SAS: A Simple Anonymity Scheme for Clustered Wireless Sensor Networks

Satyajayant Misra and Guoliang Xue

Abstract— **In this paper, we propose a simple and efficient scheme for establishing anonymity in clustered wireless sensor networks. This scheme is applied to a clustered sensor network in which the nodes in a neighborhood share pairwise keys for authentic and confidential communication. The scheme, named Simple Anonymity Scheme (SAS), uses a range of pseudonyms as identifiers for a node in the network, to ensure concealment of its true identifier (ID). After deployment, neighboring nodes in the network share their individual pseudonyms and use them to ensure that the communication is anonymous and that a node's true ID is kept private. Even when many nodes in a given neighborhood of the network are compromised and are colluding, our scheme ensures that non-compromised nodes are still guaranteed complete anonymity. The compromised nodes cannot identify the sender or the receiver of communication happening between non-compromised nodes. Our scheme requires reasonably low memory and has very low computation cost, needing no change in other protocols of the network stack. It can be embedded into any wireless sensor network routing protocol to ensure anonymity and privacy during node discovery and routing in the network.**

Index Terms: Wireless sensor networks, clustering, authentication, confidentiality, anonymity, privacy, pseudonym.

I. Introduction

Large scale distributed wireless sensor networks (*WSNs*) are becoming increasingly common in a variety of applications [1]. Despite significant improvements in the robustness of the sensor nodes (*SNs*), they are still hugely constrained, having limited power, memory and computing abilities [8]. The available redundancy and inherent energy scarcity of a sensor network encourages the use of aggregation of data while on its way to the base station from the point of stimuli.

Clustering of the *WSNs* vastly improves this aggregation ability. In a clustered wireless sensor network (*CWSN*), the nodes in a neighborhood organize themselves into a cluster with one node designated as the cluster head (*CH*) [2], [15]. The *CH* gets information regarding a stimulus from the *SNs* in its neighborhood and uses an aggregation scheme to aggregate this information, sending the information to a neighboring *CH* in the direction of the base station (*BS*). This neighboring *CH* may aggregate the information further and send it ahead.

In many applications of the *WSN*, identity of the nodes sending data from a locality to the *BS* might be extremely sensitive information. In a mobile wireless ad hoc network, identity of a node is generally given by a tuple {location, identifier, time} [5]. In a static *WSN*, in general, the identifier (ID) is sufficient for unique node identification. An intelligent adversary analyzing the traffic in the network may obtain access to this identity information. Such an informed adversary can infer and destroy/compromise the identified *SNs*, rendering the network ineffective. The problem of traffic analysis becomes even more critical in a *CWSN*. Identity information of the *CH* in a region can allow an adversary to compromise the *CH*, effectively compromising the complete cluster and ensuring that the *BS* gets no information from the cluster's locality. To solve the problem of traffic analysis in the *CWSN*, design and deployment of effective anonymity solutions is essential. Anonymity solutions allow the *SNs* to use dynamic pseudonyms during communication, thus reducing the scope of traffic analysis significantly.

In this paper, we propose an anonymity solution for a *CWSN*, where the *SNs* in a neighborhood share pairwise symmetric keys, generated using exchange of pre-deployed information [3]. We assume that clustering is done in the network using the clustering scheme proposed in [15]. We also assume that the Tiny OS beaconing scheme [8] is used for creating inter-cluster routes for communication of a *CH* with the *BS*. We propose a simple, yet effective scheme that provides the *SNs* with dynamic pseudonyms to use as their identity during communication, ensuring complete anonymity. The *SNs* are given pseudonym ranges that are non-contiguous and chosen uniformly at random from a pseudonym space. Our scheme guarantees complete anonymity to a communicating *SN* even when several of its neighbors are compromised and are colluding.

The rest of this paper is organized as follows. In Section II, we briefly survey related work in the areas of clustering, anonymity and privacy in wireless ad hoc networks. In Section III, we give the problem statement and define the models, security assumptions and requirements for anonymity in a *CWSN*. In Section IV, we describe the framework for our proposed anonymity scheme. In Section V, we propose the SAS and analyze the protocol. In Section VI, we give our conclusions and scope of future work.

II. Related Work

Ibriq and Mahgoub [7] specified the design criteria and challenges for cluster based wireless sensor networks (*WSNs*). In [4], the authors proposed an on-demand distributed clustering algorithm for ad hoc networks. In [15], a hybrid,

This research was supported in part by ARO grant W911NF-04-1-0385 and NSF grants CNS-0524736 and CCF-0431167. The information reported here does not reflect the position or the policy of the federal government.

Both authors are with the Department of Computer Science and Engineering, Arizona State University, Tempe, AZ 85287-8809. Email: {satyajayant, xue}@asu.edu.

energy efficient and distributed clustering protocol (HEED), was proposed, which does not depend on the network topology or size, nor makes any assumptions on the node degree. In [2], the authors proposed a distributed and randomized clustering algorithm that generates a hierarchy of *CHs*.

Security in *WSNs* has been a topic of intensive study in the last few years. In [8], the authors considered routing security in *WSNs*, identifying attacks and proposing countermeasures. Zhu *et al.* [17] proposed a key management protocol for *WSNs*, that supports in-network processing. Liu and Ning [10] presented a general framework for establishing pairwise keys between sensors, on the basis of a polynomial-based key pre-distribution protocol. Du *et al.* [6] proposed a novel secret key pre-distribution scheme to improve resilience of the network.

Anonymity and security in *CWSNs* has not been studied in great details, although there has been significant work in ad hoc networks that may be applied to these sensor networks. In [12], the authors used both public and symmetric key cryptography to provide security in the cluster based routing protocol for ad hoc networks. In [5], the authors presented a novel architecture that provides location anonymity to a mobile node by splitting the identification information among the entities in the network. Zhu *et al.* [16] proposed the ASR protocol that provides a form of identity anonymity and location privacy. In [13], the authors proposed an on-demand position based private routing protocol for an ad hoc network. Kong *et al.* [9] proposed an anonymous on-demand routing protocol, for mobile ad hoc networks deployed in hostile environments. To our best knowledge, no research on anonymity in wireless networks has addressed the problems we are studying in this paper.

III. Problem Statement

A. System Model

We consider a wireless sensor network composed of a large number of similar, small, low cost and immobile sensors. These sensors are assumed to have unique IDs. They have limited power, memory and computation abilities, and are not tamper resistant. The network is partitioned into clusters. The links in the network are assumed to be bidirectional. The *SNs* send sensed data to the elected *CH*. The role of a *CH* rotates between the *SNs* in a neighborhood. We assume that the neighborhood of a *SN* consists of all other *SNs* within its transmission range and includes itself.

There are many clustering algorithms that have been proposed in the literature such as, [2], [4], [15]. We assume that the network uses some clustering mechanism. We propose a scheme for establishing anonymity in a given *CWSN*. Our anonymity scheme can work on top of any clustering scheme. The *CH* aggregates data in its cluster and sends it to the *BS* using a multi-hop path created using intermediate *CHs*. The intermediate *CHs* may also aggregate data from their neighboring *CHs* before sending it towards the *BS*. The *BS* acts as the interface for the sensor network to the Internet or a wired network. It is assumed to have unlimited power source and computation ability that is orders of magnitude higher than the sensor nodes themselves. We assume that the *BS* is secure and is not compromised by any malicious user.

We assume that the clustering algorithm, HEED [15] is used for cluster formation, with an added assumption that the *SNs* are static after deployment. These *SNs* have the ability to transmit at several discrete power levels. The highest power level is used for inter-cluster communication. The lower power levels are used for intra-cluster communication and are called the *cluster power levels* [15]. Clustering and *CH* election are done on the basis of residual energy, the *average minimum reachability power* as proposed in [15] is used to break ties. Each node declares itself a *CH* with a probability that is dependent on its residual energy. The process of *CH* selection goes through many iterations. At the end of the iterations a node that is neither a *CH* nor part of any cluster, declares itself as a *CH*. To ensure the inter-cluster connectivity, we assume that the routing mechanism used in Tiny OS, namely, *base station beaconing* [8] is used. The base station beaconing protocol constructs a breadth first spanning tree of the network, rooted at the *BS*. The *BS* broadcasts a route update beacon periodically. All nodes receiving the *BS*'s beacon, designate it as their parent and forward the beacon using their ID as the sender ID. A node receiving this beacon designates its parent as the node whose ID is in the sender field. In our scheme only the *CHs* forward the *BS* beacon message, thus creating a connected network of *CHs* from the *BS* to the periphery of the network. Given the higher order transmit power used for inter-cluster communication, we can assume as in [14], [15] that complete network connectivity is guaranteed.

B. Security Assumptions

The *BS* acts as the key server and also shares a key with every *SN* in the network, for authentic and confidential communication. The *SNs* are identical to the current generation TelosB motes [11] in their computation, communication and power resources. We assume that the *SNs* have adequate memory for storing up to hundreds of bytes of keying material to be used by the anonymity scheme. During initial setup and neighborhood discovery, the *SNs* in a neighborhood exchange their identity information for key setup. For the complete mechanism of such a setup we refer the readers to [6]. After key setup, each *SN* can communicate securely with every other *SN* in its neighborhood and authenticate messages using the shared pairwise keys. In [17], the authors assumed that there exists a lower bound on the time interval (T_{min}) that is necessary for an adversary to compromise a *SN*. The initial setup for our anonymity scheme, which involves exchange of a few encrypted range messages between neighbors, is possible in time much less than T_{min}.

C. Characteristics of Clustered Wireless Sensor Networks

In this paper, we propose an anonymity and privacy solution for a *CWSN* that is characterized by the following attributes:

1) All nodes in the *CWSN* are loosely time synchronized.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

2) All *SNs* outside the range of transmission of a given *SN* cannot comprehend its transmission and treat it as noise.
3) The *SNs* have enough compute power to generate pseudo-random numbers.
4) *SNs* do not communicate among themselves, other than during the initial setup or the *CH* election phase. All other communications happen between the *CH* and the *SNs* or between the *CHs*, and are of broadcast or unicast nature.
5) Similar to [9] we assume that the *SNs* have the ability to obfuscate address fields in their MAC header. This ensures that a *SN*'s MAC header does not give out its identity to a compromised *SN* in the neighborhood.

D. Adversary and Threat Model

Generally two types of attackers are considered for a wireless sensor network: a *sensor* class attacker (*inside attacker*) and the high power *laptop* class attacker (*outside attacker*). The inside attacker might consist of more than one compromised *SNs*, that can mount a concerted attack on the network. On the other hand, a laptop class attacker has higher capacity than the *SNs* in the network and can jam or eavesdrop on the entire network, or create wormholes or sinkholes [8]. In this paper, we assume an adversary that is much stronger than the sensor nodes in the network. The adversary is capable of both insider and outsider attacks, but has bounded computing and traffic analyzing abilities.

Communication of a *SN* with the *BS* and pairwise one-hop neighbors is confidential and authenticated. The adversary should not be able to decrypt any communication until it compromises the nodes in the network. However, it can identify the centers of stimuli in the network by looking at the source and destination IDs in the packet headers. As a direct consequence, it can identify the clusters sending important information and infer the IDs of the *CHs* in the clusters. Knowing the ID of the *CH* the adversary may be able to infer its location and compromise/destroy it, in turn rendering all communications in the cluster compromised.

A compromised *CH* closer to the *BS* shall also allow the adversary to monitor any communication happening through it. Furthermore, an adversary can obtain routing information from the compromised nodes or by eavesdropping. By analyzing this information it can obtain knowledge of the network's topology, thus becoming equipped to disrupt the network.

E. Requirements for Anonymity in a Clustered Wireless Sensor Network

Based on the above subsections we can define the anonymity requirements of a *CWSN* to consist of the following:

1) Every *SN* can communicate with any other *SN* in its neighborhood and the *BS* securely and with anonymity.
2) Routing of messages is anonymous. The *CHs* that are in the forwarding path of a *CH* to the *BS* cannot infer its true ID.
3) The nodes in a cluster should be indistinguishable. A malicious agent, not a part of the *CWSN* should not be able to identify the nodes involved in communication.
4) *SNs* outside the neighborhood of a cluster cannot figure out the *CH* of the cluster. This entails that when the *CH* communicates in its cluster, any other node not in the cluster cannot identify that it is the *CH*. Furthermore, when a *CH* communicates with a neighboring *CH*, no other *SN* can identify it.
5) In essence, any anonymity solution in a *CWSN* environment should provide three kinds of privacy as specified by Zhu *et al.* [16], namely, *identity privacy*, *location privacy*, and *route privacy*.

We note that the final destination of all inter-cluster packets is the *BS*, inferable from the final destination field in the packet. However, this does not aid in traffic analysis for identifying the packet source, as the destination of all packets is the *BS*.

In a key exchange scheme like [3], a neighboring *SN* needs to identify the sender of a packet to be able to use the correct pairwise key to decrypt the packet's contents. The *identity privacy* requirement has to be enforced in a way that the receiver recognizes the sender, to be able to select the correct key for decryption. However, it should not be able to determine the sender's true identity. We address this important aspect in our solution. We do not consider the requirement of *location privacy*, as the clustering and routing schemes we use do not exchange any location information. We address the issue of *route anonymity* but do not propose any new routing algorithm. Our scheme can be used with any existing routing algorithm, to ensure that node discovery, route requests and route replies use pseudonyms, keeping true identity of a node private.

TABLE I
NOTATIONS TABLE

Notation	Explanation
N	number of nodes in the network
m	number of neighbors of a node in the network
K	number of bits used for the pseudonym space
u, v	nodes that we shall use in our illustrations
$2^{\lvert\ell\rvert}$	pseudonym sub-range each *SN* assigns to each neighbor
$\lvert\rvert$	the concatenation operator

IV. FRAMEWORK FOR THE ANONYMITY SCHEME

In this section we present the basic framework we use to build our anonymity scheme. Table I gives a list of notations used and their meaning. Defined below are a few terms that we are going to use in the rest of the paper.

Definition 1: A node u in the network is said to have *complete anonymity*, if no node v that captures packets sent by u has any way to identify that the sender is u, in spite of having knowledge of the sender IDs in the packets.

Definition 2: The neighborhood set $\mathcal{N}_i$ of a node i is the set containing all its neighbors including itself. The *common neighborhood set*, $\mathcal{S}_\mathcal{C}$, of a group of colluding compromised nodes $\mathcal{C}$, is defined as the intersection of the neighborhood

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

sets of the compromised nodes, excluding the compromised nodes themselves. If the set of colluding nodes is defined as, $\mathcal{C} = \{1,\ 2,\ \ldots,\ C\}$, then, $\mathcal{S}_\mathcal{C} = \{\mathcal{N}_1 \cap \mathcal{N}_2 \cap \ldots \cap \mathcal{N}_C\} \setminus \{\mathcal{C}\}$.

The scheme we propose ensures that a node in the network has complete anonymity during communication with uncompromised nodes, even when colluding compromised nodes exist in its neighborhood. Our interpretation for anonymity is that if a *SN*'s true ID is not known to other nodes in the network, they cannot infer it, hence it shall be anonymous. So, a *SN* should use a pseudonym to identify itself. However, use of a static pseudonym is as bad as using true ID, as the *SN* is still vulnerable to traffic analysis by the adversary. The idea of using the shared symmetric keys to encrypt the true ID and using the encrypted ID as a pseudonym has the same drawback as well. Also, if the ID is encrypted, the receiver does not know which node has sent the message. In the worst case, to identify the sender, a receiver might have to decrypt the encrypted pseudonym, with the symmetric keys it shares with each *SN* in its neighborhood, which is a lot of computations. A better solution is for the *SNs* to use a range of unencrypted and indistinguishable pseudonyms while communicating with other *SNs* in the network. However, we still have to ensure that the receiver is able to identify the sender, to select the proper key to decrypt the mutual communication. To identify the sender, a receiver may store a mapping of the relevant pseudonym ranges of the sender with the mutually shared key. We shall describe this in more detail in the following sections. To ensure identification, at the time of setup and neighborhood discovery, the *SNs* in a neighborhood exchange the information to recognize each others pseudonyms. Once this information is exchanged, the *SNs* delete the information needed to decipher each others true identity. We assume as specified in section III-B, that, in this initial period of exchange of the pseudonym ranges the *SNs* are not compromised [17], [18], and follow this procedure correctly.

For anonymity, we use a 'K' bit pseudonym scheme for all the nodes in the network. Hence, the pseudonym space range is, 0 - $2^K - 1$, a total of 2^K pseudonyms. We refer to it as the *pseudonym space*. Further, we assume that a sensor node u uses, L = $2^{|\ell|}$ pseudonyms for communicating with each neighbor (excluding itself). The pseudonym sub-range for each neighbor is contiguous. Each packet sent by u to neighbor v has a pseudonym chosen randomly from the corresponding sub-range. This usage of dynamic pseudonyms by u for sending messages renders traffic analysis ineffective.

V. Simple Anonymity Scheme: SAS

Using the background specified in the previous section we shall propose here the simple anonymity scheme. SAS is a simple scheme for ensuring ID anonymity and privacy for a *SN* even when a significant number of its neighbors are compromised and are colluding. We describe this scheme in two broad stages in the next two sub-sections.

A. Pre-deployment Stage

Before deployment, the pre-deployment authority:

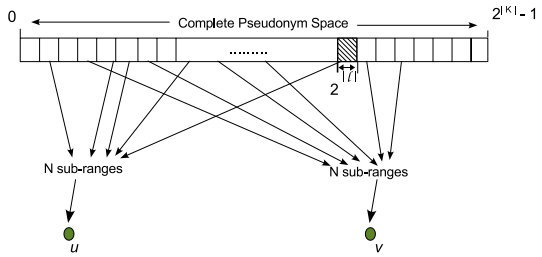

Fig. 1. Pseudonyms ID Space Assignment

1) Divides the IDs from the pseudonym space, uniformly into sub-ranges of size $2^{|\ell|}$ each. The value of ℓ is chosen in such a way that the pseudonym space can be divided into at least N^2 sub-ranges.
2) Assigns each *SN* u, N randomly chosen sub-ranges, distributed uniformly in the pseudonym space. Figure 1 illustrates such an assignment. Hence, node u shall have N ranges of size $2^{|\ell|}$ to use in place of its true ID.
3) Creates a table at the *BS* that stores the pseudonym ranges of each node u. This ensure that when the *BS* receives packets from u, it is able to figure out the correct key to decrypt and authenticate the message.

B. Post-deployment Stage

After the *SNs* are deployed:

1) Each *SN* u randomly chooses one sub-range from its N sub-ranges to ensure anonymity while forwarding the *BS* beacons as the *CH*.
2) The beacon sub-range is also used by *CH*, u, when broadcasting messages in its cluster. *SNs* outside the boundary of the cluster cannot identify these messages as they are sent using only the cluster power level.
3) Each neighbor v of u is also assigned a pseudonym sub-range, chosen uniformly at random from the remaining $N - 1$ sub-ranges.
4) Each *SN* u has a pseudonyms table it uses to store the sub-ranges for communication with other *SNs* in its neighborhood. The table maps the pseudonym sub-ranges that u uses to communicate with a neighbor v and the sub-ranges v uses for u, to the corresponding pairwise key shared between them.
5) To each neighbor v, u securely communicates the beaconing sub-range and the pseudonym sub-range that u has assigned for mutual communication with v. To prevent cases of simultaneous communication, we assume that the *SN* with higher true ID starts the communication. Here, we assume $u > v$.
6) *SN* u also sends the index in its pseudonym table where it shall store the range information for v, in the same message. We shall explain the need for the index information later.
7) When node v receives the range message from u it selects a random sub-range from its pseudonym sub-ranges for communication with u. It then stores this information

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

Index	u's range	Neighbor's Range	Neighbor's Beacon Range	Neighbor's Index	Shared Key
$Index_u$ . . .	ID_{uv1} - ID_{uv2}	ID_{vu1}-ID_{vu2}	ID_{bv1}-ID_{bv2}	$Index_v$	K_{uv}

Fig. 2. Pseudonym Table for node u

about the sub-ranges for mutual/beacon communication with u and u's index along with the mutual key in its pseudonym table. Then v sends its beacon sub-range and the sub-range for mutual communication with u along with the index of the information in its pseudonym table to u. Further, v *deletes* the true ID of u. Now v can only identify u by the pseudonyms u shares with it.

8) When u receives the message from v, it stores the sub-range and index information in the appropriate position in its pseudonym table and *deletes* the true ID of v. Figure 2 shows the pseudonym table of u with the entry for *SN* v, containing the stored sub-ranges, index of v and the mutual secret key.
9) When v wants to communicate with u, it chooses an ID, (ID_{vu}), randomly from the pseudonym sub-range, $ID_{vu1} - ID_{vu2}$, it shares with u and another random ID, (ID_{uv}), from the sub-range, $ID_{uv1} - ID_{uv2}$, u shares with it, for mutual communication. The sender ID and receiver ID are generated as follows: **Sender ID** = $Index_u||ID_{vu}$, where $Index_u$ = Index where u stores information about v, and **Receiver ID** = ID_{uv}.
10) When node u receives the message, it checks the sender ID and uses the index ($Index_u$), to index into its pseudonym table and compare the sender ID with the sub-range of v it has stored in the table. If the pseudonym is in the sub-range, it identifies that the packet is from the correct source. Note that our reference to nodes u and v is simply for illustration. Node identification is based solely on the pseudonym ranges.
11) When *CH*, u, wants to communicate with the *BS* through a neighbor *CH*, v, it uses the same sender ID for both node v and *BS*. The *BS* has information about the ranges of all nodes. It disregards the index information and uses the pseudonym itself to identify the source using a stored reference table.
12) A *CH* forwards the base station beacon using a pseudonym from its beacon sub-range as sender ID. Nodes in its neighborhood check their table for pseudonym match and identify the *CH*. They mark the *CH*'s index in their pseudonym table for future communication with the *CH*. When a *CH* communicates in its cluster neighborhood it uses a special sentinel character as index. When a cluster *SN* gets this message it uses the index it has stored as the *CH*'s index to identify the pseudonym.

Use of index will not aid traffic analysis as the same index will be used by different *SNs* in a neighborhood for mutual communication. Furthermore, the IDs used by a *SN* is dynamic. So, knowledge of the index as well cannot help the adversary to correctly infer the communicating *SNs*. We do not intend to address the issue of revocation of a compromised node in this paper. However, we would like to point out how it can be addressed. Generally, the neighborhood size of a node is smaller than the total size of the network. So, each *SN* shall have several free pseudonym sub-ranges from its N sub-ranges. When the *SNs* in a neighborhood identify a compromised *SN*, by whichever mechanism, they can exchange new pseudonym sub-ranges among themselves and use them for anonymous beacon/cluster communication. The compromised *SN* has no means of identifying these ranges as it was not involved in these exchanges. Mutual communication between two *SNs* is not at risk, as the compromised *SNs* have no idea of the sub-ranges used.

C. Anonymity Analysis of SAS protocol

Node anonymity in SAS is due to each *SN* using randomly chosen IDs, from the corresponding pseudonym sub-ranges when it wants to communicate with a given neighbor or broadcast the *BS* beacon. Given some communication between two non-compromised nodes in a neighborhood. A group of colluding, compromised neighbors cannot infer the source/destination of the communication. This is because, a *SN* chooses the pseudonym sub-range for each neighbor randomly. These chosen pseudonym sub-ranges are non-overlapping and non-contiguous. So, there is no way, given the knowledge of the sub-ranges of a *SN*, the colluding nodes can figure out the other sub-ranges the *SN* is using. Thus, our scheme ensures that a sender is guaranteed complete anonymity in communication with an uncompromised *SN* despite the existence of compromised, colluding *SNs*, in the neighborhood. We give below a theorem that further illustrates the level of anonymity provided by SAS.

Theorem 1: Let, $\mathcal{S}$ denote the common neighborhood of k compromised nodes that are colluding. Assume that a unicast communication between two uncompromised nodes in $\mathcal{S}$ is heard by the k colluding nodes:

a) If $|\mathcal{S}| > 1$, then there is no way for the k colluding nodes to identify the sender/receiver of the message.

b) The probability that the k compromised colluding nodes will guess the sender correctly is, $1/|\mathcal{S}|$.

Proof: a) The N sub-ranges for each node are chosen randomly. Further, each node selects a sub-range randomly for each of its neighbors and the beacon. Each neighbor of u shares two pseudonym sub-ranges with it, the beacon sub-range being common to all. The common neighborhood is formed by the k nodes sampling only the packets accessible to all of them. The k colluding nodes know only $k+1$ sub-ranges of a node u in this neighborhood. If $|\mathcal{S}| = 1$, the k nodes shall be able to identify the communication is from u, the only node in $\mathcal{S}$. If $|\mathcal{S}| > 1$, the k nodes cannot identify

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

the sender/receiver of the message, as they cannot infer the pseudonym ranges.

b) The compromised nodes know $k+1$ of the N sub-ranges of each common neighbor. A pseudonym that belongs to any of these sub-ranges will be recognized by the compromised nodes with probability 1, because it is addressed to one of them. Any pseudonym from outside these sub-ranges is chosen uniformly from the remaining $\{N \cdot |\mathcal{S}| - (k+1) \cdot |\mathcal{S}|\}$ sub-ranges. The probability that the pseudonym belongs to a node, $u \in \mathcal{S}$, is $1/|\mathcal{S}|$. Hence, if the colluding nodes try to guess the sender, the chance that their guess is right is only $1/|\mathcal{S}|$. ■

According to theorem 1(b), even if there are only 2 uncompromised in the neighborhood, the probability that the compromised neighbors can even guess correctly which uncompromised node is transmitting, is only $1/2$. The larger the number of uncompromised nodes in a neighborhood, the lower the probability that the compromised nodes can guess the sender of a packet correctly.

D. Memory and Computation Requirements

In the SAS scheme each node u stores 2 sub-ranges for each of its neighbors. u also stores its N pseudonym sub-ranges. Considering a K bits pseudonym space and the neighborhood size of a node upper bounded by M. Each node has to store $4 \cdot K \cdot M + 2 \cdot K \cdot N$ bits for the ranges. For example, for a ID space of 64 bits, with a *CWSN* network of 1000 nodes, and an average neighborhood size of 100 nodes. The memory requirement is, $4 \cdot 64 \cdot 100 + 2 \cdot 64 \cdot 1000 = 153.6$ Kbits $= 19.2$ KB. Assuming 2 bytes for storing the index per entry in the table, the total memory requirement is, 19.4 KB. In a *CWSN* network of $10,000$ nodes, and average neighborhood size of a 1000 nodes, total memory requirement would be 192.2 KB. The TelosB motes [11] we use have an external flash memory of size 1 MB along with the internal RAM of 48 KB. These motes use the Von Neumann architecture, wherein, the complete memory space is accessible for code. Hence, the range information could be stored in the external flash as dynamically loadable modules.

The computation involved in deciphering the sender pseudonym involves, indexing into the pseudonym table using the index pre-pended to the sender pseudonym, and checking if the pseudonym falls in the sender's sub-range. Hence, the total computation complexity for pseudonym checking is, $\mathcal{O}(1)$. There are ways of improving the memory utilization by using hashing, compression, etc, however we do not discuss them in this paper.

VI. Conclusions and Future Work

In this paper, we have discussed the requirements for anonymity in a *CWSN*. We have proposed a memory and computation efficient anonymity scheme that preserves node identity and privacy and ensures complete anonymity. In the future, we would like to implement and test the scheme on a real sensor network application. We would also like to extend our scheme to handle addition of nodes in the network post setup.

References

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: A survey. In *Computer Networks, 38 (4)*, pages 393–422, 2002.

[2] S. Bandyopadhyay and E. Coyle. An energy efficient hierarchical clustering algorithm for wireless sensor networks. In *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (Infocom 2003)*, volume 3, 2003.

[3] R. Blom. An optimal class of symmetric key generation systems. In *Proc. of the EUROCRYPT 84 workshop on Advances in cryptology: theory and application of cryptographic techniques*, pages 335–338, Paris, France, 1985. Springer-Verlag New York, Inc.

[4] M. Chatterjee, S. Das, and D. Turgut. WCA: A Weight Based Clustering Algorithm for mobile ad hoc networks. In *Journal of Cluster Computing (Special Issue on Mobile Ad hoc Networks), vol. 5, pp. 193–204, April*, 2002.

[5] C. Delakouridis, L. Kazatzopoulos, G. F. Marias, and P. Georgiadis. Share the Secret: Enabling Location Privacy in Ubiquitous Environments. In *Lecture Notes in Computer Science*, volume 3479, pages 289–305, 2005.

[6] W. Du, J. Deng, Y. S. Han, P. K. Varshney, J. Katz, and A. Khalili. A pairwise key predistribution scheme for wireless sensor networks. *ACM Trans. Inf. Syst. Secur.*, 8(2):228–258, 2005.

[7] J. Ibriq and I. Mahgoub. Cluster-Based Routing in Wireless Sensor Networks: Issues and Challenges. In *Proceedings of the 2004 Symposium on Performance Evaluation of Computer Telecommunication Systems (SPECTS)*, 2004.

[8] C. Karlof and D. Wagner. Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures. *Elsevier's Ad Hoc Networks Journal, Special Issue on Sensor Network Applications and Protocols*, 1(2–3):293–315, September 2003.

[9] J. Kong and X. Hong. ANODR: ANonymous on Demand Routing with Untraceable Routes for Mobile Ad-Hoc Networks. In *MobiHoc '03: Proceedings of the 4th ACM international symposium on Mobile ad Hoc networking & computing*, pages 291–302, Annapolis, Maryland, USA, 2003. ACM Press.

[10] D. Liu and P. Ning. Establishing pairwise keys in distributed sensor networks. In *CCS '03: Proceedings of the 10th ACM conference on Computer and Communications Security*, pages 52–61, Washington D.C., USA, 2003. ACM Press.

[11] J. Polastre, R. Szewczyk, and D. Culler. Telos: Enabling Ultra-Low Power Wireless Research. In *IPSN/SPOTS '05: Proceedings of the Fourth International Conference on Information Processing in Sensor Networks: Special track on Platform Tools and Design Methods for Network Embedded Sensors*, pages 364–369, Los Angeles, California, USA, 2005.

[12] R. Poosarla, H. Deng, A. Ojha, and D. P. Agarwal. A cluster based secure routing scheme for wireless ad hoc networks. In *IEEE IPCCC, 2004*, pages 171–175, Phoenix, Arizona, 2004.

[13] X. Wu and B. K. Bhargava. AO2P: Ad Hoc On-Demand Position-Based Private Routing Protocol. In *IEEE Trans. Mob. Comput.*, volume 4, pages 335–348, 2005.

[14] F. Ye, G. Zhong, S. Lu, and L. Zhang. PEAS: A Robust Energy Conserving Protocol for Long-lived Sensor Networks. In *International Conference on Distributed Computing Systems (ICDCS)*, pages 28–29, 2003.

[15] O. Younis and S. Fahmy. Distributed clustering in Ad Hoc Sensor Networks: A Hybrid, Energy-Efficient Approach. In *IEEE, INFOCOM 2004*, volume 1, pages 629–640, 2004.

[16] B. Zhu, Z. Wan, M. S. Kankanhalli, F. Bao, and R. H. Deng. Anonymous secure routing in mobile ad-hoc networks. In *LCN '04: Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN'04)*, pages 102–108. IEEE Computer Society, 2004.

[17] S. Zhu, S. Setia, and S. Jajodia. LEAP: Efficient security mechanisms for large-scale distributed sensor networks. In *CCS '03: Proceedings of the 10th ACM conference on Computer and Communications Security*, pages 62–72, Washington D.C., USA, 2003. ACM Press.

[18] S. Zhu, S. Setia, S. Jajodia, and P. Ning. An interleaved hop-by-hop authentication scheme for filtering false data injection in sensor networks. In *IEEE Symposium on Security and Privacy*, pages 259–274, 2004.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.