Proceedings of the Twenty-Ninth International
Florida Artificial Intelligence Research Society Conference

Data Authenticity and Integrity in Wireless Sensor Networks Based on a Watermarking Approach

Farid Lalem,[1] Muath Alshaikh,[1] Ahcène Bounceur,[1] Reinhardt Euler,[1]
Lamri Laouamer,[2] Laurent Nana,[1] Anca Pascu[1]
[1]Lab-STICC UMR CNRS 6285, Université de Bretagne Occidentale, UBO
[2]Qassim University, College of Business and Economics, KSA
Email: Farid.Lalem@univ-brest.fr

Abstract

Wireless Sensor Networks are emerging as an innovative technology that can revolutionize and improve our daily lives. Nevertheless, the use of such a technology raises new challenges regarding the development of reliable and secure systems. Securing WSN is thus imperative and challenging. Unfortunately, the conventional security measures based on data encryption are not well suitable to WSNs due to energy and computational resource constraints. However, watermarking techniques usually have light requirements of resource. This paper proposes a new fully distributed watermarking approach for WSNs. This approach is focused on ensuring integrity and authenticity of data. Moreover, in our approach watermark payload and computational complexity are low. The proposed approach is implemented and simulated with the CupCarbon simulator. The simulation results show that the proposed method is energy efficient[1].

Introduction

Wireless Sensor Networks (WSNs) are distributed embedded systems where each unit is equipped with a certain amount of computation, communication, storage and sensing resources (Wong et al. 2004).

With their capacity of self-organization, they allow to create large scale applications. So far, their deployment is always complex in many domains such as environment, home automation, medicine and military. However, especially for military or medical applications, they need secure and reliable solutions, which seems important to cover that crucial gap. It is proved that security in this type of networks is of strategic importance, even vital, since their proper functioning involves human lives. Moreover, given the fact that sensors are resource intensive, the traditional intensive security algorithms are not well suited for WSNs (Harjito, Potdar, and Singh 2012) (the encryption and decryption process must be done at each node which generates high computation overhead). Furthermore, malicious attacks such as data modification, data deletion and insertion can affect the quality of data collected by sensor nodes. Therefore, protecting data integrity and authenticity is a necessary process to ensure the quality of sensor data before its use for making decisions. Some work based on watermarking techniques has been done to address some of these issues like tampering, data authentication and integrity, copyright and detection (Harjito and Potdar 2015), etc.

Watermarking technique is an interesting mechanism to ensure data integrity and authenticity for WSNs. It is the art of hiding data in the host data in a secure manner, where the authorized user can extract and use that data (Dragoi and Coltuc 2015). It is used for copyright protection, proof of ownership, monitoring of illegal data use, authenticity and other security issues (Wang et al. 2015). Watermarking can be classified based on the embedding technique into spatial and frequency domains. A spatial technique embeds the watermark into the data directly. It is an easy and fast technique but weak regarding some attacks, especially geometric attacks (Sindhush, Rao, and Babu 2015). A frequency technique embeds the watermark into the coefficients of the data. It is robust but more complex than spatial techniques (Benhocine et al. 2013).

Based on the robustness of the watermarking approach, we can classify the technique into robust, semi-fragile and fragile. Robust watermarking should resist against geometric and non-geometric attacks. This kind of approach is useful for copyright and ownership issues (Laouamer and Tayan 2015). Semi-fragile watermarking can accept slight modifications from the authorized user, while fragile watermarking will destroy the watermark in case of any modification. These watermarking approaches aim to prove the integrity and the authenticity of the data (Qi and Xin 2015).

In this paper, we propose a new distributed approach based on a semi-blind watermarking technique which requires only the original watermark for the extraction process. In the step of collecting data, each network node uses the same locally fixed watermark in order to integrate it dynamically into the data packet and transmits it to its neighbors. The watermark verification is performed by the receiver nodes. When the data packet is received by a node, it calculates a new watermark and then compares it with the locally fixed watermark. If their values are not the same then the received data packet is rejected.

The remainder of the paper is organized as follows. In the next section, we present related work. Section III presents the proposed method. In Section IV, the proposed authen-

[1]This work is part of the research project PERSEPTEUR supported by the French National Research Agency ANR.

tication watermarking technique is shown. In Section V, an overview of the CupCarbon simulator is given. Simulation results and a performance analysis compared with other methods are presented in Section IV. Finally, Section VII concludes the paper.

Related Work

In this section we will discuss some existing work dealing with the authenticity and the integrity of data in WSNs using digital watermark based techniques. The technique presented in (Sun et al. 2013) is based on a fragile watermarking method to protect data integrity in WSNs. Collected data from each source sensors are encapsulated into new data packets, which contain diverse data fields for particular sensing sources, no intrusions to the original data are performed. Instead, redundant space of data bytes is employed. Also, source sensors use a one-way hash function for collected data to create watermark information, which is then associated with the data by embedding it into the redundant space of the targeted bytes. At the base station side, a watermarking algorithm is designed to extract the watermark information, which is compared to recalculated watermark information in order to verify the integrity of the data during the transmission.

In (Kamel and Juma 2011), the authors proposed a fragile watermarking algorithm (FWC-D) to detect unauthorized alterations in WSN data streams. FWC-D organizes the sensor data readings into groups of constant sizes. FWC-D uses a hash function, which is applied to the concatenation of all individual data elements in the group along with a secret key to compute the watermark. The hash function can be MD5 or SHA. The watermark is stored in the previous group to make it more difficult for the attacker to insert or delete a complete group without detection. Using the secret key, the receiver can extract the watermark (calculated at the transmitter side) from the received data. To verify the integrity of the received group, the receiver recalculates the watermark and checks against the extracted watermark. If the two watermarks are matching, the group is considered authentic; else, the group is reported as not authentic.

In (Wang et al. 2011), the authors proposed a multiple watermarking method, called Multi-Mark. It consists of an annotation part and a fragile part. At the data source node, the annotation watermark is embedded into the routine monitoring data. Then, the fragile watermark is generated and embedded into the obtained result of the first watermark embedding. When the final watermarked data are transmitted through the WSN any mistakes might happen because of the bad network condition or malicious attacks. They can be detected from the sink, where the tampering detection is based on authenticating fragile watermarking technique. When needed, the annotation watermark can be extracted.

In (Ding et al. 2015), the authors proposed an authentication scheme (RDE) based on a lossless fragile watermarking algorithm for WSNs. Source sensors use a one-way hash function to generate the watermark information depending on the adjacent data and then embed it into these data. After receiving the data, the manager node restores the original data and verifies the reliability. An RDE scheme can verify the sensory data through the embedded watermark bits, and restore the original data completely.

In (Dong and Li 2009), the authors proposed algorithms for identity generation, embedding and detecting. The identity of a sending node was generated by transforming a key and the data collection time. The transformed result formed the watermark is embedded into the data to send. The receiving node judges the authenticity of the data by verifying this watermark. Once the watermark was detected, the data would be stored and transmitted. Otherwise the data would be discarded.

Most of the proposed solutions are based on centralized algorithms in which the watermarked data is sent entirely to be verified either by the cluster head or by the base station. Unfortunately, the centralized models suffer from high communication overheads in transmitting the entire data for verification. As mentioned before, the main part of the energy of any sensor is consumed during the transmission rather than the processing task. Therefore, it is better to consider the distribution of the integrity and authenticity detection in order to minimize the energy consumption. Then, each sensor node can check locally the integrity and the authenticity by extracting the watermarked data from the received data. This allows to obtain quickly the authenticity of data. The node will reject the data in case of a non-authentic watermark, otherwise, it will accept it. Furthermore, if the authentication of the data is checked by the centralized destination and the data is fallacious, all the nodes of the communication path that will route the data to the centralized destination will consume a lot of energy while transmitting fallacious data. This is inappropriate with respect to energy constraints in the sensor nodes.

The Proposed Watermarking Technique

To ensure data integrity and authenticity in wireless sensor networks, we will now present the proposed method that uses a semi-blind watermarking technique. This technique is convenient for the spatial domain since the watermark can be embedded directly into the original data in order to reduce the complexity by avoiding several additional operations and to save the node energy. The method is composed of two phases: an embedding phase and an extraction phase. Depending on its role, each node can act as a transmitter or a receiver.

Watermark Embedding Process

The embedding scheme is illustrated by Figure 1. The embedding of the watermark is done by using the linear interpolation (1) as follows:

$$v' = (1 - \alpha) \cdot w + \alpha \cdot v \quad (1)$$

where v' represents the watermarked data, $\alpha \in]0, 1[$ determines the degree of the visibility and the degradation of the embedded watermark in the original data, w the original watermark data and v represents the original data.

Watermark Extraction Process

The proposed watermarking approach is semi-blind, because the original watermark w and the received watermarked data

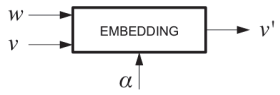

Figure 1: Watermark embedding scheme.

Figure 2: Watermark extraction scheme.

v' are required during the extraction phase in order to calculate the extracted watermark w' (after attack). The integrity and the authenticity of the data is based on this extracted watermark. The extraction scheme is illustrated by Figure 2. Moreover, the extraction process is done using the following linear interpolation (2).

$$w' = (1/\alpha) \cdot w - ((1-\alpha)/\alpha) \cdot v' \quad (2)$$

In our type of WSN, each node has the ability to embed or extract the watermark based on its role in the system. If a node is a transmitter node, then this node has to embed the watermark. However, if a node is a receiver node, then it has to extract the watermark from the watermarked data. The extracted watermark will be compared to the original one in order to prove the data reliability. If the reliability is proven then the original data can be calculated using the inverse of the Equation (1). To achieve high imperceptibility and low degradation, we assign to α the value 0.98 which is close to 1.

The proposed authentication method

In this section, we will present the proposed authentication method for data integrity and authenticity based on digital watermarking in a WSN. First, we will present the general idea of the approach. Then, we will present the algorithms for the embedding and extraction processes.

The proposed model

The flowchart of Figure 3 describes the process of the proposed technique and summarizes the phases executed by each node.

Any node can be a transmitter or a receiver. When a node receives data, it extracts the watermark w' and compares it with the original watermark w. If these values are the same, the node concludes that the data is authentic and accepts to receive it for storing, processing or transmitting. Otherwise, the data will be rejected by the node. If the node is a transmitter, then it embeds the watermark into the data before sending it.

Embedding Algorithm

The main steps of the watermark embedding algorithm, executed in each sensor node S_i, are described as follows. Note,

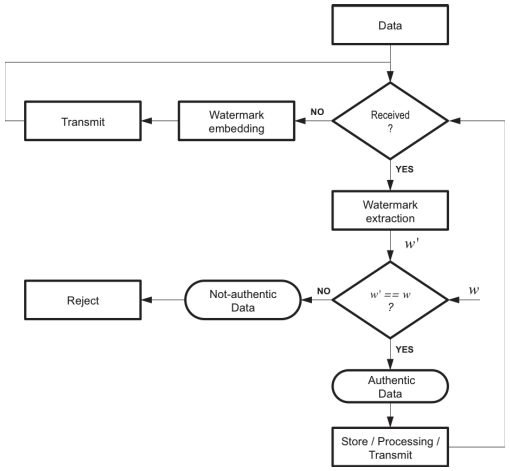

Figure 3: Flowchart of the proposed authentication method.

Algorithm 1 Embedding algorithm

Input: w, α

1: **repeat**
2: v = getSensedValue()
3: $v' = (1-\alpha) \cdot w + \alpha \cdot v$
4: send(v', d)
5: **until** false

that the proposed algorithm works with digital data. In the following, we assume for simplicity that each sensor node S_i sends only one digital data v_i to the sensor node S_j. The sensor S_i node will then:

- *Step 1*: prepare the data v_i to send,
- *Step 2*: embed the watermark w into the data v_i using Equation (1), the obtained data will be v'_i,
- *Step 3*: send the watermarked data v'_i to the destination sensor node S_j.

The pseudo-code of the transmitter node is then given by Algorithm 1, where v is the sensed value obtained by the sensor unit of the current sensor node which is obtained using the function `getSensedValue()`. The function `send(`v'`,` d`)` allows to send the watermarked data v' to the sensor node having the identifier d.

Extracting Algorithm

The main steps of the extracting algorithm, executed in each sensor node S_i, are described as follows:

- *Step 1*: read the received data v'_i from the transmitter node S_j,
- *Step 2*: extract the watermark w'_i from the received packet v'_i,
- *Step 3*: compare the extracted watermark w'_i with the original watermark w. If they are equal, then the data is authentic. Otherwise, the sensor node rejects the data and declares the transmitter node as a *malicious node*.

Algorithm 2 Extracting algorithm

Input: w, α

repeat
  v'=read()
  $w' = (1/\alpha) \cdot w - ((1-\alpha)/\alpha) \cdot v'$
  if (w'==w) **then**
    store(v')
  else
    reject(v')
  end if
until false

The receiver pseudo-code is given by Algorithm 2. The function `read()` allows to read the received data by the radio module. The function `store()` allows to store, process or transmit the received data by the radio module. The function `reject()` allows to reject the received data by the radio module. This means that a malicious node is detected.

CupCarbon simulator

CupCarbon (Mehdi et al. 2014)(CupCarbon 2015) is a free and open source simulator used in this work in order to validate the proposed method. It is a Smart City and Internet of Things Wireless Sensor Network (SCI-WSN) simulator. Its objective is to design, visualize, debug and validate distributed algorithms for monitoring, collecting environmental data, etc. and to create environmental scenarios such as fires, gas, mobiles, and generally within educational and scientific projects.

Networks can be designed and prototyped by an ergonomic and easy to use interface using the OpenStreetMap (OSM) framework to deploy sensors directly on the map. It includes a script called SenScript (CupCarbon 2015) which allows to program and to configure each sensor node individually. The energy consumption can be calculated and displayed as a function of the simulated time. This allows to clarify the structure, feasibility and realistic implementation of a network before its real deployment.

Performance Evaluation

The objective of this section is to evaluate by simulation the effectiveness of our approach and its energy efficiency. This allows us to validate whether the proposed approach is really useful for saving data authenticity and integrity. In the context of this work, we have used the platform CupCarbon that allows to visualize the simulation process and offers an easy to use and debug interface. In the following, we present first two attack models that are used in this work, together with the simulation setup, followed by the simulation results, and subsequently, we present a comparison of the proposed method with two existing approaches.

Attack Scenarios

In this work, we focus on two types of attacks that are:

1. Data modification Attack (Tampering Packet): A compromised node modifies all or some of the data that it is supposed to route (Kumar and Reddy 2013).
2. False Data Insertion (Forgery Packet): An adversary can compromise existing nodes and inject a false message with false information. It is also possible that the adversary adds new nodes to the sensor network that feed false data. Such an attack also consumes the energy resources of other sensor nodes (Kamel and Juma 2011).

Simulation Setup

Sensor networks are generated manually in a two dimensional space. Sensor nodes are deployed in a chosen rectangular area, where the communication range of each sensor node is fixed to $100m$ (meters). The sensor nodes are assumed to be static during the simulation. We select M transmitter nodes and K malicious nodes to perform the two attacks cited above. We used the energy model of the TelosB sensor node. Its energy consumption is estimated to $59.2\mu J$ to transmit one byte, and to $28.6\mu J$ to received one byte (Wander et al. 2005). For the battery, we have used the Super Alkaline AALR6 model which is a portable energy source with a capacity of 9580 Joules.

Results and Discussion

To better understand how our approach behaves in the conditions given in the previous subsection, we have used two case studies. In the first case study, as illustrated by Figure 4, we have generated $N = 15$ sensor nodes including $M = 1$ transmitter node and $K = 1$ malicious node. In the second case study, as illustrated by Figure 5, we have generated $N = 100$ sensor nodes including $M = 8$ transmitter nodes and $K = 5$ malicious nodes.

In the first case study, the transmitter node S_5 marked with yellow color is programmed to send data in a broadcast mode after the embedding process. All its neighbor nodes S_1, S_2, S_4, S_6, S_7 and S_{10} will receive this data. These nodes will check the data's authenticity and integrity. A malicious node S_{21} marked with red color is programmed to create either false data with false information or to modify all or some of the received data that it is supposed to route. In these two cases, all neighbor nodes of the malicious node will detect its presence in the network and reject the data routed by it. Figure 4 illustrates the situations when a node receives a data and after the watermark extraction. A node will be marked with a green color when the received data is authentic and with an orange color when a malicious node is detected.

In Figure 5, transmitter nodes are marked with yellow color and malicious nodes are marked with red color. As we can see, all malicious nodes are detected by all their neighbor nodes, i.e., those that are marked with orange color. The nodes that received authentic data from transmitter nodes are marked with green color.

According to simulation results, Figure 6 shows that the proposed watermarking scheme can effectively verify the integrity of the data, and ensure authenticity and reliability when the scheme achieves 100% detection of data tampering and forgery attacks. Moreover, we calculated the BER (Bit Error Rate) for all cases as zero which means that the extracted watermark is perfectly equal to the original water-

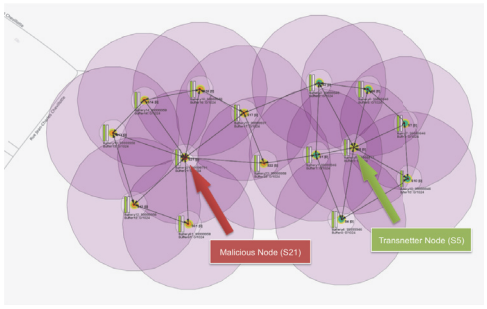

Figure 4: Case study 1: WSN with 15 nodes.

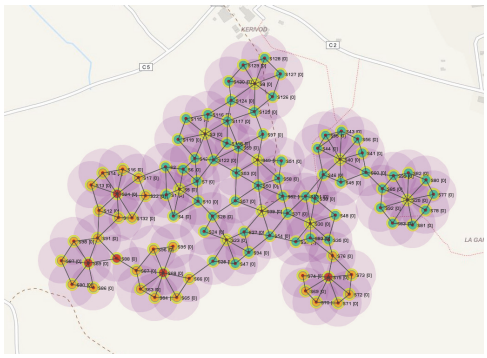

Figure 5: Case study 2: WSN with 100 nodes.

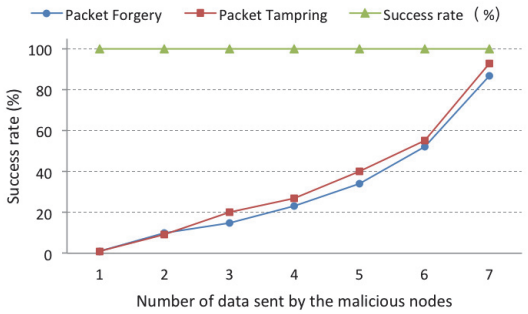

Figure 6: Accuracy of extracted watermark under tampering and forgery attack.

Figure 7: Energy consumption per node.

mark. As a result, our approach is absolutely ensuring integrity and authenticity.

To study the efficiency of the proposed method in terms of energy consumption, we calculated the energy consumed by the sensor nodes in each one of the two considered case studies. For the case study 1, where we have one transmitter node and one malicious node, the energy consumed by each sensor is shown by Figure 7. The energy consumption of each node is the sum of its consumptions in transmission, receiving and processing. We remark that in the first case study the average of the energy consumption is equal to $784.3\mu J$ which represents $8.19 \cdot 10^{-06}\%$ of the capacity of the initial battery. Furthermore, we can also see in Figure 7 that the sensor nodes $S5$ and $S21$ are the most energy consuming. Which is obvious since $S5$ is a transmitter node and $S21$ is a malicious node. Note that these results are directly related to the model of the considered battery.

For the case study 2, presented by Figure 5, where we have 9 transmitter nodes and 5 malicious nodes, the network consumes an average energy equal to $2673.2\mu J$, which represents $2.79 \cdot 10^{-05}\%$ of the capacity of the considered battery.

Classically, when the number of the nodes of a WSN is increasing, the number of data exchanged between these nodes will increase. Therefore, the energy consumption of the network will increase too, which is obvious. However, in our case, the receiver nodes will check the integrity and the authenticity of the received data. If the data is rejected then it will not be routed, which leads to a reduction of the number of exchanged data in the network, which will reduce the energy consumption of the network. Therefore, the proposed approach of watermarking is able to reduce the energy consumption and the traffic of the network considerably due to the self-verification performed by each node.

Note that, in the proposed method, the energy consumption related to data processing, which includes embedding and extraction processes, is negligible compared to the energy consumption of transmitting and receiving data. This is due to the data processing operation as based on a simple computation.

Comparative Study

To compare the proposed method with those of (Wang et al. 2015) and (Sun et al. 2013) we have calculated the number of bits of the embedding watermark payload depending on the number of the data sent and compared it with those of the existing methods. Figure 8 shows the obtained results. In our case, the embedding payload is fixed to 4 bits, which is very small compared with the two considered existing methods. In addition, these methods generate a new watermark for each received data and the false data is detected by the sink (centralized location). This means that if there exist many malicious nodes in the network trying to send false data, all these data will be routed to the sink which increases the energy consumption of the network dramatically due to the very large number of false data exchanged in the network. In

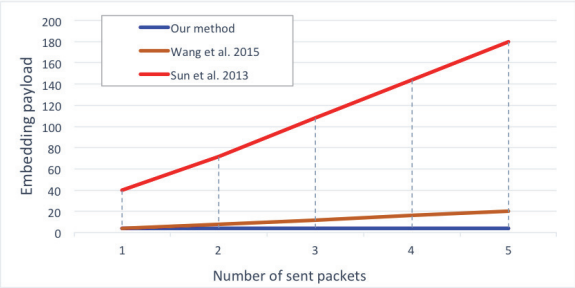

Figure 8: Watermark Payload.

the case of the proposed method, each node is able to verify by itself the integrity and the authenticity of received data and can decide if it rejects or transmits the received data. Therefore, the energy consumption in this case will be reduced significantly and will be very small compared to the one of the other methods.

In addition, the computational complexity of the proposed method in terms of execution time of embedding and extracting processes is very low compared to the other existing methods. In fact, the size of the selected watermark is very small and it does not depend on the size of the collected data. Moreover, the watermark in our method is not repeatedly generated, but fixed in advance and ready to be used by any sensor for embedding or extracting processes. Also, decision making with respect to authenticity and integrity of the received data is done instantly in each sensor while the other methods expect the decision regarding the integrity of data from a centralized location. Hence, the proposed watermarking technique is much faster than the other ones, which makes it very practical for real time WSN applications.

Conclusion

In this paper we have proposed a distributed method based on a semi-blind watermarking technique. Each node in the network verifies the authenticity and integrity of the received data. To study the performance of the proposed method, we have used the simulator CupCarbon. We have considered two types of attacks: data forgery and data tampering. The obtained results show that our method is efficient in terms of energy consumption. Also, the number of exchanged data is reduced drastically by detecting locally in each sensor all the false data. This allows to reduce the network traffic and the energy consumption of the whole network. The distributed approach compared with centralized approach on one hand and the linear interpolation algorithm of watermarking on the other hand leads to a minimum consumption of energy. As future work, we plan to integrate the proposed method on real WSNs which is already designed to perform a given task. This allows us to study the difficulty of this integration.

References

Benhocine, A.; Laouamer, L.; Nana, L.; and Pascu, A. C. 2013. New images watermarking scheme based on singular value decomposition. *Journal of Information Hiding and Multimedia Signal Processing* 4(1):9–18.

CupCarbon. 2015. French national research agency project perspecteur, cupcarbon simulator, http://www.cupcarbon.com.

Ding, Q.; Wang, B.; Sun, X.; Wang, J.; and Shen, J. 2015. A reversible watermarking scheme based on difference expansion for wireless sensor networks. *International Journal of Grid Distribution Computing* 08(2):143–154.

Dong, X., and Li, X. 2009. An authentication method for self nodes based on watermarking in wireless sensor networks. In *5th IEEE International Conference on Wireless Communications, Networking and Mobile Computing, 2009. WiCom'09*, 1–4.

Dragoi, I.-C., and Coltuc, D. 2015. On local prediction based reversible watermarking. *IEEE Transactions on Image Processing* 24(4):1244–1246.

Harjito, B., and Potdar, V. 2015. Secure transmission in wireless sensor networks data using linear kolmogorov watermarking technique. *arXiv preprint arXiv:1501.01376*.

Harjito, B.; Potdar, V.; and Singh, J. 2012. Watermarking technique for wireless multimedia sensor networks: a state of the art. In *Proceedings of the CUBE International Information Technology Conference*, 832–840. ACM.

Kamel, I., and Juma, H. 2011. A lightweight data integrity scheme for sensor networks. *Sensors* 11(4):4118–4136.

Kumar, B. K., and Reddy, G. V. N. 2013. Identification of packet dropping and modification in wireless sensor networks. *International Journal of Computational Engineering Research (IJCER)*.

Laouamer, L., and Tayan, O. 2015. A semi-blind robust dct watermarking approach for sensitive text images. *Arabian Journal for Science and Engineering* 40(4):1097–1109.

Mehdi, K.; Lounis, M.; Bounceur, A.; and Kechadi, T. 2014. Cupcarbon: a multi-agent and discrete event wireless sensor network design and simulation tool. In *Proceedings of the 7th International ICST Conference on Simulation Tools and Techniques*, 126–131. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

Qi, X., and Xin, X. 2015. A singular-value-based semi-fragile watermarking scheme for image content authentication with tamper localization. *Journal of Visual Communication and Image Representation*.

Sindhush, B. S.; Rao, R. K.; and Babu, R. B. 2015. Digital data theft detection using watermarking. *Global Journal of Computer Science and Technology* 14(9).

Sun, X.; Su, J.; Wang, B.; and Liu, Q. 2013. Digital watermarking method for data integrity protection in wireless sensor networks. *Int. Journal of Security and Its Applications* 7(4):407–416.

Wander, A. S.; Gura, N.; Eberle, H.; Gupta, V.; and Shantz, S. C. 2005. Energy analysis of public-key cryptography for wireless sensor networks. In 3^{rd} *IEEE International Conference onPervasive Computing and Communications (PerCom'05)*, 324–328.

Wang, B.; Sun, X.; Ruan, Z.; and Ren, H. 2011. Multi-mark: multiple watermarking method for privacy data protection in wireless sensor networks. *Information Technology Journal* 10(4):833–840.

Wang, B.; Su, J.; Zhang, Y.; Wang, B.; Shen, J.; Ding, Q.; and Sun, X. 2015. A copyright protection method for wireless sensor networks based on digital watermarking. *International Journal of Hybrid Information Technology* 8(6):257–268.

Wong, J. L.; Feng, J.; Kirovski, D.; and Potkonjak, M. 2004. Security in sensor networks: watermarking techniques. In *Book chapter in Wireless sensor networks*. Springer. 305–323.