TARP: *T*iming *A*nalysis *R*esilient *P*rotocol for Wireless Sensor Networks

Adnan Majeed, Ke Liu and Nael Abu-Ghazaleh
Department of Computer Science
State University of New York at Binghamton
(adnan, kliu, nael)@cs.binghamton.edu

***Abstract*— In timing analysis attackers study the transmission pattern of different nodes in a network with the goal of extracting information about users, applications, or the structure of the network, even when the traffic is encrypted. Defeating timing analysis attacks requires expensive traffic mixing measures that equalize the transmission pattern at all nodes; such measures are especially expensive for battery operated wireless devices. In this paper, we first introduce TARP, a traffic mixing approach for defeating timing analysis tailored towards sensor networks. While TARP improves on traffic mixing approaches by combining multiple packets destined to different destinations in a single frame (amortizing packet overhead), traffic mixing remains expensive. To this end, we propose two techniques for improving the energy efficiency of TARP: (1) Using *multi-path routing* to exploit the available capacity engineered to defeat timing analysis; and (2) Adaptive transmission control to allow the transmission pattern to be adapted to the offered load without exposing the structure of the network. Furthermore, we define and explore the notion of relaxed timing analysis resilience where resilience is provided with a limited scope that is well defined in space and/or time. By controlling the scope to fit the application requirements, substantial savings in energy (or delay) can be achieved, while retaining desired levels of timing analysis resilience. Together, the proposed techniques significantly reduce the overhead of TARP, making timing analysis resilience more affordable for critical applications.**

I. Introduction

Technological advances in VLSI, MEMS, and wireless communication have ushered in a new age of miniature, low cost, low-energy, micro-sensors. Networks of such devices, called Wireless Sensor Networks (WSNs) hold the promise of revolutionizing sensing across a range of civil, scientific, military and industrial applications. WSNs differ from conventional networks in a number of respects. They are data driven, often with complex traffic patterns. In addition, several low level services are typically run collaboratively, for example, synchronization, location estimation or data aggregation. Finally, micro-sensors are resource constrained. As a result, sensor networks require new protocols that cater for their unique characteristics and requirements.

WSNs pose new security challenges that require specialized solutions [1], [2]. Because of their self-configuring and collaborative nature, WSNs are vulnerable to attacks on their basic services such as routing and localization. They are also vulnerable to physical attacks on the transducer or the wireless communication channel (jamming). Compromising the data can expose private information, or other types of information of interest to the attacker. Perrig et al summarize security concerns in WSNs [1], which vary significantly with the nature of the application and the network [2].

In this paper, we consider a passive traffic analysis attack on sensor networks [3]. Even if the data and basic services are secure, an attacker can monitor the transmission pattern to discern sensitive information. Such an attack is called a *timing analysis* attack [4]. Timing analysis can expose activity in the network (e.g., tipping off an intruder that they have been detected), or the structure of the network (e.g., where a base station is, allowing a focused jamming attack on it).

Traffic mixing [5] is a form of timing analysis protection that makes the transmission pattern at nodes uniform; orthogonal to the data pattern. We start by proposing a Timing Analysis Resilient Protocol (TARP) that prevents the above mentioned passive attacks. TARP is a traffic mixing protocol tailored towards WSNs. It uses one transmission to send out multiple packets to different neighbors at the same time, amortizing the cost of transmission headers over multiple packets. Furthermore, it achieves timing analysis resilience by de-correlating the transmission pattern from network events.

Timing analysis resilience comes at a high price: the traffic must be equalized in space and time across all nodes. It is critical to reduce this overhead for such techniques to be practical. The two main contributions of this paper: using the capacity available in multiple routes, and adaptive adjustment of the TARP parameters to the offered load, do just that. The paper also introduces the notion of bounded timing analysis resilience and its use for further reductions in overhead. We believe that these techniques together, represent a substantial improvement of overhead with respect to state of the art in this area. Together, they bring timing analysis resilience closer to being practical for critical applications that require it.

The remainder of this paper is organized as follows. We discuss TARP and evaluate it in Section II. The extensions to TARP: multi-path routing and adaptive TARP, are discussed and evaluated in Section III. The notion of bounded timing analysis resilience is introduced in Section IV, related work is discussed in Section V and conclusions are presented in Section VI.

II. TARP: Timing Analysis Resilient Protocol

It is common in WSNs to have a data gathering communication pattern where multiple sensors that detect an event or

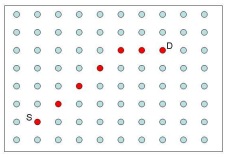


Fig. 1. Point to Point

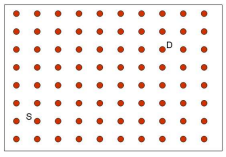


Fig. 2. TARP

are in a position to respond to a query report to a base-station where the data is collected. In Figure 1, the path followed by the packets from a source S towards a base-station D is shown by shaded circles. Each node in this path receives a packet and forwards it. The exceptions to this are S and D; S generates the packets and D consumes them. An observer that simply monitors S, without having access to any information within the packets, can know that an event occurred. Moreover, since wireless sniffers are inexpensive, it is possible that an attacker can overhear all transmissions occurring in the network. Such an attacker can then track the path taken by a packet and expose the structure of the network (e.g., the location of a base station or critical relays).

Such timing analysis attacks can be defeated by de-correlating the transmission pattern from the data generation/forwarding pattern. More specifically, the transmission pattern should be de-correlated (1) temporally, from the data generation and forwarding pattern and (2) spatially, so that the behavior of any location in the network is identical to other locations. This spatial de-correlation defeats attacks such as packet counting to identify differences between nodes and identify communication paths or the structure of the network.

TARP achieves de-correlation by using an approach called traffic mixing [5], [6] whereby the transmission rate at all nodes is identical, achieving both spatial and temporal de-correlation. This transmission pattern is independent of the data being exchanged, hiding the information exploited by timing analysis attacks. TARP specializes traffic mixing to WSNs by allowing every transmitted frame to hold multiple data packets that could be destined to different neighbors. Since the packet is broadcast, each neighbor receives it and is able to extract the packets that are destined to it (reliability can be layered on top of this mechanism if desired). This approach exploits the broadcast nature of the wireless medium to reduce the overhead of communication by combining multiple packets in one packet. In sensor networks where the size of the sensed data is often small, this saving can be substantial.

The TARP protocol works as follows. Each node sets a timer for when to send the next frame according to its local TARP schedule. The TARP schedule has to be invariant across all the nodes to provide spatial de-correlation; one simple way to achieve this invariance is to use a fixed transmission schedule at all nodes. When it is time to transmit, the sender packs up to n packets in the TARP frame where n is the maximum number of slots in the frame. Note that these packets could be destined to different neighbors. If the sender has less than n packets, it fills the extra slots with dummy packets. Each packet can be encrypted with a receiver specific key, or the whole frame could be encrypted with one key. The sender then passes the packet to the MAC protocol for transmission.

Parameter	Value
Simulation Time	250 sec
Number of Nodes	100
Node Layout	10x10 Grid
Internode Distance	26 m
Transmission Range	40 m
Propagation Model	TwoRayGround
MAC Protocol	802.11
Data Packet Size	32 bytes
Number of Packets in a TARP frame	5

TABLE I

SIMULATION PARAMETERS FOR NS-2

TARP is independent of the underlying MAC protocol; it simply reshapes the traffic forwarded from the network layer before handing it to the MAC layer for transmission. The MAC then broadcasts this packet based on its transmission algorithm. A receiver extracts the packets destined to itself, and those it has to forward. Forwarding occurs according to a routing protocol, which is also independent of TARP.

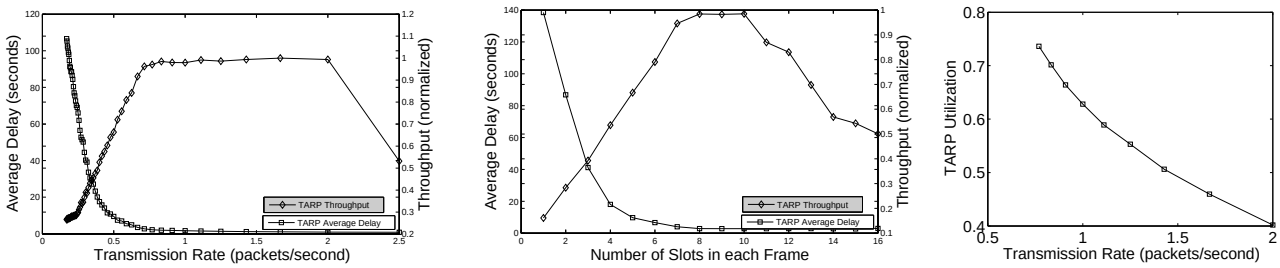
Figure 2 shows the resulting transmission pattern observed in a network with TARP enabled nodes. The traffic pattern is identical to the one in Figure 1 but the transmission pattern is completely different. Using TARP, there is no way of determining the source, destination, path of a flow, or even if there is a flow.

We evaluate TARP with the NS-2 simulator. We implement TARP as a shim layer between the MAC and routing layer. Unless otherwise stated, the simulations use the parameters shown in Table I. We use a scenario where the sink is at the center of the grid and sources are at the left edge of the network. Six sources generate constant bit rate packets at 1 packets/second. This sort of a transmission scenario is typical in a sensor network where a number of nodes sense an event and report it to a base station. Figure 3 presents the behavior of TARP as the number of slots in a TARP frame and the frame transmission frequency are varied. For Figure 3(a) each frame has five slots and in Figure 3(b) the transmission frequency is 1 packet/second. Increasing the capacity of TARP by increasing either of these parameters improves performance, in terms of throughput and delay. However, this improvement is limited by the physical capacity of the medium. When transmission rate or TARP frame size is increased beyond the physical capacity of the channel, performance drops sharply due to excessive collisions.

Figure 3(c) shows another view of the overhead of using TARP - the percentage of transmissions that actually carry data. For low transmission rates the throughput is very low as seen in Figure 3(a), even though there is significant unused capacity in the network (signified by a low utilization). These results show that there is additional capacity in the network that can be utilized.

III. IMPROVEMENTS TO TARP

The overhead incurred for defeating traffic analysis is significant; equalizing traffic in time requires expensive extra transmissions even when little data is being reported, and in areas where little activity is occurring. These additional



(a) Varying Transmission Frequency of TARP Packet (b) Varying Number of Slots in TARP Packet (c) Overhead

Fig. 3. Effect of Varying TARP Parameters on Throughput, Average Delay and Utilization

transmissions have two adverse effects: (1) They drain the energy of sensors unnecessarily; and (2) they can take up available bandwidth reducing the capacity of the network to carry the actual data traffic. Thus, it is critical to develop approaches to reduce the overhead of traffic mixing in order to make it practical for energy and bandwidth challenged networks such as WSNs. This section presents the two main contributions of this paper: (1) multi-path routing; and (2) rate adaptation. Their goal is to reduce the overhead of TARP.

A. M-TARP: Multipath TARP

Whether a node has data or not, it transmits according to the TARP schedule. This produces extra capacity that can be utilized to improve the performance of the network. Instead of sending all traffic through one path, the traffic can be spread over multiple paths to the destination, taking advantage of the available unused capacity in these paths.

The idea of multi-path routing has been investigated in the context of multi-hop wireless networks [7], [8]. In this approach, the routing protocol discovers multiple paths that are then all used to deliver packets to the destination. Multi-path routing may increase capacity or resilience to path failure. However, in conventional multi-hop wireless networks, it may not lead to appreciable improvement in capacity because some of the links making up the different paths may be in interference range with each other competing for the channel (for example, near the sink). For TARP, this effect does not come into play, because multi-path routing simply takes advantage of the available slots in packets that are being sent anyway, and does not require any additional transmissions–it comes for free.

Multipath routing was implemented to run on top of TARP. Each node caches multiple paths to the sinks as it receives their periodic advertisements. These advertisements are the sending node's routing table. The nodes use hop count to determine which paths to use. Each node forwards packets to nodes closer to the destination (in terms of hop-count) or of equal distance to the destination. This enables packets to take slightly longer but less congested paths. We call the resulting protocol M-TARP.

B. A-TARP: Adaptive TARP

The original TARP specification can accommodate pre-planned variation in the traffic rate as long as all nodes follow the same long term pattern; for example, different sensors can alternate between being active and sleeping, and as long as all nodes follow the same pattern of variation over time, timing analysis is defeated. In this way, the pattern can be pre-planned to enable optimizations such as sleeping sensors. In this section, however, we propose to adaptively modify the transmission pattern in response to the current state of the traffic.

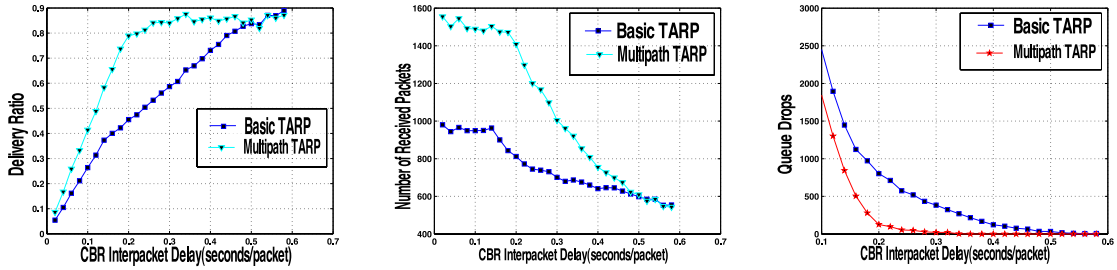
To support a given traffic demand TARP would have to function at a rate that can provide enough capacity for peak demand at hot-spots. In doing so there would be extremely high overhead for non-peak demands. A middle ground would be to have TARP adapt its transmission schedule to cater to current demand. We propose to vary the sending rate at all nodes identically. As a result, TARP is provisioned at the level appropriate for the current traffic, but no higher: it will be able to adapt to low activity periods, reducing overhead in those instances, or adapt up in high activity periods to improve performance in those periods.

Note that adapting the sending rate in this way exposes information about the level of activity at the hot-spots in the network. However, the spatial aspect of traffic analysis is not compromised since all nodes continue to behave identically. This approach represents a relaxed form of the timing analysis resilience requirement that may represent an acceptable trade-off for some applications. We explore the notion of relaxed timing analysis resilience more formally in Section IV.

C. Evaluation of the Extensions

In this section we compare the impact of the two proposed extensions and their combination on TARP performance. Unless otherwise specified, the experimental setup is identical to that described in section II. In Figure 4 we see relative performance of basic and multipath TARP. For delivery ratio, Figure 4(a), multipath TARP utilizes available capacity to give much higher delivery ratio and reaches its peak performance much faster than basic TARP. Basic TARP is limited by the capacity of single path routing and faces much more queue drops. As Figure 4(c) shows, multipath TARP reduces the number of queue drops hence delivering a much higher number of packets (Figure 4(b)). Alternatively, we can achieve the same performance as basic TARP at a significantly lower overhead by using a lower TARP transmission rate and exploiting the extra capacity in other paths.

We used two more scenarios to evaluate the different flavors



(a) Effect on Delivery Ratio (b) Effect on Number of Packets Received (c) Effect on Queue Drops

Fig. 4. Varying Application Transmission rate and Comparing Performance of Basic TARP and Multipath TARP

of TARP. (1) Four nodes each at two corners of the grid sending data to a sink at the opposite corner. This scenario causes cross-traffic to generate a bottleneck at the center of the grid; and (2) Twenty Senders lining the left side of the grid sending data to a single sink on the right side of the grid. This causes data to "funnel" towards the sink, a typical data flow pattern in WSNs. Figure 5 shows the results of these scenarios.

Figure 5 shows the relative performance of two versions of Basic TARP, multi-path TARP (M-TARP), adaptive TARP (A-TARP) and TARP with both multipath and adaptive implemented together (AM-TARP). The two versions of BASIC tarp are TARP Low, with a low transmission rate of one TARP frame every 4 seconds, and TARP High, with a high rate of one TARP frame every 2 seconds. The transmission rate of A-TARP at its maximum and that of TARP High is the same.

TARP High performs much better than TARP Low at high traffic load for the cross traffic scenario, Figure 5(a). However, the better performance comes at a much higher overhead, Figure 5(c). A-TARP chooses the optimal spot between these two approaches. At high load, it adapts its transmission rate upwards and gets high throughput at a high overhead; at low load it adapts downwards and saves on overhead. At high load, A-TARP takes some time to adapt to the traffic load and hence has a lower throughput than TARP High. A-TARP also reduces transmission rate as the queues at the bottleneck nodes are emptied, this contributes to the difference in throughput and delay (Figure 5(d)) performance.

M-TARP also performs better than TARP Low. For M-TARP, the maximum capacity is limited by the number of neighbors of the sink that can forward traffic coming through different paths. In this case, this number is two, giving a performance twice as good as TARP Low. M-TARP performs better in terms of delay by using multiple paths and getting more data across in the same time. This also results in fewer queue drops, Figure 5(b). AM-TARP benefits from both rate adaptation and multipath routing and performs better than the alternatives.

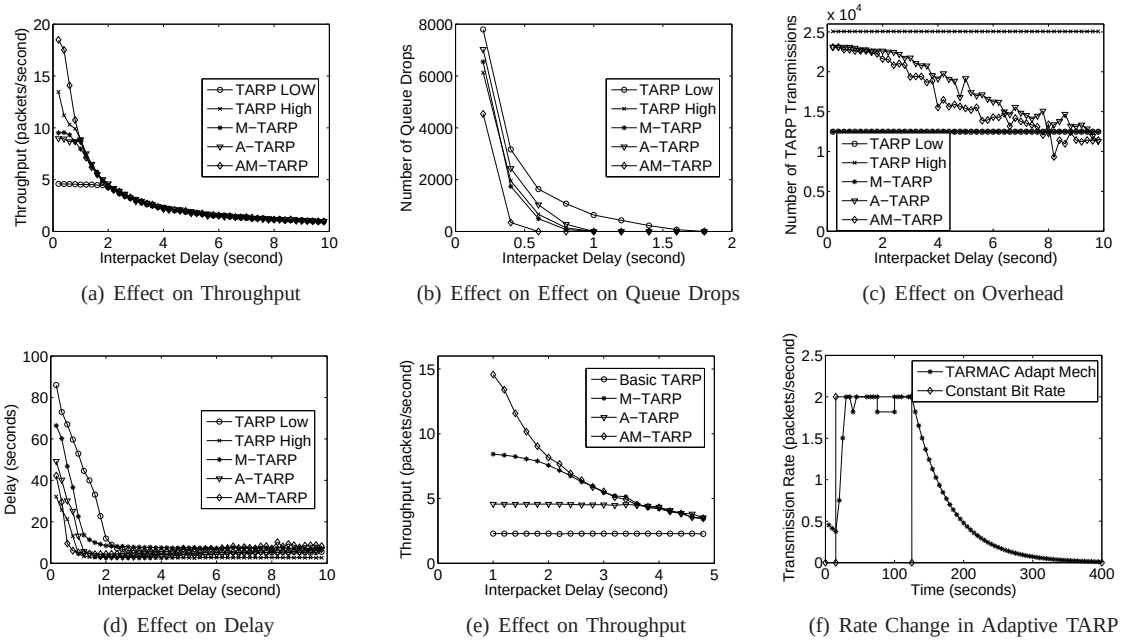
It should be clear that the performance argument has a dual overhead argument – to get the same performance as basic TARP, the proposed techniques can make do with a much lower average TARP transmission rate. In terms of overhead, M-TARP and TARP Low have the same overhead, as shown in figure 5(c). Adapting the transmission rate causes A-TARP and AM-TARP to have high overhead in the presence of high traffic load and low overhead otherwise. However, in both cases A-TARP and AM-TARP maintain high throughput. However, since AM-TARP also has the benefit of using more of the available capacity, it's overhead is lower than A-TARP. This is because AM-TARP does not need to adapt to the highest transmission rate.

To evaluate how A-TARP changes frame transmission rate over time we use the same scenario as described in section II. The senders send data for 100 seconds and then stop sending. Figure 5(f) shows how A-TARP modifies its transmission rate according to the offered traffic. A-TARP adaptively changes its transmission rate according to the load on the network. Rather than fixing the transmission rate, giving packet drops if the rate is too low or unnecessary energy drain if it is too high, A-TARP delivers packet under high load and reduces energy consumption under low load.

IV. Relaxing Requirements–Bounded Timing Resilience

In this section, we introduce the idea of a bounded timing analysis resilience, which relaxes absolute timing analysis resilience to achieve higher energy efficiency or lower delay. TARP provides absolute timing analysis resilience – exposing no information for all time, and across all areas of the network. For most applications, this represents overprotection. By relaxing this requirement to a less restrictive one that still meets the application desired protection level, we may be able to significantly reduce the overhead. For example, adaptive TARP presented in the previous section represents a relaxation of TARP in that it exposes some limited information about the level of activity in the network, but hides all spatial information, to achieve significant savings in energy level.

In this section, we explore one important example of relaxation of TARP. Specifically, for many applications, it is likely that timing analysis resilience is of interest only for a bounded amount of time. When the behavior of the network changes, we desire, with a certain probability, that an observer is not able to discover the change for a given period of time. For example, we may want a 95% confidence that it will take intruders ten minutes or more to discover that they were detected (as evidence a higher reporting rate of the sensors). If ten minutes represent sufficient time for security to move in and apprehend the intruder, this time represents sufficient



(a) Effect on Throughput (b) Effect on Effect on Queue Drops (c) Effect on Overhead

(d) Effect on Delay (e) Effect on Throughput (f) Rate Change in Adaptive TARP

Fig. 5. Performance Comparison of Various TARP Flavors. Effect of Varying CBR Traffic Rate.

protection from timing analysis. Absolute protection requires that we provision the network for all time at the higher rate needed when the network detects an intruder–we are able to significantly reduce this overhead with relaxation. In contrast, adaptive TARP is insufficient here because it only provides spatial but not temporal protection; the intruder can surmise from the change in reporting rate that she has been detected.

We consider a version of TARP where all nodes transmit at the same rate according to an exponential probability distribution. Note that this version satisfies the TARP requirement since all nodes are behaving identically at all times. We develop the probability of detection discretely, assuming transmissions occur in slots; however, a continuous formulation is also possible. Assume that the probability of transmission at a given time slot is T_{slow}; an attacker can estimate this probability by observing the number of transmissions over time. The timing analysis problem is then to detect whether the transmission rate has increased since it signifies that the attacker has been detected. In the active mode (e.g., when an intruder is detected), the transmission probability increases to T_{fast} as the sensors report data more frequently. Thus, instead of having to provision for T_{fast} at all times, we can make do with the lower T_{slow}.

Assume that an intruder takes a sample at the start of the observation period and determines the mean and standard deviation over n time slots. To determine whether the transmission rate has increased, the observer must test the hypothesis that the rate has changed. For example, if a 95% confidence is needed that a change is not detected the following inequality must be satisfied

$$T_{slow} \leq T_{fast} - \frac{1.96}{T_{fast} * n * \sqrt{n}}$$

[9] where n is the number of slots (which can be mapped to detection time) and 1.96 is the value (or alternatively normal distribution value) for the desired confidence level. For example, the equation can provide the value of T_{slow} for a known T_{fast} maximum transmission rate with the desired confidence level, for a given n number of slots before detection. The overhead saving ratio of T_{slow} to T_{fast} is shown for different values of n at 95% confidence in Figure 6. As the value

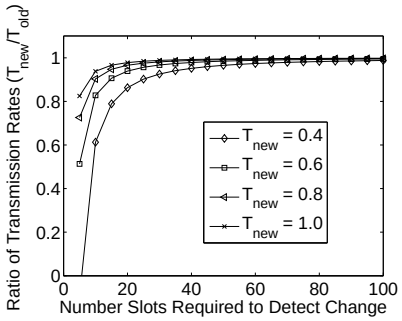


Fig. 6. Number of Time Slots Required to Observe a Difference in Transmission Rate

of n increases the allowable (undetectable) change in T_{slow} reduces. In other words, larger values of n give a higher confidence with which a change in transmission probability is detected. However, a larger value of n translates to more time required to determine that an event was detected. Hence, a node can increase its transmission rate briefly to deliver the urgent data and then shift back to the original sending rate without the attacker detecting the change. Therefore, n is set based on the time that needs to pass before timing analysis fails and an attacker has 95% confidence or higher that the rate has changed. If n is set to ∞ we have absolute TARP; the smaller the value of desired protection period n, the more energy can be saved. The above example is the behavior of one node's transmission, however, this same approach can be used

to activate a path by controlling the transmission probability of the whole path together.

V. Related Work

Several existing works focus on defeating traffic analysis [5], [10]–[14] based on masking flows, or hiding source and/or destination. They do not consider the problem of an adversary determining the topology of the network or the location of critical features in the network. Preserving of location privacy of sensor nodes are also studied by Kong et al [15], who try to protect the location information and identity of each single sensor node from detection. Xi et al. [16] show how accurate location analysis can be achieved by an adversary and propose a *random walk* solution to prevent the compromise of the data sink location.

Deng et al. [17] discuss the traffic analysis problem as it relates to Multi-Hop Wireless Networks (MHWNs). They isolate the properties that enable traffic analysis. In a followup work [18] the authors propose using randomized paths and false paths (not leading to their destination) to prevent such attacks.

These schemes are at the routing layer and require a particular routing protocol. They also do not prevent analysis of control packet traffic. The communication between nodes is point-to-point with built in reliability that has to be used for the protocol to function correctly. This ties down the applicability of the protocols to specific and limited scenarios.

The work closest to TARP is ANODR [6]. ANODR proposes transmitting packets at constant rate in the context of ad hoc networks. Packets are buffered at a node until it is time for that node to transmit. At such a time, a fixed number of packets are transmitted from the node's buffer. The major differences between ANODR and TARP is our focus on WSN and energy efficiency. ANODR transmits single slotted fixed-size packets where as TARP transmits frames; given the small sample sizes commonly used in sensor networks, multi-slotted frames decrease transmission overheads. Furthermore, we use multi-path routing to harvest the available capacity present in the network, and rate adapting TARP transmissions to meet hot-spot demands. In addition, we also contribute and explore the concept of relaxed timing analysis resilience. The resulting energy efficiency is critical for making timing analysis resilience feasible in WSNs.

VI. Concluding Remarks

In this paper, we investigated techniques for making timing analysis resilience approaches efficient in the context of WSNs. We propose TARP, a traffic mixing approach that uses a single packet with multiple slots; this amortizes the overhead associated with packet transmission over multiple data samples. In TARP, all nodes transmit using identical patterns, completely decorrelating transmissions from data, and making all nodes appear identical. We then propose using multi-path routing and adaptive rate control to improve performance of TARP and reduce overhead. Finally, we explore the notion of relaxed traffic analysis resilience to achieve further savings in performance (but restricting the protection from timing analysis resilience).

Acknowledgements

This work was partially supported by National Science Foundation grants CNS-0751161 and CNS-0916323, as well as a US Air Force Research Lab grant FA8750-09-1-0137.

References

[1] A. Perrig, J. Stankovic, and D. Wagner, "Security in wireless sensor networks," *Commun. ACM*, vol. 47, no. 6, pp. 53–57, 2004.

[2] E. Sabbah, A. Majeed, K.-D. Kang, K. Liu, and N. Abu-Ghazaleh, "An application-driven perspective on wireless sensor network security," in *Q2SWinet '06: Proceedings of the 2nd ACM international workshop on Quality of service & security for wireless and mobile networks*. New York, NY, USA: ACM Press, 2006, pp. 1–8.

[3] A. Back, U. Møller, and A. Stiglic, "Traffic analysis attacks and trade-offs in anonymity providing systems," in *Proceedings of the 4th Information Hiding Internation Workshop(IHW'01)*, 2001, pp. 245–257.

[4] R. E. Newman, I. S. Moskowitz, P. Syverson, and A. Serjantov, "Metrics for traffic analysis prevention," in *Privacy Enhancing Technologies*, 2004, pp. 48–65, workshop, PET 2004, Revised Selected Papers.

[5] X. Hong, P. Wang, J. Kong, Q. Zheng, and J. Liu, "Effective Probabilistic Approach Protecting Sensor Traffic," in *Proceedings of the IEEE Military Communications Conference (MILCOM'05)*, 2005, pp. 1–7.

[6] 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*. New York, NY, USA: ACM Press, 2003, pp. 291–302.

[7] S. Lee and M. Gerla, "Split multipath routing with maximally disjoint paths in ad hoc networks," in *Proceedings of the IEEE International Conference on Communication (ICC'01)*, 2001, pp. 3201–3205.

[8] S. D. A. Nasipuri, "On-demand multipath routing for mobile ad hoc networks," in *Eight International Conference on Computer Communications and Networks*, 1999, pp. 64–70.

[9] J. C. J. Crawshaw, "A concise course in advanced level statistics," 1990.

[10] S. Jiang, N. Vaidya, and W. Zhao, "A dynamic mix method for wireless ad hoc networks," in *Proceedings of the IEEE Military Communications Conference (MILCOM'01)*, vol. 2, 2001.

[11] X. Fu, Y. Zhu, B. Graham, R. Bettati, W. Zhao, and A. Texas, "On Flow Marking Attacks in Wireless Anonymous Communication Networks," in *Proceedings of the 25th IEEE International Conference onDistributed Computing Systems (ICDCS'05)*, 2005, pp. 493–503.

[12] X. Fu, Y. Guan, B. Graham, R. Bettati, and W. Zhao, "Using Parasite Flows to Camouflage Flow Traffic," in *Proceedings of 3rd Annual IEEE Information Assurance Workshop 2002*, 2002.

[13] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, "Enhancing Source-Location Privacy in Sensor Network Routing," in *Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICDCS'05)*, 2005, pp. 599–608.

[14] S. Jiang, N. Vaidya, and W. Zhao, "Preventing traffic analysis in packet radio networks," in *Proceedings of DARPA Information Survivability Conference & Exposition II (DISCEX'01)*, vol. 2, 2001.

[15] J. Kong, X. Hong, M. Sanadidi, and M. Gerla, "Mobility Changes Anonymity: Mobile Ad Hoc Networks Need Efficient Anonymous Routing," in *Proceedings of the 10th IEEE Symposium on Computers and Communications (ISCC'05)*, 2005, pp. 57–62.

[16] Y. Xi, L. Schwiebert, and W. Shi, "Preserving Source Location Privacy in Monitoring-Based Wireless Sensor Networks," in *Proceedings of the 20th International Parallel and Distributed Processing Symposium (IPDPS'06)*, 2006.

[17] J. Deng, R. Han, and S. Mishra, "Intrusion tolerance and anti-traffic analysis strategies for wireless sensor networks," in *DSN '04: Proceedings of the 2004 International Conference on Dependable Systems and Networks (DSN'04)*. IEEE Computer Society, 2004, p. 637.

[18] ——, "Decorrelating wireless sensor network traffic to inhibit traffic analysis attacks," *Pervasive and Mobile Computing*, vol. 2, no. 2, pp. 159–186, 2006.