remove (revoke) a user from the group. It typically results in a new group public key and/or a set of auxiliary information for either signers or verifiers (or both).

Some recently proposed group signature schemes require less than 10 exponentiations to sign [29]. Though still more expensive than regular signatures, group signatures are rapidly becoming practical considering fairly powerful laptop-class nodes in MANETs.

[9] Sometimes, the task of adding new members is given to a separate entity called a Membership Manager. Similarly, revocation duties are sometimes delegated to a separate Revocation Manager. In this paper, however, for simplicity's sake, we use a unified off-line GM for all of these tasks.