Privacy Preserving Communication in MANETs

Heesook Choi, Patrick McDaniel, Thomas F. La Porta
Department of Computer Science and Engineering
The Pennsylvania State University
E-mail:{hchoi,mcdaniel,tlp}@cse.psu.edu

Abstract— **Mobile ad hoc networks often support sensitive applications. These applications may require that users' identity, location, and correspondents be kept secret. This is a challenge in a MANET because of the cooperative nature of the network and broadcast nature of the communication media. In this paper, we propose a Privacy Preserving Communication System (PPCS) which provides a comprehensive solution to anonymize communication end-points, keep the location and identifier of a node unlinkable, and mask the existence of communication flows. We present an analysis of the security of PPCS against passive internal attackers, provide a qualitative discussion on its strength against external attackers, and characterize its performance trade-offs. The simulation results demonstrate that PPCS has only 3% lower packet delivery ratio than existing multi-path routing protocols, while effectively providing privacy service in MANETs.**

I. Introduction

In MANETs, mobile nodes cooperate to forward data on behalf of each other. Typical protocols used for self-organizing and routing in these networks expose the node identifiers (network and link layer addresses), neighbors, and the end-points of communication. Some modes of operation further mandate that the nodes freely divulge their physical location. In short, nodes must advertise a profile of their online presence to participate in the MANETs. This is, in many cases, highly undesirable.

Both military and civilian MANETs may find the mandated exposure of information unacceptable. For example, in a military setting, identities of officers and soldiers, their locations, and their communication patterns are critically sensitive intelligence. Civilian applications have similar concerns. Consider students communicating on campus: it is neither desirable nor appropriate for students to expose who they are or where they are to the larger campus community.

Ideally, a node should be able to keep its identity, its location and its correspondents private, i.e., remain *anonymous* [4], [22], [23]. Any solution providing anonymity must overcome the broadcast nature of wireless environments (which enables eavesdropping) and operate under often tight resource constraints. Past "wired world" privacy solutions do not map well to MANETs because of the processing requirements they place on the nodes. Simple solutions like packet encryption are also largely ineffective because of ease of traffic analysis over a broadcast media. Hence, supporting privacy in MANETs is enormously challenging.

In this paper, we propose a Privacy Preserving Communication System (PPCS) which provides a comprehensive solution to anonymize communication end-points, keep the location and identifier of a node unlinkable, and mask the existence of communication flows.

To realize this level of privacy, we propose a series of lightweight cryptographic techniques. These are effective at combating eavesdropping by individual nodes. To further defend against more sophisticated collaborative attacks via traffic analysis, we introduce a resilient packet forwarding scheme. To evaluate the effectiveness of PPCS, we define the optimal guessing strategy that may be used by one or more adversaries in cooperation and show that with PPCS, the probability of correctly guessing the source or destination of a flow is independent of the number of compromised nodes on the path. Even in this case, the adversary cannot confirm that it has guessed correctly, and it cannot learn the real identifier of the source or destination. To quantify the overhead of this solution, we perform extensive simulations that show that there is minimal impact on packet delivery.

This paper is organized as follows: Section II describes the network model and examines passive attacks. Section III presents an anonymous communication system (PPCS). Section IV inspects the effectiveness of an adversary in PPCS. In Section V, we evaluate the performance impact of PPCS. In Section VI, we discuss the trade-offs of PPCS. Section VII reviews previous work on anonymity in Internet and MANETs.

II. Network and Threat Model

A. Network Model

We assume that the wireless interface between nodes is bidirectional, i.e., if node i hears the transmission of node j, then node j is also able to hear node i.

We assume that there exists a symmetric key management service to establish pair-wise keys between nodes,

and that the source and destination establish symmetric keys prior to communications. Such services are well studied in ad hoc and sensor networks [8], [28], [3], [7], [16], and their design is explicitly outside the scope of this work. The source and destination know each other's real identifier. Non-compromised nodes in the network do not disclose any information to compromised nodes beyond what is required for the normal operation of the network.

Note that the privacy services we propose operate solely on the network layer; we assume that the contents of the communication have been duly masked, e.g., via end-to-end encryption.

We use the following notation throughout:

- S: Identifier of a source
- D: Identifier of a destination
- n: Average number of neighboring nodes in transmission range
- P_{Si}: Source S's i-th flow pseudonym
- P_{Di}: Destination D's i-th flow pseudonym
- K_{ij}: Symmetric key between nodes i and j
- $E_{K_{SD}}(.)$: Encryption with a key K_{SD}
- $D_{K_{SD}}(.)$: Decryption with a key K_{SD}

B. Threat Model

We adopt Diaz et al.'s [1] classification of adversaries based on the following characteristics: Internal-External, Passive-Active, and Local-Global. We define an internal adversary as a node that is compromised and on the routing path. An external adversary is a compromised node not on the path, or an external node not directly participating in the MANET, i.e., it only eavesdrops on traffic between nodes.

This paper only considers passive attacks, i.e., attacks that consist of eavesdropping on communications to collect private data. A local adversary can see and launch attacks in a limited range. A global adversary covers the entire path or the network. A set of colluding local adversaries may form a global adversary by sharing information. We defer the active attacks to future work.

Traffic analysis is often used to subvert anonymity [1], [24], [21]. In this attack, adversaries monitor packet transmission to infer important information such as a source, destination, and source-destination pair. We consider the following traffic analysis attacks in this work:

Packet Tracing Attack: A packet may be traced from source to destination by eavesdropping the transmission of the same packet as it traverses the network. Note that the adversary need not be able to recover the packet content to infer the source and destination of the flow.

Packet Counting Attack: Eavesdropping nodes collaborate to discover a path by overhearing and simply "counting" packets that traverse nodes. In a network with low load, this is a straight-forward way to discern data paths.

Timing Attack: Adversaries may analyze the time correlation between packets passing through nodes to discover a flow [15]. If two adversaries perform this analysis and compare results, they may infer a source-destination pair.

TTL Attack: Adversaries exploit the packet time-to-live (TTL) field to discover the destination. The value of the TTL field in a packet is set by a source to limit the number of hops a packet takes in the network. Every intermediate node decreases the TTL by 1 before it forwards the packet. Because this information is sent in the clear, adversaries may determine the relative position of a node on a path, and perhaps the source or destination if they are located near these nodes.

Adversaries may also try to discover information about paths of which they are a part. Many routing protocols expose control information, such as the source and destination or the other nodes on the path, to all nodes on a path. Nodes can also typically overhear the next-hop node on a path as it forwards a packet. Combining this information, adversaries on a path can learn source-destination pairs, next hop nodes, and the entire path of a flow.

Mobile nodes may obtain their own location information using global positioning system (GPS) or other similar techniques. If a node knows the identifiers of its neighboring nodes, it also may estimate their locations. An adversary may also use location information to launch various attacks by tracing an object's location. Therefore, dissociation of location and identity is an important issue.

III. Privacy Preserving Communication

In the following subsections, we present a privacy preserving system which is composed of three mechanisms to anonymize the communication in MANETs. *Dynamic Flow Identification* is aimed at preventing identification of source-destination pairs. *Random Node Identification* dissociates the identity and location of nodes. *Resilient Packet Forwarding* is targeted at thwarting sophisticated traffic analysis attacks.

A. Dynamic Flow Identification

Traditional MANET routing protocols require each control and data packet to contain the source and destination addresses to find a route and identify a flow. With this general approach, an adversary close to the source or destination, or an adversary on the communication path between the two, will be able to link the correspondents, and perhaps learn their location.

To define a flow without releasing the source and destination addresses, we propose a dynamic flow identification scheme based on forward chaining. In the dynamic flow identification scheme, two flow pseudonyms, P_{Di}

and P_{Si}, are defined for the forward and backward flows respectively. The flow pseudonym replaces the source and destination addresses in the packets. A source broadcasts a RREQ packet which contains these flow pseudonyms, <RREQ, P_{Si}, P_{Di}, $E_{K_{SD}}(.)$>.

Intermediate nodes receive a RREQ packet and check if they are the destination by attempting to successfully decrypt and interpret the flow pseudonyms, i.e., "open the trapdoor" [12], which conceals the source and destination address as decribed below. If they are not the destination, they add a routing table entry for the backward flow identified by the flow pseudonym P_{Si} in the RREQ. A destination receives a RREQ and determines that it is the destination by checking the received P_{Di}.

Since each node must perform the trapdoor check, it is important for the check to be efficient. The initial flow pseudonyms, P_{D0} and P_{S0}, of the forward and backward flows are generated by using the symmetric key and real identifiers of the source and destination. Either a source or a destination can change the flow pseudonym at anytime. To do this, subsequent flow pseudonyms are generated based on the previous flow pseudonym using *forward chaining* as follows:

$$P_{S0}=f_{K_{SD}}(S)\rightarrow P_{S1}=f_{K_{SD}}(P_{S0})...\rightarrow P_{Sn}=f_{K_{SD}}(P_{Sn-1})$$

$$P_{D0}=f_{K_{SD}}(D)\rightarrow P_{D1}=f_{K_{SD}}(P_{D0})...\rightarrow P_{Dn}=f_{K_{SD}}(P_{Dn-1})$$

, where f is a cryptographic keyed one-way hash function (HMAC [13]). The results of function f appear random to the intermediate nodes. The trapdoor check is very lightweight, consisting only of computing a hash and a simple search for a matching node. Also note that the trapdoor check only occurs when processing the RREQ message; once the flow has been routed, the check is not required for forwarding subsequent packets. To further improve the efficientcy of the trapdoor check in each node, an optimal data structure such as binary search tree can be used.

B. Random Node Identification

Location privacy requires node identity and location to be unlinkable and untraceable. We propose to use a random node identifier to dissociate a real node identifier from location information. In normal operation, a mobile node has two addresses: a layer 2 address (MAC address) and a layer 3 address (node identifier).

Every node in the network generates random layer 3 and MAC addresses, referred to as random node identifiers (RNI), and advertises itself using its RNI via a message such as a HELLO message in AODV [19]. Neighboring nodes know each other only through their RNIs. The RNI is *locally* used for routing and communicating with neighboring nodes.

Each node changes its RNI after a random interval to prevent an adversary from learning its location and then starts advertising itself with the new RNI. The protocol to change RNI is the same as for an update due to mobility.

Since the source and destination associate with one another using end-to-end flow psuedonyms described in the previous subsection, they do not have to know each other's RNI. This has two benefits. First, the RNI may be changed without end-to-end coordination. Second, since the source and destination do not know each other's RNI, the communication between a source and destination does not disclose the location of either party to the other.

Due to the randomness and independence of the new and old RNI, an adversary cannot trace the changes of node RNI. One risk with this approach is identifier collision, in which two nodes choose the same RNI, might occur. However, the probability that two nodes generate the same RNI (MAC address (48 bits) and layer 3 address (32 bits)) is statistically insignificant, ($\frac{1}{2^{48}}\frac{1}{2^{32}}=\frac{1}{2^{80}}$).

C. Resilient Packet Forwarding

To combat traffic analysis attacks by eavesdropping nodes we propose a resilient traffic forwarding scheme which is composed of multi-path random forwarding (MPRF), Hint, and random TTL (RTTL).

Multi-Path Random Forwarding (MPRF): In a relatively stable network (mild traffic load and low mobility), a path between a source and destination may be used for an extended period of time. This type of path, in particular, is susceptible to a traffic analysis attack. To thwart attacks on a single path, MPRF establishes multiple paths between the source and destination. For each packet, an intermediate node en route randomly selects a next hop from its local list of possible next hop nodes, and forwards the packet to the selected node. Thus, a path that a packet takes is decided dynamically at each intermediate node.

Multi-path routing protocols have been proposed for improving reliability and providing quality of service in ad hoc networks [14], [17], [26]. These multi-path routing protocols establish link/node disjoint paths to distribute traffic to avoid congestion. However, node/link disjoint paths are also vulnerable to traffic analysis attacks. Collaborating eavesdroppers may easily obtain exact packet counts and reconstruct the end-to-end paths. To resolve these vulnerabilities and establish a sufficient number of multiple paths, we relax the node/link disjointness condition present in most multi-path routing protocols. By allowing non-disjoint paths, MPRF diffuses traffic in an irregular manner making traffic analysis more difficult, i.e., requiring a larger number of colluders. In addition, when a node selects multiple paths, the most recently joined node is not be chosen since compromised nodes

can continuously change their identifiers to hamper the communication (Denial Of Service).

Hint: Although a packet is encrypted by a source, if the encrypted packet is transmitted without any modification on each link, it is vulnerable to traffic analysis attacks which determine a data path by observing the incoming and outgoing packets of nodes. To address this problem, the encrypted packet is transformed on a hop-by-hop basis.

To make the hop-by-hop transformation more efficient and anonymous, we propose an HMAC [13] based scheme, called *Hint*. An intermediate node randomly selects a next hop node according to MPRF. It encrypts a packet using a shared key with the selected node and computes an HMAC over the encrypted packet. This HMAC result is called the **Hint**. Then it broadcasts the packet which consists of the Hint and encrypted packet. As an example, the following shows **Hint** operations of each intermediate node:

N_S**:** $C = E_{K_{SD}}(Data)$
$MC =< P_{S0}, P_{D0}, TTL, C >$
$EL_S = E_{K_{N_S N_1}}(MC)$, Hint $= HMAC(K_{N_S N_1}, EL_S)$
Broadcast $< Hint, EL_S >$
N_1**:** $MC = D_{K_{N_S N_1}}(EL_S)$
$EL_1 = E_{K_{N_1 N_2}}(MC)$, Hint $= HMAC(K_{N_1 N_2}, EL_1)$
Broadcast $< Hint, EL_1 >$
N_2**:** $MC = D_{K_{N_1 N_2}}(EL_1)$
$EL_2 = E_{K_{N_2 N_D}}(MC)$, Hint $= HMAC(K_{N_2 N_D}, EL_2)$
Broadcast $< Hint, EL_2 >$
N_D**:**
$MC = D_{K_{N_2 N_D}}(EL_2) \rightarrow MC =< P_{S0}, P_{D0}, TTL, C >$
$Data = D_{K_{SD}}(C)$

Neighboring nodes check if a received packet is for a flow which they serve by simply computing the HMAC for the received packet. If the check results in success, it decrypts the received packet with the corresponding key and forwards it according to MPRF. The HMAC calculation takes a few micro seconds as shown in [5]. Only the corresponding local receiver decrypts the packet. If $D(.)$ denotes the overhead for packet decryption, and n is the average number of neighbors in transmission range, Hint reduces the average computation overhead at a node from $\frac{1}{2}n^2D(.)$ to $\frac{1}{2}n^2HMAC(.)$ when compared to schemes that encrypt and broadcast a packet.

Due to the transformation on each link combined with broadcast transmission, eavesdroppers are not able to learn the relationship between incoming and outgoing packets of a node. Although a compromised node en route may see several control fields like TTL in clear text, it cannot discover which node will be the next hop of its neighboring next hop. For each traffic flow, since there is no relation between flows, an adversary will have difficulty in discovering the flow. Furthermore, when a destination receives a packet, it broadcasts a random packet as a response, hiding its role from neighboring nodes. This random packet is not distinguishable from a transformed packet by Hints. Neighboring nodes discard the packet.

During route discovery, Hints are used to transform a RREP in the same way. Otherwise, an adversary may discover a route through tracing RREP messages.

Random Time-To-Live (RTTL): The TTL field is used for discarding packets which have not found a destination and circulated through the network. In MANETs, the TTL is set to the length of a path by a source node. Each node on the path decreases the value by 1. Thus, the TTL value reveals the position of a node on a path from a source or a destination. The receiver anonymity set may be reduced to a set of nodes neighboring a compromised node from a set of all possible receivers.

To prevent compromised nodes from learning their position on a path, we propose a Random Time-To-Live (RTTL). A source node generates a random value and sets the TTL field with the sum of this random value and path length, RTTL. The RTTL should be less than the maximum hop count (Network diameter). The source includes the initial random value in the encrypted data packet. Intermediate nodes decrease the TTL value of a packet by 1 as they do in the normal packet forwarding. This TTL field does not release the absolute position of a node due to the random value. A destination decrypts the received packet and checks if the received RTTL is valid by subtracting its initial random value.

IV. Security Analysis

In Section II-B, we presented a classification of attackers. In this section, we characterize the anonymity provided by PPCS against attacks by internal compromised nodes and then argue informally about the anonymity provided by our system against eavesdropping attacks. To support this analysis, we present an optimal guessing strategy to be used by an adversary for each attack.

A. Internal Attackers

In this subsection we examine the effectiveness of PPCS against collaborating internal adversarial nodes. Intermediate nodes on the path can see the flow pseudonym and TTL field of a packet. Intermediate nodes also know the previous and next hop nodes of a packet on the routing path. Using this information, the compromised intermediate nodes on a path collude to make an educated guess as to the source and destination of a flow.

To characterize the probability that a set of internal compromised nodes collaborate on successfully discovering anonymity we first derive a general equation which can be applied to each case of anonymity (source/destination and communicating pair).

The following notation is used in the remainder of our analysis.

- N: Total number of nodes
- C: Number of compromised nodes in the network
- L: Average path length
- T: Number of uncompromised nodes disclosed by intermediate compromised nodes en route
- W: Number of intermediate nodes on multiple paths established between the source and destination
- G: $(N - C) - T$
- p: probability that a node is compromised
- $P_{f,s}$=$P_{l,r}$: probability that the first/last hop node guesses a source/destination correctly, respectively
- $P_{i,s}$=$P_{i,r}$: probability that an intermediate node guesses a source or a destination correctly
- $P_{i+f,l}$=$P_{i+l,l}$: probability that the first/last hop node and intermediate nodes together guess linkability of the source correctly and destination
- $P_{f+l,l}$: probability that the first and last hop nodes together guess linkability of the source and destination correctly
- $P_{i+i,l}$: probability that intermediate nodes together guess linkability of the source and destination correctly

Let $P(A = s)$ and $P(A = r)$ denote the probability that an adversary discovers a source or a destination. Note that the adversary can determine only which node is a source or a destination, not the identifier due to the random node and flow identification schemes. Since the values, $P(A = s)$ and $P(A = r)$, are the same, we discuss the probability $P(A = s)$ below. Let $P(A = (s, r))$ denote the probability that an adversary discovers the source and destination pair.

1) **Generalization***:* Without loss of generality, we assume that the probability of a compromised node being able to exploit a vulnerability is dependent on its position on a path. In particular, the first and last hop nodes on a path may have a higher probability of finding a source or destination, respectively, than an intermediate node on the path depending on the characteristics of the security solution. To this end we derive the probability of four cases of node compromise as in Table IV-A.1. We determine the probabilities of $P(CH)$, $P(HC)$, $P(CC)$, and $P(HH)$ for a path that has k compromised nodes in each case.

$$\begin{aligned} P(CH) &= (1-p)^{L-k}p^k\tbinom{L-2}{k-1} \\ P(HC) &= (1-p)^{L-k}p^k\tbinom{L-2}{k-1} \\ P(CC) &= (1-p)^{L-k}p^k\tbinom{L-2}{k-2} \\ P(HH) &= (1-p)^{L-k}p^k\tbinom{L-2}{k} \end{aligned}$$

Let $P_{CH}|P_{HC}|P_{CC}|P_{HH}$ denote the probability that an adversary discovers target anonymity in each case.

TABLE I

CLASSIFICATION OF NODE COMPROMISE

CH	the first hop of a source is compromised and zero or more other compromised nodes are on the path, but not the last hop.
HC	the last hop is compromised and zero or more other compromised nodes are on the path, but not the first hop node.
CC	the first and last hop nodes are compromised, as well as zero or more compromised nodes on the path.
HH	the first and last hop nodes are not compromised nodes, but one or more compromised nodes are on the path

$$\begin{aligned} P_{CH} &= P(A|CH)P(CH) \\ P_{HC} &= P(A|HC)P(HC) \\ P_{CC} &= P(A|CC)P(CC) \\ P_{HH} &= P(A|HH)P(HH) \end{aligned}$$

In these equations, $P(A|X)$ is the probability that anonymity is discovered given that the compromise scenario X has occurred.

The probability that an adversary discovers target anonymity is defined

$$P(A) = P_{CH} + P_{CC} + P_{HC} + P_{HH} \quad (1)$$

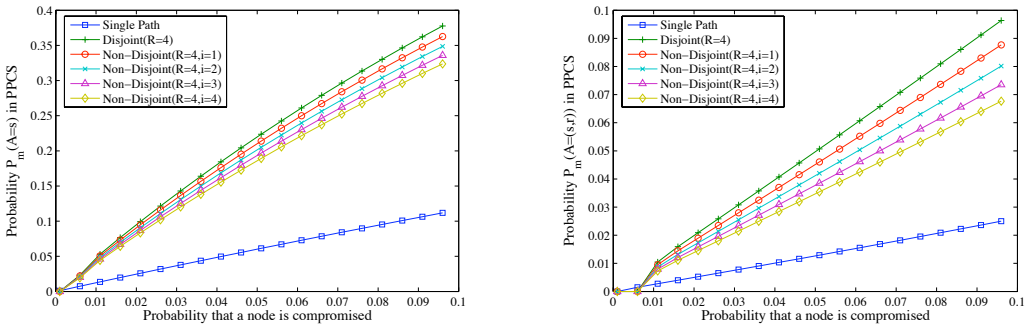
This is a measure of the effectiveness of compromised nodes. In disjoint multi-paths environments, the probability that an adversary discovers anonymity is

$$P_m(A) = 1 - (1 - P(A))^R \quad (2)$$

where R is the number of disjoint paths established between the source and destination.

2) **Optimal Guessing Strategy***:* We now present the optimal strategy that an adversary may use to discover flow endpoints (a source, a destination, or both). First, consider an optimal anonymity solution in which no information is leaked. In this case a compromised node does not know its previous or next hops, or its position on a path. It only knows of other compromised nodes. In this situation, the best an adversary can do is to guess the source from the set of uncompromised nodes. The probability of guessing correctly is $\frac{1}{(N-C)}$.

Now consider a non-ideal anonymity solution in which an adversary can identify its position on the path, but not other nodes on the path except for its direct previous and next hops. If the node is the first hop (information learned by seeing the TTL in the reverse path), it knows its previous hop is the traffic source. If a node is not the first hop on a path, its best guess is a random choice of all nodes in the network not counting the nodes it knows to be compromised or the nodes that compromised nodes can rule out as the source, such as their next hop nodes or previous hop nodes if they are not the first on the path. We call this set U, which has $G = N - C - T$ members. Thus



(a) Source Anonymity **(b) Source and Destination Linkability**

Fig. 1. Probability of an adversary

the probability of an intermediate node guessing correctly is $\frac{1}{G}$.

Finally, consider the situation when RTTL is used within PPCS. In this case an adversary knows it is on the path, but cannot tell its position on the path. Therefore, a different guessing strategy will be used. The adversaries have two choices. First, they can make a random guess of all nodes in set U, in which case their chance of guessing correctly is $\frac{1}{G}$. A better strategy is simply to guess its previous hop as being the source. Although the adversary does not know its place on the path, it has a $\frac{1}{L}$ chance of being the first hop node and thus guessing correctly. Even if several nodes on the path are compromised and collaborate, the only information they can learn is which adversary is closest to the source, and guess the previous hop to that node, i.e., they will all guess the same node. This strategy results in a probability of guessing the source that approaches $\frac{1}{L}$, independent of the number of compromised nodes on the path. The only way that the random guess strategy will be better for an individual node is if $G \leq L$, i.e., the average path length is greater than the number of uncompromised nodes in the network which is an unlikely scenario.

Based on the discussion above, we assume the following three strategies to guess the source node on a path: (1) In an ideal environment, adversaries make a random guess from the set of non-compromised nodes; (2) If an adversary is on a path, and it knows its position on the path, it will guess its previous hop as the source if it is the first hop node, otherwise it will make a random choice from the set U; (3) If an adversary is on a path, and it does not know its position on the path, it will always guess its previous hop on the path as the source.

3) **Source/Destination Anonymity***:* Compromised internal nodes collaborate to determine a source using explicit information such as the flow pseudonym, TTL value, and next and previous hop nodes.

Let us suppose that there is more than one compromised node on a routing path. These nodes conspire to discover a source of traffic. $P_{f,s}$ and $P_{i,s}$ are the probabilities that the first hop and intermediate nodes guess a source, respectively. The probability $P(A = s)$ is

$$\begin{aligned} P(A=s) &= P_{CH} + P_{HC} + P_{CC} + P_{HH} \\ &= P_{f,s} \sum_{k=1}^{L-1} (1-p)^{L-k} p^k \binom{L-2}{k-1} \\ &+ P_{f,s} \sum_{k=2}^{L-2} (1-p)^{L-k} p^k \binom{L-2}{k-2} \\ &+ P_{i,s} \sum_{k=1}^{L-1} (1-p)^{L-k} p^k \binom{L-2}{k-1} \\ &+ P_{i,s} \sum_{k=1}^{L-2} (1-p)^{L-k} p^k \binom{L-2}{k} \end{aligned} \tag{3}$$

The first two terms correspond to the first two terms in equation 1. The last two terms correspond to the last two terms in equation 1. Note that we do not need to account for intermediate nodes compromised in the scenarios covered by the first two terms in equation 1 because of the manner in which compromised nodes will collaborate. That is, if two nodes on a path are compromised and collaborate, they can compare the TTL field of the packets they receive and determine who is closer to the source. This is the only node that can correctly guess the source if an optimal guessing policy is used as discussed.

Based on the optimal guessing strategy discussed in the previous subsection, we can now evaluate $P_{f,s}$ and $P_{i,s}$ and determine the impact of PPCS. If an adversary knows its position on the path, $P_{f,s} = 1$ and $P_{i,s} = \frac{1}{G}$. In cases in which an adversary does not know its position on the path, such as if RTTL is used with PPCS, $P_{f,s} = 1$ and $P_{i,s} = 0$. This is because all adversaries will always guess the previous hop of a first adversary (the same guess), so in cases in which the first hop node is an adversary, all guess will be correct, and in cases in which the first hop node is not an adversary, all guesses will be incorrect.

We now extend this analysis to consider the impact of MPRF on security. PPCS establishes multiple paths

between the source and destination. With the assumption that each path of the R paths is disjoint, the probability an adversary discovers a source or destination is

$$P_m(A = s) = 1 - (1 - P(A = s))^R \tag{4}$$

In a disjoint multi-path environment, intermediate nodes have only one previous and next hop nodes. Since intermediate nodes do not know their position on a path, compromised nodes have the same probability $P_{f,s} = 1$ and $P_{i,s} = 0$ which is used to compute $P_m(A = s)$.

Figure 1 shows the effectiveness of compromised nodes in a disjoint multiple-path environment. An adversary has a higher probability of guessing the source in a multiple disjoint path environment since more information may be open to more compromised nodes.

However, MPRF uses multiple non-disjoint paths. Thus every intermediate node may have multiple forward and backward hops for a flow. Furthermore, the first hop node on one path may be a non-first hop node on a different path of which it is a part. These multiple incoming links increase the number of choices for guessing, and hence reduce the probability of an adversary guessing correctly.

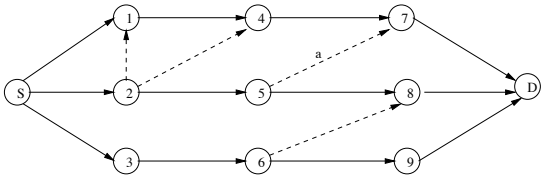


Fig. 2. Non-Disjoint Multi-Paths

In Figure 2, the addition of each dotted link increases the incoming degree of the corresponding nodes(1, 4, 7, and 8). From this, we can compute the average incoming degree of a node, $\frac{W+i}{W}$, where W is the number of nodes on disjoint multipaths and i is the number of added directed links.

TABLE II
IMPACT OF PPCS ON PROBABILITY

Prob.	Perfect Anon.	No PPCS	PPCS (Previous Hop Policy)		
			Single Path	Disjoint Multi-path	Non-Disjoint Multipath
P_{fs}	$\frac{1}{N-C}$	$\frac{1}{G}$	1	1	$\frac{W}{W+i}$
P_{is}	$\frac{1}{N-C}$	$\frac{1}{G}$	0	0	0

i: number of directed links added to disjoint multipaths

Hence, the probability that an intermediate node determines a node from candidate previous hop nodes is $\frac{W}{W+i}$. $P_{f,s}$ becomes $\frac{W}{W+i}$. $P_{i,s}$ is still 0 since intermediate nodes beyond the first hop will always guess wrong. Figure 1 (a) compares the probability that an adversary may guess a source in disjoint multi-path and non-disjoint multipath environments where 4 disjoint multipaths exist and the average path length is 5. This result demonstrates that MPRF in PPCS reduces the effectiveness of an adversary.

In summary, Table II shows the effect of using PPCS on the probability that intermediate and first hop nodes guess a source correctly. For destination anonymity, the analysis and equations are similar.

4) **Source and Destination Unlinkability***:* If the path between a source and destination is known, the source and destination pair is also discovered. The probability that an adversary discovers the source and destination pair in a single path environment is

$$\begin{aligned} P(A{=}(s,r)) &= P(A{=}(s,r)|CH)P(CH) \\ &+ P(A{=}(s,r)|HC)P(HC) \\ &+ P(A{=}(s,r)|CC)P(CC) \\ &+ P(A{=}(s,r)|HH)P(HH) \\ &= P_{i+f,l}\sum_{k=1}^{L-1}(1-p)^{L-k}p^k\tbinom{L-2}{k-1} \\ &+ P_{i+l,l}\sum_{k=1}^{L-1}(1-p)^{L-k}p^k\tbinom{L-2}{k-1} \\ &+ P_{f+l,l}\sum_{k=2}^{L-2}(1-p)^{L-k}p^k\tbinom{L-2}{k-2} \\ &+ P_{i+i,l}\sum_{k=1}^{L-2}(1-p)^{L-k}p^k\tbinom{L-2}{k} \end{aligned} \tag{5}$$

$P_{*,l}$ denotes the probability that nodes en route guess the source and destination pair.

As discussed in the previous section, if an adversary knows its position on a path, the probability that the first/last hop node determines a source or a destination is 1. The probability that other intermediate nodes guess a source/destination becomes $\frac{1}{G}$, since intermediate nodes know that their previous/next hop is not the source/destination and may guess one node of a set of possible sources/destinations. Therefore, if intermediate nodes know their position, $P_{f+i,l}$ and $P_{i+l,l}$ are $\frac{1}{G}$, $P_{i+i,l}$ is $(\frac{1}{G})^2$, and $P_{f+l,l}$ is 1.

If the adversary does not know its position on a path because of RTTL, the same guessing strategy as previously discussed is used. Thus, $P_{f+l,l}$ is 1, and $P_{i+f,l}|P_{i+l,l}|P_{i+i,l}$ become 0.

By extending the above single path case to a disjoint multi-path, the probability of discovering the source and destination pair is

$$P_m(A = (s,r)) = 1 - (1 - P(A = (s,r)))^R \tag{6}$$

In disjoint multi-path environments, intermediate nodes have the same probability as the single path to guess the source and destination pair. Figure 1 (b) shows the probability that an adversary discovers the communicating pair in a disjoint multi-path environment.

In a non-disjoint multi-path environment, we can apply the same reasoning as for the source anonymity case to determine that $P_{f+l,l}$ is $(\frac{W}{W+i})^2$, and $P_{i+f,l}|P_{i+l,l}|P_{i+i,l}$ become 0.

As Figure 1 (b) shows, an adversary has a lower probability to discover the communicating pair in non-disjoint multi-path environments than disjoint multi-path environments. This verifies that MPRF of PPCS mitigates the effectiveness of internal compromised nodes, while providing defense against eavesdropping attacks.

B. Eavesdropping

Since nodes in MANETs share a common broadcast channel, they overhear all communication within transmission range. Hence, an adversary may learn information by collecting and analyzing overheard data without revealing its existence. A set of local eavesdroppers form a global eavesdropper to cover a path. They may have a dedicated communication channel to exchange information.

In PPCS, every node en route uses the *Hint* to prevent correlation between forwarded packets and locally broadcasts the transformed packet. The eavesdroppers may not learn which node is the local sender and receiver of a packet, due to the local broadcasting and hop-by-hop transformation of packets. This limits eavesdroppers from obtaining information about the relationship between the incoming and outgoing packet of a node.

MPRF in PPCS spreads traffic over multiple paths, preventing eavesdroppers from learning the source, destination, or communicating pair by counting broadcast packets. Eavesdroppers located in different areas see different amounts of broadcast traffic with varying delay. Thus, a global eavesdropper is unable to discover significant information about node identity or flows.

To fully characterize eavesdropping requires a model of traffic that encompasses the amount of information an adjacent eavesdropping node can observe, and distribution of information sent through that victim and intermediate nodes, and the frequency and structure of the underlying traffic. We are currently developing a analytical model for this exceedingly complex environment.

V. Performance Evaluation

In this section, we evaluate the effect of PPCS on the performance of routing and data transmission. We performed our simulation in the ns2 simulator [9]. Specifically, we evaluate the effect of MPRF in which multiple paths are established and each packet on a flow may take a different path.

As a baseline multi-path routing protocol we use ad hoc on-demand multipath distance vector routing (AOMDV) [17]. To implement MPRF, we modified AOMDV to relax the node/link disjointness requirement and to randomly choose a next hop node at each intermediate node. Finally, to determine the impact of randomly changing the node pseudonym during the life of a flow, we modified MPRF to create a version that uses stable node pseudonym, called S-MPRF. Table V summarizes the simulation environment.

TABLE III
Simulation Parameters

Simulation Time	900 seconds
Number of nodes	50
Area	900X900
Speed	Maximum 20 m/sec
Mobility model	Random Waypoint Model
Packet size	512 bytes
Traffic pattern	10 CBR/UDP connections (4 packets/s)

We measured packet delivery ratio (PDR), end-to-end packet delay, and routing overhead with different pause times under a random waypoint mobility model.

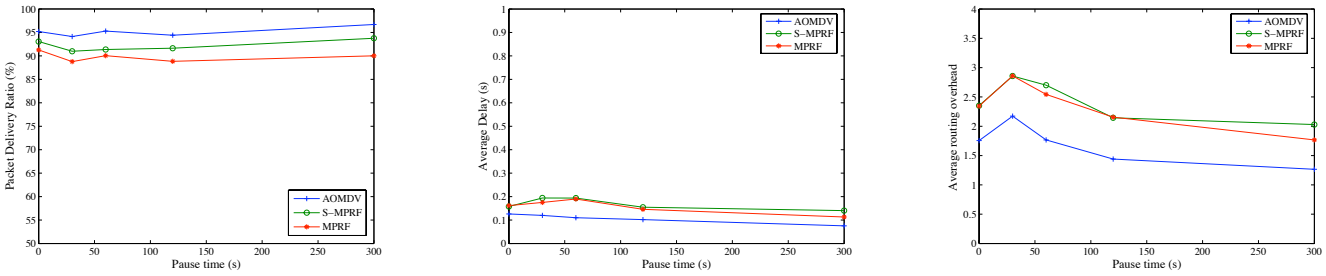
MPRF increasingly degrades the packet delivery ratio as mobility increases. Since each packet takes a different path, packets are more vulnerable to link failure or network congestion. Figure 3 (a) shows that the packet delivery ratio is decreased 3% and 5% in S-MPRF and MPRF, respectively. This result shows that the impact of changing node pseudonyms is small. The fact that multiple paths are susceptible to breaking for each flow, increases the routing overhead required to overcome these failures. As shown in Figure 3 (c), there is a 42% increase in routing overhead in MPRF over AOMDV.

In traditional routing protocols, packets are transmitted on the shortest path. With MPRF packets are randomly distributed to across multiple paths. Because some paths will be longer than the shortest path, the end-to-end packet delay will increase. Figure 3 (b) shows a 51% increase in packet delivery delay in MPRF and S-MPRF. We discuss the trade-offs between the security and performance in the next section.

VI. Discussion

In this section we discuss the trade-offs of MPRF. According to the analysis in Section IV-A, as the number of paths increases, the probability of an internal adversary compromising anonymity increases. While using non-disjoint paths is better than using disjoint paths, both are less secure against internal attackers.

Although a single path solution is more secure against internal compromised nodes, it is less secure against eavesdroppers. To combat these attacks, it is better to establish more paths to distribute traffic. As an extreme example, if a packet is broadcast over the entire network (the number of multiple paths is infinite), eavesdroppers may not discover a flow at all.



(a) Packet Delivery Ratio **(b) Packet Delay** **(c) Routing Protocol Overhead**

Fig. 3. Performance with different pause times

Based on a security perspective alone, the choice of using MPRF should be based on a risk analysis of the network. If an attacker is more likely to be external, MPRF should be used. If the attacker is more likely to be internal, it should not.

If MPRF is to be used, the packet forwarding performance of the network will decrease as discussed in V. Disjoint multi-path forwarding provides better packet delivery ratio (3-5%) than the non-disjoint multi-path forwarding used in MPRF. In non-disjoint multi-path environments, an intermediate node may receive packets of a flow from multiple neighbors which may cause more collisions on the wireless interface. However, given that the difference in performance is small, using MPRF is advisable as it does improve security as shown in Figure 1.

VII. Related Work

A great deal of previous research has focused on providing confidentiality, integrity, and authenticity of data in MANETs, but anonymity remains an open problem. Pfitzman and Hansen [20] define general terminologies of anonymity. In their article, anonymity is defined as "state of being not identifiable within a set of subjects, the anonymity set."

Chaum's [4] pioneering anonymity solution introduces a mix or a series of mixes (mix network) into a network for hiding communicating endpoints [10] in the Internet. A source selects the route (set of mixes) and encrypts data packets with the public key of each mix in reverse order (from last mix to the first mix). Each mix peels off one layer by decrypting the received packet with its private key and forwarding it to the next hop. The last mix processes the packet in the same way and transmits it to the destination.

Onion routing [22] is built on a mix-net approach. An onion consists of next hop information and an onion for the next hop. Each intermediate onion router decrypts the received message with its private key to get the next hop and onion for the next hop. The last onion peels off its layer and transmits the encrypted data to the destination. Tor [6] extended onion routing with features that provide forward secrecy.

Mix-nets are not applicable to MANETs, because the resource demands of the underlying public key operations are too expensive for mobile nodes with energy and computation limitations. Moreover, with high mobility, it is not easy to maintain the full path from the source.

In Crowds [23], groups of users (called *crowds*) cooperate to ensure client anonymity in web systems, e.g., web-browsing. *Jundos* run by each client decide randomly if they should relay the packet to another jundo or transmit it to the web server directly. All users in the group share their symmetric keys to encrypt the relayed packet. Hordes [18] is based on Crowds and proposes to use multicast routing to provide initiator anonymity. Brent [25] proposes receiver anonymity based on incomparable public keys and multicast. In MANETs, however, the maintenance cost of multicast is known to be high.

Most solutions proposed for the Internet use a proxy function (Mix, Jundo, and Onion Router) to provide anonymity. In MANETs, Jian et al. [11] propose a dynamic mix method that accommodates dynamic topology changes. Blaze et al. propose WAR [2], in which anonymous routing is combined with a key distribution protocol and an onion routing structure. However, in MANETs, it is not feasible to form a set of proxy functions since mobile nodes all play an equal role. In civilian applications of MANETs, in particular, mobile nodes may not cooperate to play the larger role of a proxy.

J. Kong and X. Hong [12] apply MIX-Net to MANETs by using symmetric key cryptography to provide anonymity. This approach uses a cryptographic trapdoor within a broadcast message to hide the identifiers of local intermediate nodes and the destination. However, in a situation in which adversaries are located on each link, they may simply monitor the transmission to determine who is broadcasting and how many packets are being broadcast.

Recently, Zhang et al. proposed MASK [27] in which a Trusted Authority (TA) assigns a large number of random identifiers and a set of corresponding secret points to each node sufficient for the lifetime of a node.

VIII. Conclusion

In this paper we presented PPCS, a comprehensive system for providing anonymity in a MANET. The solution is efficient, so it is appropriate for a MANET environment. The solution is comprised of several components. The use of node and flow pseudonyms (dynamic and random) provides a level of node anonymity and unlinkability between a source and destination. The use of multipath random forwarding combined with transforming packets on each link and using broadcast mechanisms to forward packets raises the level of difficulty in performing traffic analysis attacks. Obscuring the hop counts provided by many MANET protocols in the form of a TTL field reduces the ability of an adversary to determine its position on a path and use this information to derive a source or destination.

We provided a detailed security analysis of PPCS for passive internal attackers. The analysis showed that PPCS is effective at reducing the effectiveness of adversaries. We also provided a discussion of the trade-offs between performance and the security solution.

ACKNOWLEDGMENTS

This work was supported by NSF Grant NSF CNS-0519460. Research was sponsored in part by the U.S. Army Research Laboratory and the U.K. Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation heron.

References

[1] A. Back, U. Moller, and A. Stiglic. Traffic Analysis Attacks and Trade-Offs in Anonymity Providing Systems. *Proceedings of Information Hiding Workshop (IH 2001)*, 2001.

[2] M. Blaze, J. Ioannidis, and A. D. Keromytis. WAR: Wireless Anonymous Routing. *Security Protocols Workshop*, 2003.

[3] H. Chan and a. S. A. Perrig. Random key predistribution schemes for sensor networks. *IEEE Symposium on Security and Privacy*, 2003.

[4] D. L. Chaum. Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. *Communications of the ACM*, 1981.

[5] H. Choi, W. Enck, P. Mcdaniel, and T. F. L. Porta. Secure Reporting of Traffic Forwarding Activity in Mobile Ad Hoc Networks. *Proceedings of The Second Annual International Conference on Mobile and Ubiquitous Systems*, 2005.

[6] R. Dingledine, N. Mathewson, and P. Mathewson. Tor: The Second-Generation Onion Router. *Proceedings of the 13th USENIX Security Symposium*, 2004.

[7] W. Du, J. Deng, S. Han, and P. Varshney. A pairwise key predistributionscheme for wireless sensor networks. *ACM Conference on Computer and Communications Security*, 2003.

[8] L. Eschenauer and V. Gligor. A key management scheme for distributed sensor networks. *ACM Conference on Computer and Communications Security*, 2002.

[9] http://www.isi.edu. The Network Simulator - ns-2, 2000.

[10] A. Jerichow, J. Muller, A. Pfitzmann, B. Pfitzmann, and M. Waidner. Real-Time Mixes: A Bandwidth-Efficient Anonymity Protocol. *IEEE Journal on Selected Areas in Communications*, 1998.

[11] S. Jiang, N. Vaidya, and W. Zhao. A Mix Route Algorithm For Mix-net in Wireless Mobile Ad Hoc c Network. *IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, 2004.

[12] J. Kong and X. Hong. ANODR:ANonymous On Demand Routing with Untraceable Routes for Mobile Ad-hoc Networks. *In ACM MOBIHOC*, 2003.

[13] H. Krawczyk, M. Bellare, and R. Canetti. HMAC: Keyed-Hashing for Message Authentication. *IETF RFC 2104 (http://www.ietf.org/rfc/rfc2104.txt)*, 1997.

[14] S.-J. Lee and M. Gerla. Split Multipath Routing with Maximally Disjoint Paths in Ad Hoc Networks. *IEEE International Conference on Communications*, 2001.

[15] B. N. Levine, M. K. R. C. Wang, and M. Wright. On timing attacks in low-latency mix-based systems. *In Proceedings of the 8th International Conference on Financial Cryptography*, 2004.

[16] D. Liu and P. Neng. Establishing pairwise keys in distributed sensor networks. *ACM Conference on Computer and Communications Security*, 2003.

[17] M. K. Marina and S. R. Das. AOMDV: Ad hoc On-demand Multipath Distance Vector Routing Protocol. *IEEE ICNP*, 2001.

[18] B. Neil and C. Shields. Hordes: A Protocol for Anonymous Communication Over the Internet. *ACM Journal of Computer Security*, 2002.

[19] C. Perkins and E. Royer. Ad hoc On-Demand Distance Vector (AODV) Routing. *IETF RFC 3561 (http://www.ietf.org/rfc/rfc3561.txt)*, 1999.

[20] A. Pfitzmann and M. Hansen. Anonymity, unlinkability, unobservability, pseudonymity, and ident ity management -a consolidated proposal for terminology version v0.23. dud.inf.tu-dresden.de/literatur/.

[21] J.-F. Raymond. Traffic Analysis: Protocols, Attacks, Design Issues and Open Problems. *Proceedings of International Workshop on Design Issues in Anonymity and Unobservability*, 2000.

[22] M. G. Reed, P. F. Syverson, and D. M. Goldschlag. Anonymous Connections and Onion Routing. *Journal on Selected Areas in Communication Special Issue on Copyright and Privacy Protection*, 1998.

[23] M. K. Reiter and A. D. Rubin. Crowds: Anonymity for Web Transactions. *ACM Transactions on Information and System Security*, 1(1):66–92, 1998.

[24] A. Serjantov and P. Sewell. Passive attack analysis for connection-based anonymity systems. *In European Symposium on Research in Computer Security*, 2003.

[25] B. R. Waters, E. W. Felten, and A. Sahai. Receiver anonymity via incomparable public keys. *ACM conference on Computer and communications security*, 2003.

[26] Z. Ye, S. V. Krishnamurthy, and S. K. Tripathi. A Framework for Reliable Routing in Mobile Ad Hoc Networks. *IEEE INFOCOM*, 2003.

[27] Y. Zhang, W. Liu, and W. Lou. Anonymous Communications in Mobile Ad Hoc Networks. *IEEE INFOCOM*, 2005.

[28] S. Zhu, S. Setia, and S. Jajodia. LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks. *ACM conference on Computer and communications security*, 2003.