Malicious traffic analysis in wireless sensor networks using advanced signal processing techniques

Alexandros Fragkiadakis, Ioannis Askoxylakis
Institute of Computer Science
Foundation for Research and Technology-Hellas (FORTH)
P.O. Box 1385, GR 711 10, Heraklion, Crete, Greece
email: {alfrag, asko}@ics.forth.gr

***Abstract*—The recent advances in micro-sensor hardware technologies, along with the invention of energy-efficient protocols, have enabled a world-wide spread in wireless sensor networks deployment. These networks are used for a large number of purposes, while having small maintenance and deployment costs. However, as these are usually unattended networks, several security threats have emerged. In this work, we show how an adversary can overhear the encrypted wireless transmissions, and detect the periodic components of the wireless traffic that can further reveal the application used in the sensor network. Traffic analysis is performed in a very energy-efficient way using the compressed sensing principles. Furthermore, the periodic components are detected using the Lomb-Scargle periodogram technique.**

***Index Terms*—compressed sensing, malicious traffic analysis, signal processing, energy-efficiency, wireless sensor networks, Lomb-Scargle periodogram, Contiki**

I. Introduction

The recent advances in micro-electro-mechanical systems and low power and highly integrated digital electronics, have enabled the development of low-cost micro-sensors. These devices are used for measuring a number of physical attributes such as temperature, light, humidity, barometric pressure, acceleration, velocity, acoustics, magnetic field, etc [1]. The sensors are not used in isolation but are grouped into the so-called motes. Motes are integrated devices (e.g. [2]) that contain CPU and memory functionalities under a common board. The advances in sensor operating systems (e.g. Contiki, TinyOS) along with the standardization of new protocols (e.g IEEE 802.15.4, Zigbee) and the adoption of already existing networking protocols (IPv4/IPv6), have made feasible the deployment of wireless sensor networks (WSNs).

Nowadays, WSNs are used for a large number of purposes such as for environmental monitoring [3], critical infrastructure protection [4], emergency response and disaster relief [5], life-logging [6], health monitoring [7], surveillance [8], water-use efficiency [9], earthquake localization [10], structural damage detection [11], etc. Their main advantage is that they are easily deployed in large and harsh areas. Information is sensed and collected by (often) battery-operated motes, and transmitted through a multi-hop routing scheme to a central server, known as sink, for further processing. The sink is a node with enhanced hardware capabilities that performs the more complex tasks required, as motes themselves are severe-constrained devices in terms of processing, storage, computation, and power.

As WSNs become worldwide, their security issues have become a major concern. WSNs face a number of security threats at different layers such as: (i) jamming (interference) attacks at the physical layer, (ii) guaranteed time slot attacks at the medium access layer, (iii) sinkhole, wormhole and other routing attacks at the network layer. A number of countermeasures have been introduced for thwarting these attacks, mainly focusing on intrusion detection, and cryptographic schemes [12].

Except the aforementioned attacks that are successfully detected and mitigated using intrusion detection schemes, another type of attack, the *malicious traffic analysis attack*, cannot be detected and easily mitigated. In this attack, an adversary has the role of a passive listener that collects information from the network, and tries to detect and identify different periodic components in the captured network traffic. Essentially, the ultimate scope of the adversary is to detect information such as the type of applications that execute in the WSN, the paths related to the routing algorithm, etc. Such an information disclosure can severely violate the privacy and security of information-sensitive applications, like those used in wireless body area sensor networks [13]. In this work, we show how an adversary by using advanced signal processing techniques, can effectively detect the periodic components in the network traffic, in a very energy-efficient way. Traffic analysis is performed using the Lomb-Scargle periodogram (LSP) technique, while power consumption reduces through the use of the compressed sensing (CS) principles.

This work has been supported in part by the EC Marie Curie project MESH-WISE (FP7-PEOPLE-2012-IAPP: 324515)

978-1-4673-5828-6/13/$31.00 ©2013 IEEE

Related work focuses on the study of traffic analysis that reveals periodic patterns of the captured traffic. As the authors in [14] show, signal processing techniques can be very effective in traffic analysis. We complement this work by considering an adversary that by using CS, significantly reduces the power consumption required for traffic analysis. Other contributions like [13], [15], [16], propose countermeasures against malicious traffic analysis. On the contrary, we work on the attacker side and show how it can perform energy-efficient malicious traffic analysis.

The rest of this work is organized as follows. Section II describes signal processing techniques used for traffic analysis. In Section III we give the background on CS theory. Section IV presents the adversary model, while the performance evaluation is shown in Section V. Finally, conclusions appear in Section VI.

II. Traffic analysis using signal processing techniques

Very often in communication networks, when information has to be protected by eavesdroppers, security primitives like encryption, authentication, and data integrity are used. A second level of protection usually follows with intrusion detection schemes. This is more imperative in WSNs, due to the broadcast nature of the wireless medium. However, regardless the strength of the security algorithm, and the effectiveness of the intrusion detection system, an adversary can still overhear the wireless channel and identify different periodic components by observing the encrypted traffic. These observations will allow him later to infer regarding the applications used or the routing algorithm decisions taken.

The key idea for identifying periodic components in an encrypted traffic is to convert packet traces into signals, and then process these signals using appropriate signal processing techniques [14]. This will allow the identification of prominent recurring frequencies and time-periods. A common spectral processing technique used for periodic component identification is the standard Discrete Fourier Transform (DFT). DFT computes the spectral power densities and requires the encoded signal to be uniformly sampled. Supposing there is a uniformly sampled signal $x(n)$ with N samples, DFT gives a N-point discrete spectrum $X_N(k)$:

$$X_N(k) = \sum_{n=0}^{N-1} x(n) * e^{-j2\pi kn/N} = DFT[x(n)] \tag{1}$$

$X_N(k)$ can be computed using the Fast Fourier Transform (FFT), and the resulted peaks in the spectrum correspond to the periodic components in the observed traffic. However, the resulted spectrum can contain many harmonically related peaks and furthermore, it does not provide a good unbiased estimate in the presence of noise [13]. Another technique available, the Welch Averaged Periodogram [17] (WAP) can give more reliable results, as periodograms' main characteristic is that they can perform well in the presence of noise or interference [14]. WAP utilizes averaging in order to reduce noise influence and is generated by averaging the K separate spectra $X_N^{(r)}$, computes over K different segments of data, each of length L $(\leq N)$

$$P_x(k) = \frac{1}{KU} \sum_{r=0}^{K-1} |X_L^{(r)}(k)|^2 \tag{2}$$

where $X_L^{(r)}(k) = DFT[w(n)x_r(n)]$ and $U = \frac{1}{L}\sum_{n=0}^{L-1} w^2(n)$, where the windowed data $x_r(n)$ is the r^{th} windowed segment of $x(n)$, $w(n)$ is a windowing function that reduces the artifacts caused by the abrupt changes at the end-points of the window, and U is the normalized window power. The peaks given by P_x are real values that correspond to frequencies of event times of arrival.

As mentioned before, WAP can be efficiently used for the detection of periodic events in the presence of noise or interference. The authors in [13] use WAP for the detection of periodic events in a simulated single-hop wireless body area sensor network using the packet time arrivals. However, as packet arrivals in communication networks are inherently unevenly spaced, they result in a signal encoding that is also unevenly spaced. The FFT and WAP methods perform well only when the packet arrivals are evenly spaced. In order to overcome this limitation and perform efficient traffic analysis, a method called as the Lomb-Scargle periodogram (LSP) can be used. LSP is a spectral analysis technique designed for data that are unevenly spaced. Compared to the WAP and FFT techniques, although LSP requires more computational power, it has the added advantage that the input data are sparse, hence they consume less memory [14].

The LSP technique estimates a power spectrum of N points of data for arbitrary angular frequencies. The power density for an angular frequency ω is given by:

$$P_N(\omega) = \frac{1}{2\sigma^2}\{\frac{[\sum_n (h_n - \bar{h})\cos\omega(t_n - \tau)]^2}{\sum_n \cos^2\omega(t_n - \tau)} + \frac{[\sum_n (h_n - \bar{h})\sin\omega(t_n - \tau)]^2}{\sum_n \sin^2\omega(t_n - \tau)}\} \tag{3}$$

where

$$\bar{h} = \frac{1}{N}\sum_{n=0}^{N-1} h_n$$

$$\sigma = \frac{1}{N-1}\sum_{n=0}^{N-1}(h_n - \bar{h})$$

$$\tau = \frac{1}{2\omega}\tan^{-1}(\frac{\sum_n \sin 2\omega t_n}{\sum_n \cos 2\omega t_n})$$

The samples h_n, $n \in [0, N-1]$, are the N unevenly spaced samples of the observed signal at times t_n.

In Section IV we show how the LSP technique is used to reveal the packet flows traversing a simulated WSN. As we are primarily concerned with energy-efficient traffic analysis, the LSP method is used jointly with CS for reducing the number of data required to detect the network flows. In the next section, we describe the background on CS theory.

III. Compressed sensing background

The recently proposed theory of compressed sensing (CS) ([18]) unifies compression and encryption in order to minimize the overhead for data acquisition and sampling in a WSN. CS exploits the signal structure in order to enable a significant reduction in the sampling and computation costs at a central unit. The key principles in the development of CS theory are *sparsity* and *incoherence*. A signal $\mathbf{x} \in \mathbb{R}^N$ is called sparse if most of its elements are zero in a specific transformation basis. Incoherence satisfies the fact that the sampling/sensing waveforms have an extremely dense representation in the basis. Assuming signal $\mathbf{x} \in \mathbb{R}^N$ is sparse in a basis $\mathbf{\Psi}$, it can be written as $\mathbf{x} = \mathbf{\Psi}\mathbf{b}$, where $\mathbf{b} \in \mathbb{R}^N$ is a sparse vector with S non-zero components ($\|\mathbf{b}\|_0 = S$). CS theory proves that an S-sparse signal $\mathbf{x}$ can be reconstructed exactly with high probability from M randomized linear projections of the signal $\mathbf{x}$ into a measurement matrix $\mathbf{\Phi} \in \mathbb{R}^{M\times N}$. The general measurement model is expressed as follows:

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x} = \mathbf{\Phi}\mathbf{\Psi}\mathbf{b} = \mathbf{\Theta}\mathbf{b} \tag{4}$$

where $\mathbf{\Theta} = \mathbf{\Psi}\mathbf{\Phi}$.

The original vector $\mathbf{b}$ and consequently the sparse signal $\mathbf{x}$, is estimated by solving the following ℓ_0-norm constrained optimization problem:

$$\hat{\mathbf{b}} = \arg\min \|\mathbf{b}\|_0 \quad s.t. \quad \mathbf{y} = \mathbf{\Theta}\mathbf{b} \tag{5}$$

where the $\|\mathbf{b}\|_0$ norm counts the number of non-zero components of $\mathbf{b}$. Note that the formulation of the optimization problem in (5) uses an l_0 norm that measures signal sparsity instead than the traditionally used in signal processing applications l_2 norm, which measures signal energy. However, solving (5) is both numerically unstable and NP-complete. For this reason, the ℓ_0 norm can be replaced by the ℓ_1 norm and problem (5) can be rephrased as the following ℓ_1 norm convex relaxation problem:

$$\hat{\mathbf{b}} = \arg\min \|\mathbf{b}\|_1 \quad s.t. \quad \mathbf{y} = \mathbf{\Theta}\mathbf{b}. \tag{6}$$

The ℓ_1 norm ($\|\mathbf{b}\|_1 := \sum_i |b_i|$) can exactly recover the S-sparse signal with high probability using only $M \geq CS\log(N/S)$ measurements ($C \in R^+$) [18]. Finally, the reconstructed signal is given by $\hat{\mathbf{x}} = \mathbf{\Psi}\hat{\mathbf{b}}$. A variety of reconstruction algorithms based on linear programming, convex relaxation, and greedy strategies have been proposed to solve (6). Among them, greedy strategies such as the Orthogonal Matching Pursuit (OMP) [19] are computationally efficient when the signal of interest is highly sparse.

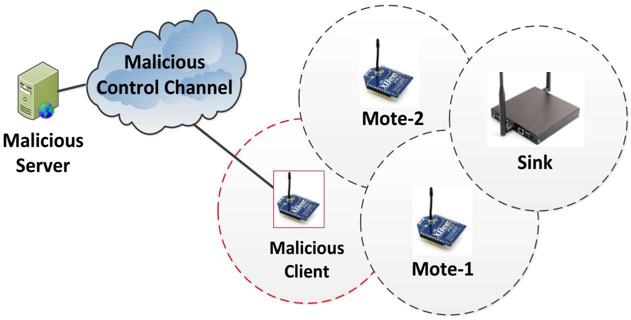

Figure 1: Wireless sensor network topology

IV. Adversary model

The adversary model we consider in this work consists of two distinct entities: (i) the malicious client (MC), and (ii) the malicious server (MS).

MC is a mote with constrained resources (CPU, memory, power) that is positioned in a strategic location within a WSN. Its mission is to observe the wireless traffic and record the timestamps of the captured packets. For this to become feasible, its network interface card is set to promiscuous mode. MC periodically encodes a signal derived from the packet timestamps and compress it, before transmitting it to a more advanced, in terms of resources node (malicious server), for further processing. Figure 1 shows a simulation testbed with two legitimate motes, a single legitimate sink, and the adversary entities (the dotted circles symbolize the transmission ranges of the motes and the MC). MC and MS communicate through a dedicated encrypted malicious control channel (MCC). As it concerns the legitimate WSN, motes periodically transmit sensed data to the sink using different packet transmission rates. Mote-1 transmits with a rate of 10 packets/sec (Flow-1), while Mote-2 transmits with a rate of 17 packets/sec (Flow-2). Therefore, the transmission frequencies of Flow-1 and Flow-2 are 0.1 and 0.059, respectively. The two legitimate motes, the sink, and the MC use ContikiOS [20], an open source operating system for WSNs. The testbed is simulated using Cooja, Contiki's simulator/emulator, while the traffic from the motes towards the sink is encrypted using IPsec [21].

The scope of this paper is to show that MC, jointly with MS can perform energy-efficient malicious traffic analysis. MC records data from the captured traffic that are transmitted to the MS for further processing. As shown later, MS uses the LSP technique (Eq. 3) in order to detect the periodic components of the captured traffic. The malicious traffic analysis is first initiated by MC performing several tasks. First, it overhears the wireless channel, recording the timestamps of the captured packets. Then, it encodes the recorded timestamps into a signal that is suitable for spectrum analysis by the MS using the

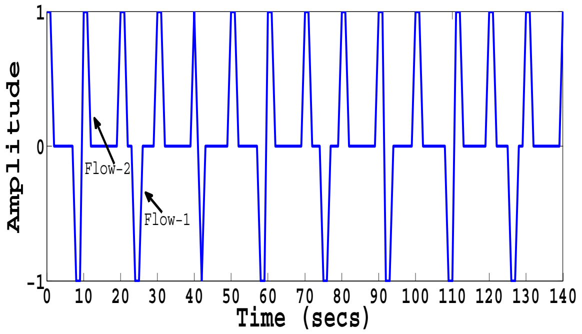

Figure 2: Encoded signal derived from the timestamps of the observed traffic

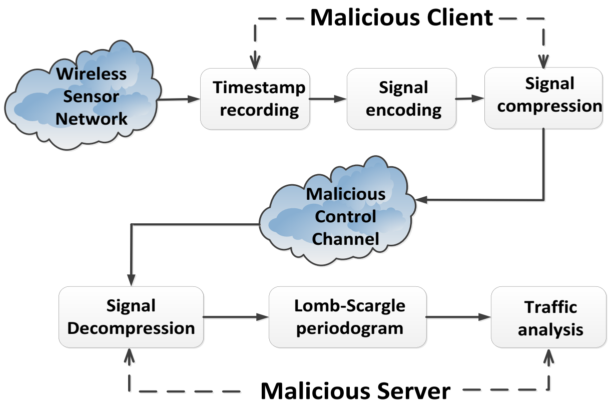

Figure 3: Malicious traffic analysis scheme

LSP technique. For this specific case, where two motes are available, we encode the recorded timestamps by assigning an amplitude of +1 for the packets belonging to Mote-1, and -1 for the Mote-2 packets. Figure 2 shows an example of an encoded signal produced by MC for a 140 seconds packet trace.

After signal encoding takes place, MC compresses the encoded signal using the CS principles in order to minimize the communication cost with the MS. As MC is a severe resource constrained device, saving energy is of paramount importance. It is well known that most of the energy consumption in WSNs is mainly due to the transmission and listening operations performed by the motes. In this paper, we minimize the energy spending due to the transmission operations between the MC and the MS, by compressing the encoded signals in MC using CS. Later on, MS decompresses the signal and feeds the LSP algorithm. As mentioned in Section III, in order to compress a signal $\mathbf{x} \in \mathbb{R}^N$, it has to be sparse in some basis $\mathbf{\Psi}$, and it should be written as $\mathbf{x} = \mathbf{\Psi b}$. Unfortunately, although the encoded signal is sparse in the basis $\mathbf{\Psi} = LSP$, it cannot be expressed as a linear function by using LSP as the orthonormal basis $\mathbf{\Psi}$. For this reason, we follow a different strategy, by compressing the encoded signal at MC by using the FFT transform as the $\mathbf{\Psi}$ basis. We have verified that the encoded signal is also sparse in the frequency domain using FFT. When the MS receives the compressed signal, it decompresses it and feeds the LSP algorithm. Signal compression at the MC involves the use of a transformation matrix $\mathbf{\Phi} \in \mathbb{R}^{M \times N}$. Hence, if $\mathbf{x}$ is the original (uncompressed) encoded signal, MC compress it using Eq. 4, obtaining $\mathbf{y}$, the compressed version of $\mathbf{x}$.

At this point, we have to choose the appropriate measurement matrix. Recent work has shown that when considering measurement matrices built using values selected independently from certain distributions, exact signal recovery can be achieved with high probability. One such choice is the Gaussian distribution used in several works (e.g. [22]). However, the generation of a Gaussian distribution may not be easily achieved in practical implementations, such in this work. The authors in [23] show that Toeplitz matrices with entries drawn from the same distributions (e.g. Gaussian) are also sufficient to recover a signal with high probability. A Toeplitz matrix has several attracting features [23]: (i) it requires the generation of $O(N)$ random variables, while independent and identically distributed (i.i.d.) matrices require the generation of $O(MN)$ variables, (ii) multiplication with a Toeplitz matrix can be performed using FFT and requires only $O(Nlog_2(N))$ operations, compared to i.i.d. matrices that require $O(MN)$ operations, and (iii) i.i.d. matrices are not easily applicable in certain scenarios (e.g., linear-time invariant systems). Considering these features, we select Toeplitz as the measurement matrix.

After MC has compressed the encoded signal using the Toeplitz matrix, it transmits it over the MCC using a suitable protocol over UDP. Figure 3 shows the malicious traffic analysis operations.

V. Performance evaluation

In this section, we show the performance evaluation of the malicious traffic analysis attacks in terms of power consumption, and reconstruction error.

A. *Reconstruction error and spectrum graph fidelity*

As mentioned in the previous section, MC compresses the signal prior to transmission to the MS. The compression ratio used directly affects the power consumption and the reconstruction error, defined as $e = \frac{||x-\hat{x}||_2}{||x||_2}$, where x and $\hat{x}$ are the original and reconstructed signals, respectively. The higher the compression ratio, the lower the power consumption, and the higher the reconstruction error. In order to show the effect of the compression ratio on the reconstruction error, we vary the compression ratio from 5% to 75%, performing CS compression at the MC, and decompression at the MS. We execute simulations for a total of 3 hours in Cooja. Regarding the CS parameters, we choose the Toeplitz as the measurement matrix, FFT as the transformation matrix, and set $N = 200$

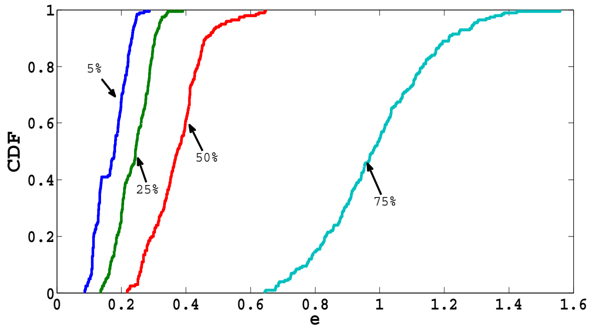

Figure 4: Reconstruction error for different compression ratios

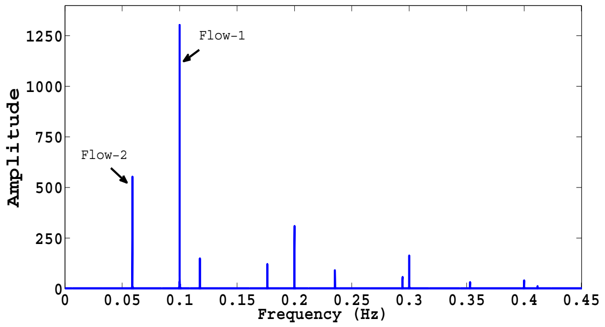

Figure 5: Traffic analysis using the Lomb-Scargle periodogram

the maximum block size of the encoded signal when applying CS. Figure 4 shows the cumulative density function (CDF) of the reconstruction error for the various compression ratios. Essentially, the reconstruction error depicts the fidelity of the reconstructed signal.

Depending on the application, the reconstruction error shown in Figure 4 could be characterized as low, medium, or high for the different compression ratios. In this work, we are primarily interested to decompress the encoded signal, and then use the LSP algorithm in order to find the highest peaks in the frequency domain that signal the basic frequencies of the periodic components in the captured wireless traffic.

Figure 5 shows the spectrum analysis using LSP for an encoded signal that was transmitted from the MC without using CS. The spectrum peaks at the frequencies 0.1 and 0.059 correspond to Flow-1 and Flow-2, respectively. The rest of the peaks correspond to the harmonic frequencies of the flows that can be eliminated by using the appropriate filtering. In Figure 6, we show the traffic analysis revealed by the LSP method when CS is used, and for the different compression rations (that appear on the left side of each graph). For the compression ratios of 5%, 25%, and 50%, the two spectrum peaks clearly reveal the two periodic flows of the WSN. When the compression gets higher (75%), the fidelity of the spectrum graph lowers. This is because, as shown in Figure 4, the reconstruction error significantly increases.

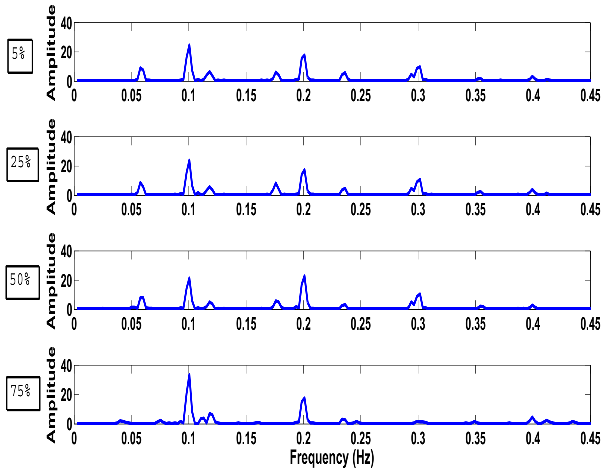

Figure 6: Traffic analysis using the Lomb-Scargle periodogram jointly with Compressed Sensing for different compression ratios

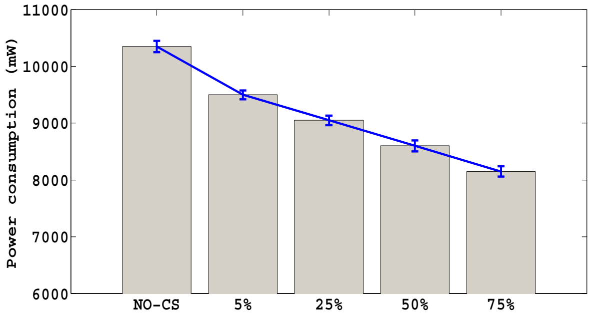

Figure 7: Power consumption of the Malicious Client for the various compression ratios

B. Power consumption

MC periodically encodes the captured timestamps and sends the encoded signals to MS for traffic analysis. As already mentioned in the literature, the power consumption related to packet transmissions is the second highest after that due to the listening operations. We apply CS for compressing the packets MC sends to MS, and so we minimize the power that is consumed for the transmission operations. For measuring the power consumption of the MC, for the different compression ratios applied during CS, we use *powertrace* [24], a built-in power measurement module of Contiki. We simulated a 3-hour run using the topology shown in Figure 1, recording the total power consumption in MC. We repeat this procedure for 50 times, and plot MC's power consumption in Figure 7, where the error bars show the 95% confidence intervals. Observe that as the compression ratio increases, MC's power consumption significantly decreases. This is because less packets are transmitted into the network, hence less power is consumed.

VI. Conclusions

In this work, we presented an adversary model that performs malicious traffic analysis in a WSN. It consists of two distinct entities: a malicious client, and a malicious server. The malicious client overhears the wireless channel, recording the timestamps of the captured packets. The timestamps are then encoded into signals that are compressed according to the compressed sensing principles. The performance evaluation shows that the power consumption significantly reduces as the compression ratio increases. Furthermore, the fidelity of the spectrum graph produced in the malicious server using the LSP method is high, and it successfully reveals the periodic components of the captured wireless traffic for high compression ratios.

References

[1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks, Elsevier*, vol. 52, pp. 2292–2330, 2008.

[2] "Zolertia z1 platform, http://www.zolertia.com/products/z1."

[3] G. Barrenetxea, F. Ingelrest, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange, "SensorScope: Out-of-the-Box Environmental Monitoring," in *Proc. of IPSN*, 2008.

[4] L. Buttyan, D. Gessner, A. Hessler, and P. Langendoerfer, "Application of wireless sensor networks in critical infrastructure protection - challenges and design options," *IEEE Wireless Communications*, vol. 17, pp. 44–49, 2010.

[5] E. Cayirci and T. Coplu, "Sendrom: sensor networks for disaster relief operations management," *Wireless Networks*, vol. 13, pp. 409–423, 2007.

[6] M. Yasutoshi, K. Akiko, and M. Takashi, "Wireless wearable vibration sensor for touch-based life log system," in *Proc. of INSS*, 2012.

[7] A. Milenkovic, C. Otto, and E. Jovanov, "Wireless sensor networks for personal health monitoring: Issues and an implementation," *Computer Communications*, vol. 29, pp. 2521–2533, 2006.

[8] T. He, "Vigilnet: An integrated sensor network system for energy-efficient surveillance," *ACM Transactions on Sensor Networks*, vol. 2, pp. 1–38, 2006.

[9] J. McCulloch, P. McCarthy, S. Guru, W. Peng, D. Hugo, and A. Terhorst, "Wireless sensor network deployment for water use efficiency in irrigation," in *Proc. of REALWSN*, 2008.

[10] G. Werner-Allen, P. Swieskowski, and M. Welsh, "Real-time volcanic earthquake localization," in *Proc. of SenSys*, 2007.

[11] K. Chintalapudi, J. Paek, O. Gnawali, T. Fu, K. Dantu, J. Caffrey, R. Covindan, and E. Johnson, "Structural Damage Detection and Localization Using NETSHM," in *Proc. of IPSN*, 2006.

[12] X. Chen, K. Makki, K. Yen, and N. Pissinou, "Sensor network security: A survey," *IEEE Communications Surveys and Tutorials*, vol. 2, pp. 52–73, 2009.

[13] L. Buttyan and T. Holczerr, "Traffic Analysis Attacks and Countermeasures in Wireless Body Area Sensor Networks," in *Proc. of WoWMom*, 2012.

[14] C. Partridge, D. Cousins, R. K. A. Jackson, T. Saxena, and W. Strayer, "Using Signal Processing to Analyze Wireless Data Traffic," in *Proc. of ACM workshop on Wireless Security*, 2002.

[15] Y. Fan, Y. Jiang, H. Zhu, J. Chen, and X. Shen, "Network coding based privacy preservation against traffic analysis in multi-hop wireless networks," *IEEE Transactions on Wireless Communications*, vol. 10, pp. 834–843, 2011.

[16] J. Deng, R. Han, and S. Mishra, "Countermeasures against traffic analysis attacks in wireless sensor networks," in *Proc. of SECURECOMM*, 2005, pp. 113–126.

[17] P. Welch, "The use of fast fourier transform for estimation of power spectra: A method based on time averaging over short, modified periodograms," *IEEE Transactions on Audio Electroacoustics*, vol. 15, pp. 17–20, 1967.

[18] E. Candes and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, 2008.

[19] J. Tropp and A. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, pp. 4655–4666, 2007.

[20] "The open source os for the internet of things, http://www.contiki-os.org."

[21] S. Raza, T. Voigt, and U. Roedig, "6LoWPAN Extension for IPsec," in *Proc. of Interconnecting Smart Objects with the Internet Workshop*, 2011.

[22] A. Fragkiadakis, S. Nikitaki, and P. Tsakalides, "Physical-layer Intrusion Detection for Wireless Networks using Compressed Sensing," in *Proc. of WiMob*, 2012.

[23] W. Bajwa, J. Haupt, G. Raz, S. Wright, and R. Nowak, "Toeplitz-structured compressed sensing matrices," in *Proc. of SSP*, 2007, pp. 295–298.

[24] A. Dunkels, J. Eriksson, N. Finne, and N. Tsiftes, "Powertrace: Network-level power profiling for low power wireless networks," Tech. Rep.