Evaluating the Impact of Broadcast Rates and Collisions on Fake Source Protocols for Source Location Privacy

Alasdair Thomason, Matthew Leeke, Matthew Bradbury, Arshad Jhumka
Department of Computer Science
University of Warwick, Coventry
United Kingdom, CV4 7AL
{*csukai, matt, csujbt, arshad*}*@dcs.warwick.ac.uk*

***Abstract*—Providing source location privacy has become a relevant issue for protocols used in the context of wireless sensor networks. In particular, where an asset is monitored using a wireless sensor network it is often the case that the location of the asset being monitored should be concealed from those eavesdropping on the network. The use of fake sources represents an approach to addressing the source location privacy problem. This paper explores practical factors for the configuration and application of fake source protocols, with a focus on the interplay between the broadcast rates of sensor nodes, message collisions and achieved privacy. Combined with existing work in energy efficient fake source protocols, these contributions evidence the existence of an effective range of broadcast rates for fake source protocols.**

***Keywords*-Collisions; Distributed Eavesdropper; Fake Source; Security; Source Location Privacy; Wireless Sensor Networks**

I. Introduction

The ongoing development of wireless sensor network (WSN) technology has facilitated the several novel applications. One such application is asset monitoring, where a WSN is used to track the movements or other properties of some valuable entity. Examples of situations where asset monitoring is applied range from safety-critical, e.g, military tracking, to non-critical, e.g., domestic automation. Privacy, which in this context is considered to be the property that information can only be observed by those intended to observe it, is relevant to many asset monitoring applications, including safety-critical and non-critical applications [1].

A WSN operates by having its constituent nodes broadcast messages that are received by some set of surrounding nodes. Operating in this medium means that attackers can intercept broadcast messages. Further, once received, it is possible for an attacker to base attacks or circumventions on broadcast messages. In the context of asset monitoring applications, such attacks will typically focus on identifying the location or properties of the asset being monitored. The privacy threats in WSNs can be classified along two dimensions: (i) content-based or (ii) context-based threats. A content-based threat relates specifically to the contents of the messages broadcast by sensor nodes. In general, such threats focus on the data generated at higher network layers, including sensed value and timestamps. In contrast, context-based privacy threats relate to the circumstances of data sensing and message broadcast. Context is a multi-attribute concept that encompasses situational aspects of broadcast messages, including environmental and temporal issues.

To address content-based threats, nodes launching attacks are typically modelled as Byzantine nodes [2], [3]. A variety of cryptographic techniques have to shown to be effective when addressing content-based threats [4], [5]. However, such techniques are not effective when addressing context-based threats, since the contents of messages are, in general, not utilised by context-based attacks.

Location is an attribute of context that is relevant to asset monitoring applications. For example, when monitoring the movements of an endangered species using a WSN, an attacker wishing to understand the location of the asset could launch a context-based attack. This is an example of where source location privacy (SLP) must be provided, i.e., the origin of the sensed data must be concealed from an attacker. In the SLP problem, a WSN is monitoring an asset. When nodes in the WSN detect the presence of the asset, we refer to these nodes as *source nodes*, they will periodically send messages, over a certain duration, to a dedicated node, referred to as a *sink*, for data collection. If the locations of the source nodes are compromised, directly or indirectly, an attacker will be able to capture the asset.

The use fake sources is one technique for providing SLP [1]. The technique involves a set of nodes being chosen to act as a decoys for real source nodes, i.e., to act as fake sources. It can be shown that, when the set of nodes is the whole network, maximal privacy can be provided [6]. However, this is not practical due to the amount of energy expended by having so many fake sources. In particular, such a scheme would be detrimental to the lifetime of the network. The intention of the fake source technique is to create network traffic that confuses an attacker. Research has addressed the issue of balancing privacy and energy expenditure, though this work did not take into account practical considerations, such as the impact of collisions [7]. Further, it has been shown that the fake source problem is NP-complete, with a heuristic based on broadcast duration and rate being proposed as an effective circumvention [8].

A. Contributions

In the paper it is shown that, whilst the SLP problem can be addressed using fake sources as in [7] and [8], there exist practical rates at which wireless sensor nodes should broadcast in order to be effective in providing privacy and energy efficiency. In particular, it is shown that (i) real and fake source broadcast rates are inversely related to the number of collisions due to message propagation increasing the potential for collisions, (ii) an increase in the proportion of collided messages on a WSN can serve to decreases the privacy afforded, and (iii) reducing the broadcast rate of source nodes in pursuit of energy efficiency and increased yield can curtail privacy.

B. Paper Structure

The remainder of this paper is as follows: Section II provides a brief survey of related research in fake source protocols. Section III provides definitions for the adopted network and attacker models. Section IV discuses the adopted fake source protocols and experimental approach. Section V presents and discusses the results of the experimentation. Section VI concludes the paper with a contribution summary.

II. Related Work

The ability of a protocol to provide SLP depends on the assumed network model. For example, Mehta et al. [6] assumes that an attacker has a small wireless network that captures messages and shows how the attacker network can infer the location of nodes after it intercepts messages. In contrast, Kamat et al. assume a single attacker, who uses the routing protocol used by the WSN to infer the location of the source nodes [1]. Several techniques to handle the SLP problem have been proposed [1] [6] [9] [10] [11]. The focus of this paper is the fake source technique [1].

Research has demonstrated that it is possible to select fake sources such that a tradeoff is achieved between energy expenditure and security [7]. Moreover, the algorithms associated with this research, when parameterised appropriately, have been shown to address solution the NP-completeness of the SLP problem [8]. These developments were based on the observation, made in [1], that permanent fake sources outperform temporary fake sources at the cost of increased energy expenditure. Specifically, research in [8] explored how a hybrid protocol that combines fake sources with different broadcast durations could be used to solve the SLP problem. However, whilst this hybrid approach demonstrated that the broadcast rate of fake source was a significant when addressing the SLP problem, it failed to account for the practicalities associated with WSN deployment. Specifically, the research advocated the adoption of higher broadcast to ensure that sufficient fake messages were received by attackers, though the impact of increased network traffic, particularly with regard to message collisions, was not considered. The intention of this paper is to explore how varying broadcast rate impacts the security that can be provided in a practical situation where collisions may impair the operation of the fake source protocol.

III. Models

This section provides formal definitions for the network and attacker models.

A. System Model

We define a wireless sensor node to be a computing device equipped with a wireless interface and associated with a unique identifier. Communication from a node is modelled with a circular communication range centred on the node. A node is thought to able to exchange data with all devices within its communication range. A *link* exists between two nodes m and m' if both m and m' can communicate with each other.

A WSN is a set of wireless sensor nodes with links between pairs of nodes. We assume that all nodes in the network have the same communication range. This network is modelled as an undirected graph $G = (V, E)$, where the set of vertices V represents the set of N wireless sensor nodes and the set of edges E represents the set of links between the nodes. Two nodes $m \in V$ and $m' \in V$ are said to be 1-hop neighbours (or neighbours) iff $\{m, m'\} \in E$, i.e., m and m' are in each other's communication range. We denote by M the set of m's neighbours. The graph $G = (V, E)$ defines the topology of the network. This paper focuses on grid-like network topology, i.e., network of size $n * n = N$. There exists a distinguished node in the network called a sink S, which is responsible for collecting data. Other nodes $v \in V \setminus \{S\}$ sense data and then route the data to the sink for collection. Any node can be a source of sensed data. We denote the distance between the sink and a node $n \in V$ by δ_n. There exists a relation on V, denoted $\prec_H$, such that $m \prec_H n$ iff $H(\delta_m, \delta_n)$.

Sensor nodes route messages to the sink, generally using data aggregation convergecast protocols [12]. It is assumed that there can be several nodes acting as message sources at the same time. We assume that the network is event-triggered - when a node senses an object of interest, it starts sending messages to the sink over a certain time period.

B. Attacker Model

We consider an attacker to be a set of sensor nodes. It has been proposed that the strength of an attacker can be factored along two main dimensions: (i) presence, and (ii) actions [13] that. Using these two dimensions, a lattice of attacker strengths was developed. Based on this lattice, one type of attacker is considered, namely a *distributed eavesdropping* attacker. There are different implementations of this type of attacker. For example, such an attacker can be a single mobile person or multiple people with sensor nodes to eavesdrop on a network [7]. We consider the single person implementation of the distributed eavesdropper attacker.

We assume that the messages sent by the source are encrypted and that the identifier of the source is included but only the sink can determine a nodes location from its identifier. As a result, even if the attacker is able to break the encryption in a reasonably short time frame it cannot ascertain the source nodes location. We assume the distributed eavesdropper attacker to be equipped with devices, such as antenna and spectrum analysers, so it can measure the angle of arrival of a message and the received signal strength to identify the immediate sender and move to that node. The attacker can not learn the source of a message by merely observing a relayed version of a message but may move at any speed and consume power. In addition, the attacker is assumed to have a large memory to keep track of information such as messages that have been heard and nodes that have been visited.

In assessing the privacy of a system, a worst case scenario should be assumed, hence the attacker is assumed to know the methods being used by the system. Specifically, that the attacker knows (i) the location of the sink node and (ii) the network topology, but cannot infer the location of a message source based on a relayed message, and (iii) the routing algorithm used. The attacker does not know the number of assets being monitored, and the possible location of the asset, i.e., the asset can be randomly located in the network. These assumptions imply that an attacker has no way of determining if a message is a fake or genuine. Apart from these assumptions, the only knowledge a distributed eavesdropper has is that which is deduced by eavesdropping on the network. For example, when an attacker finds a (relayed) message coming from a (legitimate) node within its neighbourhood, the sender of that message can be located. We also assume that the attacker does not know the number of possible assets being monitored, as is common in asset monitoring applications.

IV. Experimental Setup

The objective of the experiments presented in this paper is to explore the interplay of source broadcast rates, message collisions and privacy. In this section the WSN simulation environment and experiment configurations used to produce the results presented are described.

A. Simulation Environment

The simulation environment was based on the JProwler network simulator [14]. JProwler is a discrete event simulator that can accurately model sensor nodes and the communications between them. JProwler provides two radio models, Gaussian and Rayleigh, which determine the signal level of transmissions and the communication range of nodes. The Rayleigh model was used for all experiments because it models the situation where sensor nodes have mobility, which is consistent with the assumption that an attacker can have mobility within a network.

B. Network Configuration

A square grid network layout of size $n \times n$ was used in all experiments, where $n \in \{11, 15, 21, 25\}$, i.e., networks with 121, 225, 441 and 625 nodes respectively. A single source node generated messages and a single sink node collected messages. The source and sink nodes were distinct. The rate at which messages from the real source were generated was varied. The sets of experiments for each network size and parameter configuration were performed for sources located at grid corners. A total of 800 repeats were performed for each source location, and for each combination of parameters. The sink node was located at the centre of the grid. Nodes were located 28 meters apart. The node separation distance was determined analytically, based on the static fading values calculated by the adopted radio model. This separation distance ensured that messages (i) pass through multiple nodes from source to sink, (ii) can move only one hop at a time and (iii) can only be passed to horizontally or vertically adjacent nodes.

C. Fake Source Protocol and Protocol Configuration

The adopted fake source protocol is that developed in [8]. The protocol is a flooding algorithm augmented to address the SLP problem.

Flooding Algorithm: The real source generates an application message, as a result of detecting the asset, and broadcasts it to every node in its neighbourhood. The message contains a sequence number and a field, called *hop*, that keeps track of the hop distance the message has travelled. When a node receives the broadcast message, it checks if the message is new, i.e., whether it has previously observed an identical sequence number. If it is new, the node increments the *hop* value by one and broadcasts the message. This process is repeated until the message reaches the sink. The value of the *hop* count at the sink represents the distance of the real source from the sink.

Augmentations for SLP: When the sink receives the first broadcast message from the source, it broadcasts a fake message. This fake message has with the value of *hop* observed in the genuine message and the sequence number of the message for which fake sources have to be selected. When a node receives a fake message, it checks if it has seen such a message with the sequence number. If it has not, then it checks if the *hop* value is 1. If the *hop* value is greater than 1, the node becomes a *temporary fake source*. This means that the node starts sending *a certain number of messages* for a specified duration. When this duration is over, the node broadcasts the fake message with its *hop* value decremented by 1. If, on the other hand, the *hop* value is 1, the node

Table I: Safety period for each network size and send rate.

Network Size	Safety Period			
	1/sec	2/sec	4/sec	8/sec
11×11	33.58	16.90	8.99	9.41
15×15	49.63	24.85	13.29	14.47
21×21	73.52	36.74	19.78	22.90
25×25	89.80	44.68	24.34	28.52

generates a random number and becomes a *permanent fake source* if the number is greater than a specified threshold. i.e., with certain probability it will transmit fake messages indefinitely. The generation of a random number is done so that the number of permanent fake sources is controlled.

The structure of the messages sent by the temporary and permanent fake sources are identical to those sent by the real source. The only difference is in the payload, where in the case of the fake sources, the payload is random. Based on this, it is assumed an attacker cannot distinguish between a real message and a fake one.

As in [8], the rate at which a temporary or permanent fake source sent their fixed number of messages and the frequency at which source nodes broadcast was varied. Simulations were conducted with the duration over which temporary fakes source sent messages every 1, 2, 3, 4, 5, 6, 7 and 8 seconds and permanent fake sources broadcasted at a rate of just over twice that of the source rate, whilst the frequency at which source nodes broadcast was set to 1, 2, 4 and 8 messages per second. These settings yielded 16 configurations of the fake source algorithm for each network size, giving 128 distinct experiments. Each of these experiments was repeated on 800 occasions, meaning that the results presented are mean calculation based on 102,400 simulations. A message collision was assumed to result in the collided messages being dropped and lost permanently, hence no action could be taken based on their reception.

For clarity, an outline of the fake source protocol used in this paper is given in Figures 1- 4. These figures are reproduction of those proposed in [8].

D. Safety Period

A concept called *safety period* was introduced in [1] to capture the number of messages that has to be sent by the real source before it is detected. In this paper, we use the definition of safety period used in [8]. More specifically, for each network size and source broadcast rate the average time taken to detect the real source, i.e., capture the asset, is calculated using flooding. To ensure that an attacker has sufficient opportunity to detect a real source and to bound simulation time, the result of this calculation is doubled to establish a safety period.The safety period, for each network size and rate, for flooding is shown in Table I.

```
process $j$ - If node is a normal node
variables
    % Messages seen
    messages: set of int init $\emptyset$

    % The distance from the source to this node
    realhop: int init 0;

    % Number of messages seen from source
    messagecounter: int init 0;

    % Ignore choice variable
    ignorechoose: int init 0;

constants
    % Distance to the sink, probability threshold
    $\Delta, \sigma$: int, real;

actions
    % Receiving choose message
    receiveChoose:: rcv$\langle Choose, hash, ssd, hop, count \rangle \rightarrow$
        if (hash $\notin$ messages $\wedge$ ignorechoose $= 0$) then
            messages := messages $\cup$ {hash};
            if ($\Delta = ssd$) then
                possiblyBecomeFS(infinite duration, $\sigma$);
            else
                possiblyBecomeFS(temp duration);
            fi; fi;

    % Receiving fake messaget
    receiveFake:: rcv$\langle Fake, hash \rangle \rightarrow$
        if (hash $\notin$ messages) then
            messages := messages $\cup$ {hash};
            BCAST$\langle Fake, hash \rangle$;
        fi;

    % Receiving normal message
    receiveNormal:: rcv$\langle Normal, hash, ssd, hop, count \rangle \rightarrow$
        if (hash $\notin$ messages) then
            messages := messages $\cup$ {hash};
            messagecounter, realhop := count, hop $+ 1$;
            if (messagecounter $= 1 \wedge$ realhop $<= \frac{3}{4}$ssd) then
                ignorechoose := 1;
            fi;
            BCAST$\langle Normal, hash, ssd, hop + 1, count \rangle$;
        fi;

    % Receiving away messaget
    receiveAway:: rcv$\langle Away, hash, ssd, hop, count \rangle \rightarrow$
        if (hash $\notin$ messages) then
            messages := messages $\cup$ {hash};
            if (messagecounter $<$ count $\vee$ realhop $>$ ssd) then
                BCAST$\langle Choose$, hash$(Away), ssd, hop + 1, count \rangle$;
                possiblyBecomeFS(temp duration);
            fi;
        fi;
```

Figure 1: Source location privacy algorithm - general.

```
process $j$ - If node is Source
variables
    % The number of messages sent
    count: int init 1;

    % rate: how fast messages are sent.
    rate: timer init $\delta$;

constants
    % Distance to the sink
    $\Delta$: int;

actions
    % Sending normal messages
    sendNormal:: timeout(rate) $\rightarrow$
        BCAST$\langle Normal, \mathbf{hash}(Normal), \Delta, 0, count\rangle$;
        count := count $+ 1$;
        set($rate$, $\delta$);
```

Figure 2: Source location privacy algorithm - source.

```
process $j$ - If node is Sink
variables
    % Messages seen
    messages: set of int init $\emptyset$

    % Sink sent indicator
    sinksent: int init 0;

actions
    % Receiving fake message
    receiveFake:: rcv$\langle Fake, hash\rangle$ $\rightarrow$
        if (hash $\notin$ messages) then
            messages := messages $\cup$ {hash};
            BCAST$\langle Fake, hash\rangle$;
        fi;

    % Receiving normal message
    receiveNormal:: rcv$\langle Normal, hash, ssd, hop, count\rangle$ $\rightarrow$
        if (hash $\notin$ messages) then
            messages := messages $\cup$ {hash};
            if (sinksent $= 0$) then
                sinksent := 1;
                BCAST$\langle Away, \mathbf{hash}(Away), ssd, hop + 1, 1\rangle$;
            fi;
        fi;
```

Figure 3: Source location privacy algorithm - sink.

```
process $j$ - If node is fake source
variables
    % rate: how fast messages are sent.
    % duration: how long we will stay a fake source.
    rate, duration: timer init $\alpha, \beta$;

actions
    % Sending fake messages
    sendFake:: timeout(rate) $\rightarrow$
        if (duration $>=$ (currenttime $-$ starttime)) then
            BCAST$\langle Choose, \mathbf{hash}(Choose), ssd, hop + 1, count\rangle$;
            BECOME NORMAL;
        else
            BCAST$\langle Fake, \mathbf{hash}(Fake)\rangle$;
            set($rate$, $\alpha$);
        fi;
```

Figure 4: Source location privacy algorithm - fake source.

V. Results

This section presents the results of the simulation experiments described in Section IV. The results presented demonstrate the existence of an effective range of broadcasts rates by showing that (i) real and fake source broadcast rates are inversely related to the number of collisions due to message propagation increasing the potential for collisions, (ii) an increase in the proportion of collided messages on a WSN can serve to decreases the privacy afforded, and (iii) reducing the broadcast rate of source nodes in pursuit of energy efficiency and increased yield can curtail privacy.

A. The Impact of Broadcast Rates on Collisions

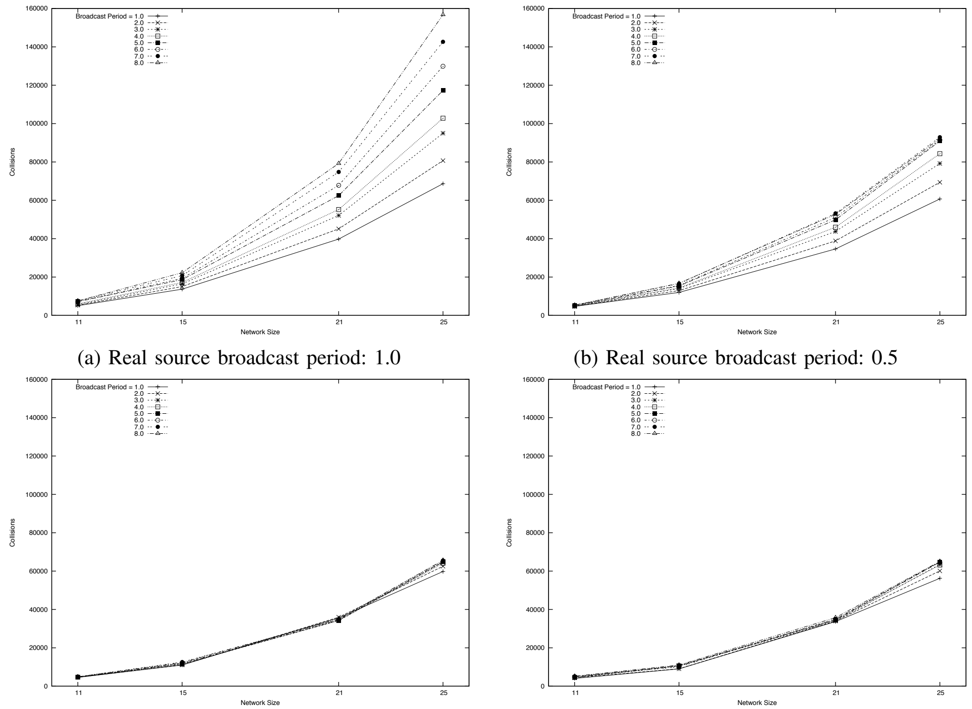
In order to examine the relationship between source node broadcast rates and message collisions, Figure 5 shows the impact of varying the broadcast rate of fake source nodes for various network sizes and real source node broadcast periods. The number of message collisions is plotted against network sizes for fake source broadcast periods of 1, 0.5, 0.25 and 0.125 seconds.

Observe from Figure 5 that increasing the broadcast rate - equivalent to reducing the broadcast period - of fake nodes does not necessarily lead to an increase in the number of collisions, as might be the intuition. This observation can be explained by considering the nature of the fake source protocol and how messages are relayed. To conserve energy, fake sources broadcast a fixed number of messages, which are subsequently received and relayed by neighbouring nodes. If message collides within a small number of hops then the message does not have the opportunity to propagate throughout the network. As a result, that message can not lead to the generation of other messages, of any type, each of which would have had the potential to result in further collision. When broadcast rates are increased there is a higher probability of a message colliding within a few hops of its origin, which means that the overall number of collisions is dramatically reduced. Further, reasoning holds in the context of real source rates, where it can be observed that increasing broadcast rates does not necessarily yield an increase in collisions due to localisation. The data points shown do not make any distinction between the collision of fake or real messages, hence the highest number of collisions can be observed when the rates of the fake and source nodes are lowest, i.e., Figure 5a with a fake node broadcast 8.

The results presented in Figure 5 provide insight regarding the relationship between broadcast rate and collisions. However, based on these results, little can be concluded with regard to the provision of privacy.

B. The Impact of Collisions on SLP

Having shown that collisions are inversely related to broadcast rates in the range considered, it remains to determine whether SLP can be compromised by increased message collisions. Figure 6 shows scatter plots of the mean asset



(a) Real source broadcast period: 1.0

(b) Real source broadcast period: 0.5

(c) Real source broadcast period: 0.25

(d) Real source broadcast period: 0.125

Figure 5: Collisions plotted against network sizes for varied fake source broadcast rates.

capture percentage, across 800 repeats, for the experiments described in Section IV against message collisions. To account for varying networks sizes, the number of collisions has been normalised by dividing by the total number of messages broadcast. This normalisation also has the impact of addressing the effects of the message propagation issue that was identified when examining the number of collisions incurred by particular fake source broadcast rates.

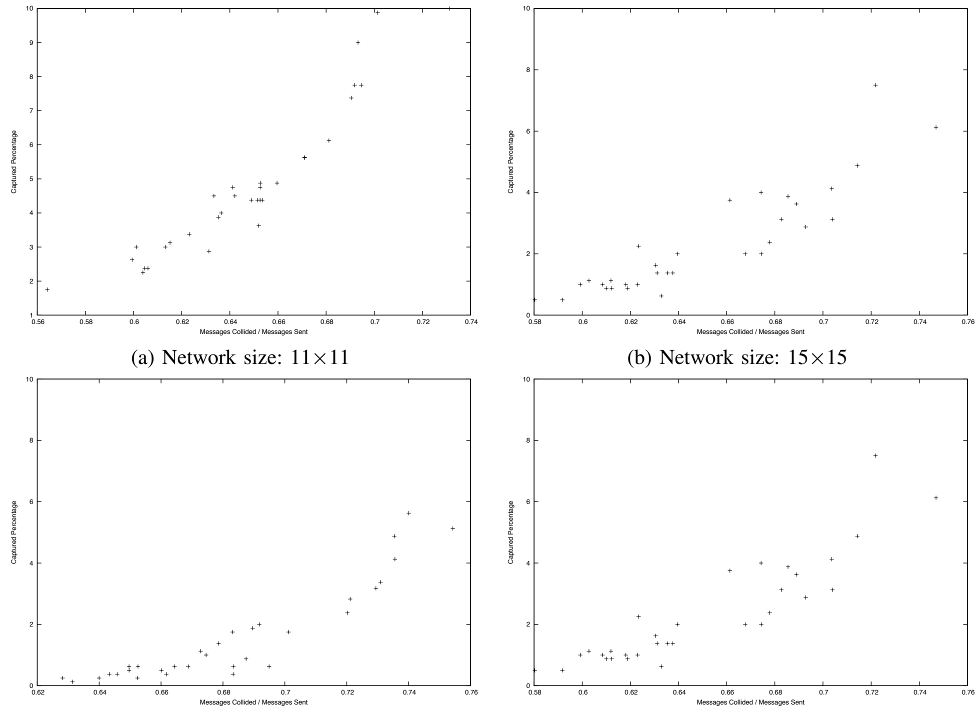
Observe from Figure 5 that, as the proportion of collisions observed in a experiment increases, the asset capture percentage increases, i.e., collisions appear to increase the likelihood of an asset being captured. This suggests that message collisions can be detrimental to the provision of privacy. Indeed, as the fake source approach is founded on the premise that a protocol can engineer traffic to mislead an attacker, it is reasonable that any practical considerations that can impact the nature of network traffic, e.g., collisions, will impact the SLP afforded by the protocol. It is interesting to note that the most pronounced relationships between collisions and privacy, as well as the best attacker performance, can be seen in Figure 6a, which is associated with the smallest network, i.e., 11×11. This is consistent with research in [8], which suggested that attackers based on a distributed eavesdropper model will perform better in smaller networks. Note also that the mean asset capture percentage ranges from almost 0% to 10%, meaning that collisions could be considered to have a significant impact, if indeed they are the true cause, on the performance of fake source protocols in a practical setting.

Having examined the relationship between broadcast rates and message collisions, and gone on to consider the impact of collisions of the provision of SLP, the potential impact of broadcast rates on the provision of SLP is now considered.

C. The Impact of Reducing Broadcast Rates on SLP

Given that increasing the broadcast rate of real and fake sources does not necessarily yield an increase in collisions, and that increasing collisions appears to reduce the privacy afforded by the fake source protocols, it may be considered appropriate to reduce the broadcast rates of real and fake sources. This approach also has the desirable characteristic of increasing the lifetime of the network. To determine the extent to which this is a reasonable approach, Figure 7 shows the mean asset capture percentage, across 800 repeats, for the experiments described in Section IV for various network sizes and real source node broadcast periods. The number of message collisions is plotted against network sizes for fake source broadcast periods of 10, 12, 14 and 16 seconds.

Observe from Figure 7 that, as the fake source broadcast rate decreased, i.e., the broadcast period increases, the privacy provided by the protocol is reduced. This relationships is particularly pronounced in Figure 7d, which depicts the largest network size under test, i.e., 25×25. In this case, the highest mean asset capture percentage is approximately



(a) Network size: 11×11

(b) Network size: 15×15

(c) Network size: 21×21

(d) Network size: 25×25

Figure 6: Asset capture percentage plotted against normalised collisions.

94%, which indicates that poor privacy is afforded. Note that the disparity between real and fake source broadcast rates increases, i.e., when a real source broadcasts at a much higher rate than fake sources, the privacy afforded is decreased. This can be explained by considering the perspective of the attacker. If a real source broadcasts more frequently that a fake source then an attacker is likely to received more real messages than fake messages, meaning that more of their decisions will be based on correct location information, which will facilitate asset capture. In general, it is evident from Figure 7 that decreasing source node broadcast rates beyond a certain threshold will yield degraded privacy, much like increasing rates beyond a certain point will result in a short-lived network [7]. These thresholds will inevitably be application domain specific but, crucially, this paper serves to identify the existence of an effective range of rates.

D. Limitations

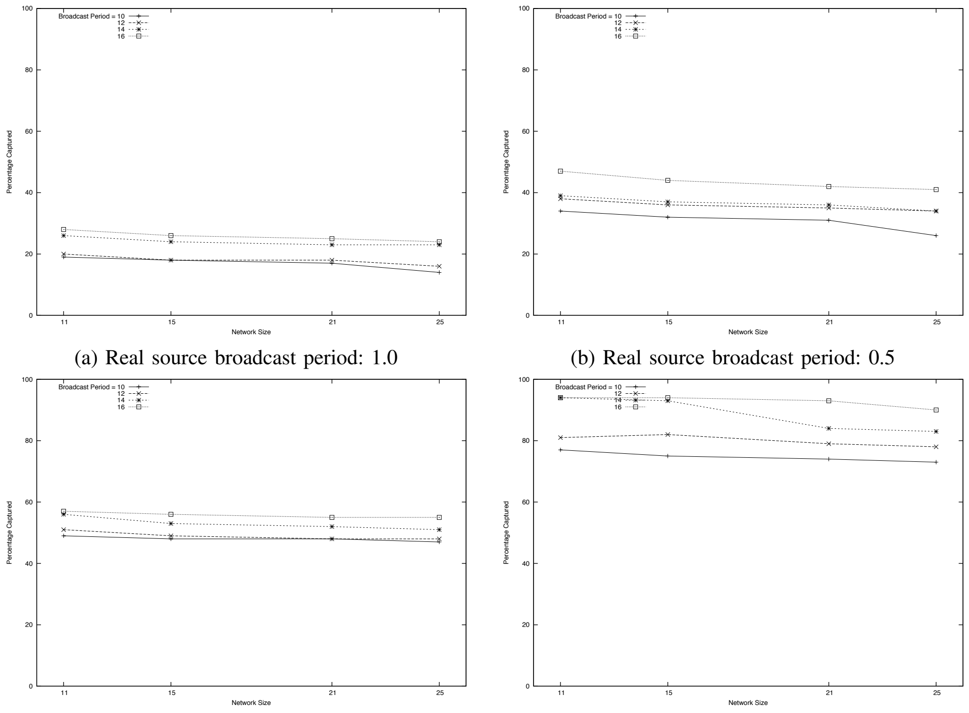
Despite demonstrating a number of practicalities relating to the configuration and application of the fake source technique, the results presented in this paper are limited in their consideration of several factors. Firstly, although the results presented were derived in the context of the general SLP problem and protocols proposed in [8], it is still the case that the adopted protocols can be viewed as specific. Secondly, despite the fake source protocols defining several message types, the analysis presented in this paper does not make distinctions between the types of messages that being generated or colliding. The decision was made on the basis that the analysis should adopt a black-box view of the protocols, providing only configuration details and observing the impact on observable properties, such as afforded privacy. Moreover, no account was taken of the messages that were received by the attacker and how these were impacted by collisions.

VI. Conclusion

In this paper it has been shown that, whilst the SLP problem can be addressed using fake sources as in [7] and [8], there exist practical rates at which wireless sensor nodes should broadcast in order to be effective in providing privacy and energy efficiency. In particular, it is shown that (i) real and fake source broadcast rates are inversely related to the number of collisions due to message propagation increasing the potential for collisions, (ii) an increase in the proportion of collided messages on a WSN can serve to decreases the privacy afforded, and (iii) reducing the broadcast rate of source nodes in pursuit of energy efficiency and increased yield can curtail privacy.

References

[1] U. Kamat, Y. Zhang, and C. Ozturk, “Enhancing source-location privacy in sensor network routing,” in *Proceedings of the 25th IEEE International Conference on Distributed Computing Systems*, November 2005, pp. 599–608.



(a) Real source broadcast period: 1.0

(b) Real source broadcast period: 0.5

(c) Real source broadcast period: 0.250

(d) Real source broadcast period: 0.125

Figure 7: Asset capture percentage plotted against network sizes for varied fake source broadcast rates.

[2] L. Lamport, R. Shostak, and M. Pease, "The byzantine generals problem," *ACM Transactions on Programming Languages and Systems*, vol. 4, no. 3, pp. 382–401, July 1982.

[3] M. Nesterenko and S. Tixeuil, "Discovering network topology in the presence of byzantine faults," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 12, pp. 1777–1789, December 2009.

[4] A. Perrig, J. Stankovic, and D. Wagner, "Security in wireless sensor networks," *Communications of the ACM - Special Issue on Wireless Sensor Networks*, vol. 47, no. 6, pp. 53–57, June 2004.

[5] I. C. Avramopoulos, H. Kobayashi, R. Wang, and A. Krishnamurthy, "Highly secure and efficient routing," in *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies*, March 2004, pp. 197–208.

[6] K. Mehta, D. Liu, and M. Wright, "Location privacy in sensor networks against a global eavesdropper," in *Proceedings of the 15th IEEE International Conference on Network Protocols*, October 2007, pp. 314–323.

[7] A. Jhumka, M. Leeke, and S. Shrestha, "On the use of fake sources for source location privacy: Trade-offs between energy and privacy," *The Computer Journal*, vol. 54, no. 6, pp. 860–874, June 2011.

[8] A. Jhumka, M. Bradbury, and M. Leeke, "Towards understanding source location privacy in wireless sensor networks through fake sources," in *Proceedings of the 11th IEEE International Conference on Trust, Security, and Privacy in Computing and Communications*, June 2012, pp. 760–768.

[9] S. Armenia, G. Morabito, and S. Palazzo, "Analysis of location privacy/energy efficiency tradeoffs in wireless sensor networks," in *Proceedings of the 6th International IFIP-TC6 Networking Conference on Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet*, November 2007, pp. 215–226.

[10] S.-W. Lee, Y.-H. Park, J.-H. Son, S.-W. Seo, U. Kang, H.-K. Moon, and M.-S. Lee, "Source-location privacy in wireless sensor networks," *Korea Institute of Information Security and Cryptology Journal*, vol. 17, no. 2, pp. 125–137, April 2007.

[11] J. Yao and G. Wen, "Preserving source-location privacy in energy-constrained wireless sensor networks," in *Proceedings of the 28th International Conference on Distributed Computing Systems Workshops*, June 2008, pp. 412–416.

[12] A. Jhumka, "Crash-tolerant collision-free data aggregation scheduling in wireless sensor networks," in *Proceedings 29th IEEE Symposium on Reliable Distributed Systems*, October 2010, pp. 44–53.

[13] Z. Benenson, P. M. Cholewinski, and F. C. Freiling, *Wireless Sensor Network Security*. IOS Press, April 2008, ch. Vulnerabilities and Attacks in Wireless Sensor Networks, pp. 22–43.

[14] JProwler, "http://w3.isis.vanderbilt.edu/projects/nest/jprowler/," March 2012.