Two-Levels Verification for Secure Data Aggregation in Resource-Constrained Environments

Omar Rafik Merad Boudia, Hichem Sedjelmaci, Sidi-Mohammed Senouci

Two-Levels Verification for Secure Data Aggregation in Resource-Constrained Environments

Omar Rafik Merad Boudia[1], Hichem Sedjelmaci[2], and Sidi Mohammed Senouci[3]

[1] Département d'Informatique, Université d'Oran 1 Ahmed Ben Bella, Algérie
[2] IRT systemx, Paris Saclay
[3] DRIVE EA1859, Univ. Bourgogne Franche Comté, France
rafik.merad@univ-oran.dz; hichem.sedjelmaci@irt-systemx.fr; Sidi-Mohammed.Senouci@u-bourgogne.fr

***Abstract*—Many resource-constrained applications make use of data aggregation in order to prolong the network lifetime. However, the resource-constrained devices are usually deployed in unattended environments in which providing security is of paramount importance. This leads to various attacks that can occur during data aggregation process. Internal attacks such as selective forwarding represent the most dangerous ones since they cannot be detected by existing cryptography-based protocols proposed to secure data aggregation. In this work, we propose a two-levels verification, in which data is verified using cryptography and intrusion detection techniques. Indeed, a lightweight homomorphic encryption is combined with a game-theory based technique to efficiently secure data aggregation. Our analysis and results show the applicability of the system for aggregation-based resource-constrained applications, especially those considering sensitive information (e.g. health monitoring, military) where the time-efficient detection is crucial.**

Keywords—Homomorphic encryption; IDS; Belief Game; Data aggregation; Internal Attacks

I. INTRODUCTION

Recently, resource-constrained devices have been involved in a wide range of applications e.g. smart home, smart parking, pollution study, medical applications and military applications. A typical Resource-Constrained Environment (RCE) is composed of several constrained devices or more usually "sensor nodes" and a Base Station (BS), which often has the function of managing the other sensors and retrieving and processing data sent by them. Resource-constrained devices generally consist of a microprocessor, a RAM memory, a flash memory as well as a radio transmitter for communicating with other devices. All of these resources are limited in capabilities. For instance, the MicaZ and Tmote Sky sensor platforms which have been widely used in wireless sensor networks applications and then incorporated with Internet of things technologies such as Lora and Sigfox [1].

The possibilities offered by RCE are very promising, but putting up these architectures poses various security problems. In fact, resource-constrained devices are by nature less expensive to produce and more flexible to set up. However, this poses major disadvantages, in the sense that they are usually deployed in unattended environment which makes them prone to different kinds of attacks [2]. As the purpose of deploying these sensors is usually to observe and measure a physical phenomena, and taking into account their limited resources, aggregation of the collected measurements is often more significant and efficient than singular measurements. Data aggregation is one of the resource-preserving techniques, widely considered in resource-constrained applications. Data aggregation process is as follows: leaves devices report their data to another device which is responsible of aggregating the received data and sending the result to the sink node. The aggregated data usually concerns an important number of nodes. From a security point of view, it is risky especially when the aggregator node is attacked [3].

Attacks against aggregation-based applications can be classified into two main categories, namely external attacks and internal attacks. External attacks such as eavesdropping, falsification, and replay can be prevented using cryptography. In fact, many end-to-end solutions have been proposed to avoid such attacks [4-7]. These solutions are based on the property where the aggregation is performed on encrypted data, and thus prevent the aggregator node from accessing to the plain data. Even if such solutions provide a high level of security with, in some cases, an end-to-end verification. These schemes remain insecure against internal adversaries. In fact, attacks such as hello flood, black hole, selective forwarding and wormhole cannot be detected by such proposals. Intrusion Detection Systems (IDS) are employed for this end [8]. For instance, in a battlefield military application, the sink node collects sensitive data about the battlefield. The end-to-end solutions mentioned above can highly secure the system by providing confidentiality, integrity and authentication of data. However, in the case of internal attacks where for example the aggregator selects or ignores some valid and sensitive packets, the system in this case cannot be secure at all, and the consequences lead to catastrophic results, especially in such applications [9]. In this paper, we propose a hybrid solution in which data aggregation is secured using cryptography, while internal adversaries are detected using a technique based on game theory. The contributions of the present paper are as follow:

First, we present our IDS based on Nash Equilibrium (NE), which serves as a decision making agent to validate the aggregated data. Second, we present our scheme, in which data is secured using cryptography. Thanks to the encoding function employed with homomorphic encryption, the sink node can retrieve the individual data and also the detection report of each resource-constrained device. Third, the proposed scheme

allows the base station to calculate any aggregation function on sensors data. This property is very important since it is needed to serve a wide range of resource-constrained applications. Finally, our analysis and results confirm the efficiency of our proposal.

II. BACKGROUND

In this section, we formalize the network and attacker models, we present homomorphic encryption and our cyber detection based on belief game, and finally we identify our design goals.

A. Network Model

We consider a network topology with a large number of resource-constrained devices (sensors) based on clustering where the Cluster Head (CH) serves as an aggregator node to process data received by its members, as shown in Fig. 1. The data is then sent hierarchically toward the Base Station, which is assumed as a powerful and trusted entity.

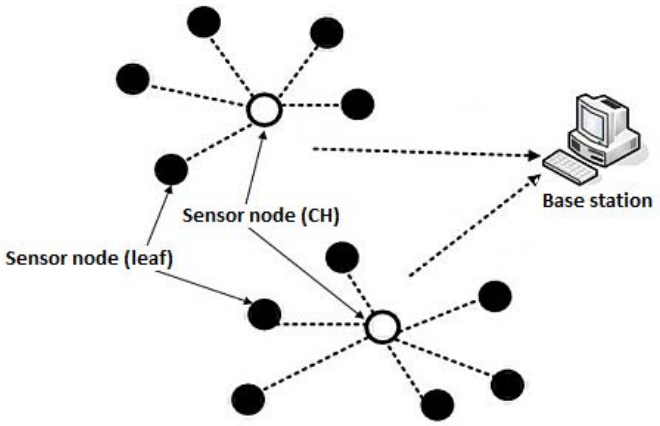

Fig. 1: Network model.

B. Attacker Model

We consider two categories of attackers namely, external and internal attackers. An external attacker A_e can eavesdrop the communication between the nodes in order to probe the content. A_e is able to modify the packet in order to stimulate the final decision. It can even send a valid packet already transmitted, known as replay, in order to deceive the decision-makers. The internal attacker A_i is aware of the cryptography keys and could launch the following cyber-threats: black hole, false data injection and bad mouthing attacks, which aim respectively to drop the packets, alters the gathered data and claim that a well-behaved resource-constrained device is malicious (or vice versa).

C. Homomorphic Encryption (HE)

HE allows calculations on ciphertexts, which have the same effect as performing these calculations on the underlying plaintext data [10]. The HE that supports any function on ciphertexts is known as Fully Homomorphic Encryption (FHE). Introduced by Gentry [11], it is the most significant advance in cryptography in the last few years. It is promising, but the time complexity of its algorithms is still too high for practical use. The other class of HE is Partially Homomorphic Encryption (PHE), which includes encryption schemes that have homomorphic property with respect to one operation.

In this paper, we consider modular addition which is additively homomorphic, a variant of the one-time pad to achieve an additively homomorphic encryption. In order to generate the keys required for encryption, a Key Derivation Function (KDF) is used. More specifically, we consider the secure KDF that uses Pseudo Random Function (HMAC) recommended by NIST, namely NIST SP800-108 HKDF (HMAC-based KDF). This is crucial for the security of the encryption as stated in [12].

D. Cyber detection based on belief game

The proposed cyber detection game is based on belief functions to prevent the cyber threats to alter the aggregated data or send false information to the aggregator node. One of the main reasons for choosing game theory to compute the belief functions is the accuracy to classify the suspected node as malicious or normal [13]. Therefore, a decrease on the false positive and negative rates is achieved. Game theory is perceived as a powerful tool to analyze the interactions among multiple players that are supposed to act toward achieving their own interests. In our case, there are three players, BS, CH and IDS agents. The IDS is embedded at each sensor device and use lightweight detection techniques to monitor the behaviors of its neighbor's nodes. It is noted that, we refer the readers to see this work [14] regarding these lightweight detection policies. Each player chooses the best possible strategy to achieve its own goals (i.e. increase their belief functions), where mutually satisfactory solution should be ensured such that no player has the incentive to individually change its strategy. This optimal solution is defined as a *Nash Equilibrium (NE).*

The game is modeled as $\mu_1(\mathrm{BS}, S_1, R_1)$, $\mu_1(\mathrm{BS}, S_2, R_2)$ and $\mu_1(\mathrm{BS}, S_3, R_3)$. The strategies of the players are defined as follow $S_1=\{x_1, x_2, x_3\}$, $S_2=\{y_1, y_2, y_3\}$ and $S_3=\{z_1, z_2, z_3\}$, where $S_i\ \{i = 1,2,3\}$ are the probability distribution vectors. x_1, x_2 and x_3 are the BS' probabilities of detecting the CH (or IDS) as an intruder, suspect and normal node, respectively. y_1, y_2 and y_3 are the CH's probabilities to be a normal node, an active attacker and a passive attacker, respectively. The active attacker is the malicious node that launches the external and internal attacks (as described in Subsection II.B), where the passive attacker spy only the packets that pass through its radio range. In this research work we assume that, the passive attack could switch to an active attack in order to cause damage in network, e.g. alters the critical information of CH node. z_1, z_2 are the IDS agent's probabilities to provide correct detection and false detection, respectively. The correct detection is defined as a correct confirmation of an IDS towards its neighbor node that sends an alert to CH. This is unlike the false detection, which is a false confirmation of an IDS towards its neighbor or/ and claims the normal CH as malicious. z_3 is the IDS agent's probability to not participate in the decision making, i.e., does not provide any detection.

Here, $\sum_{i=1}^{3} x_i$ =1, $\sum_{i=1}^{3} y_i$ =1 and $\sum_{i=1}^{3} z_i$ =1. ψ_1, ψ_2, ψ_3 are the belief functions of BS, CH and IDS agent, respectively, which increase and decrease depending on the players strategies. These functions are computed as follow:

$$\psi_1(x_1, x_2, x_3) = \alpha_1 . D - \beta_1 . F \quad (1)$$

$D \in [0,1]$ is the detection rate of malicious sensor nodes (CH_j or IDS_k) and $F \in [0,1]$ is the false detection rate (i.e., false positive and false negative). $\alpha_1 + \beta_1 = 1$ and j={1,..,n}, k={1,…,m}, where n and m are respectively the number of CHs and IDSs within the network.

$$\psi_2(y_1, y_2, y_3) = \alpha_2 . G - \beta_2 . B \quad (2)$$

$$\psi_3(z_1, z_2, z_3) = \alpha_3 . G - \beta_3 . B \quad (3)$$

$G \in [0,1]$ is the good belief probability assigned by BS to CH_j (or IDS_k). G=1, when BS classifies CH_j (or IDS_k) as a normal node and G =1/2, when BS classifies CH_j (or IDS_k) as a suspected node. This latter will be suspected to switch to a normal node in the future states. $B \in [0, 1]$ is the bad belief probability assigned by BS to CH_j (or IDS_k). B=1, when BS classifies CH_j (or IDS_k) as an intruder and B=1/2, when BS classifies CH_j (or IDS_k) as a suspected node. This latter will be suspected to switch to a malicious node in the future states. $\alpha_2 + \beta_2 = 1$ and $\alpha_3 + \beta_3 = 1$.

The goal of our cyber detection game is to find a consensus between the malicious sensor devices (occurred at CH or/ and IDS level) and BS. The consensus between these three players is a *NE* optimal solution. Since when the consensus states are reached (defined as *NE* points), no player has the incentive to individually change its strategy. In this security game, the attackers and BS aim respectively to decrease and increase the belief functions by choosing the probabilities values, x_i, y_i and z_i. Hence, BS and malicious sensor devices achieve *NE* solution if the Eq. (4) is met:

$$NE \text{ points} = \begin{cases} [(\max_{S_2} \psi_2(y_1, y_2, y_3), \min_{S_2} \psi_2(y_1, y_2, y_3)] \\ [(\max_{S_3} \psi_3(z_1, z_2, z_3), \min_{S_3} \psi_3(z_1, z_2, z_3)] \end{cases} \quad (4)$$

E. Design Goals

Under the aforementioned system models, the design goals are as follow:

1) Security: The scheme must be secure under the attacker model mentioned above. More specifically, the scheme must be secure against A_e and A_i.

2) Effeciency: The scheme must be efficient in terms of communication and computation overhead, so that the aggregated data can be fast collected by BS.

3) Time-efficient detection: The malicious device is directly and instantly identified even in the case of internal adversary.

III. Our Proposed Scheme

In this section, we propose a scheme, which consists of three phases: system setup, aggregation and verification.

A. Setup Phase

We assume that before deployment, the BS generates its pair of keys (x, Y) where $Y = xG$, and keeps the private key x secret. Each device S_{ij} is loaded with a secret key SK_{ij} shared only with BS and the elliptic curve domain parameters that are the set (Y, E, p, G, n), where Y is the public key and E the elliptic curve over prime field p with the base point G of order n. The sensors are also loaded with a large number M, a PRF-based KDF (NIST SP800- 108 HKDF) and a secure MAC (HMAC). Just after deployment, each sensor device randomly generates a curve point $Y_{ij} = r_{ij}G$ and sends it to the BS. This point will be used to generate the keys needed to secure data aggregation. Also, it sends a list containing the *IDs* of its k neighbor's nodes including the corresponding CH. As a result, the BS knows the neighbors of each sensor device. The choice of k depends on security level (the size of the modulus M). For instance, the neighbor's list of device S_{ij} is as follow: l_{ij} = {$ID_1(CH)$, ID_2,…, ID_k}.

B. Aggregation Phase

The IDS monitors its neighbors' sensor nodes and sends its 2 bits detection report (DR_{ij}) along with its measurement (m_{ij}) to the BS. The report contains the following information: The IDS confirms (or does not confirm) the alert sent by node k. The IDS detects the corresponding CH_j as malicious (regarding the alert sent by node k). To detect the malicious behavior of CH, we refer the reader to [8]. So the data sent is $D_{ij} = DR_{ij} || m_{ij}$. See Fig.2. Note that the choice of w, the length of m_{ij}, also depends on the security level. Assume that λ represents the number of bit needed to represent the data D_{ij} and used in the encoding function. Note that higher is λ lower is the number of nodes per cluster L allowed in the encoding function.

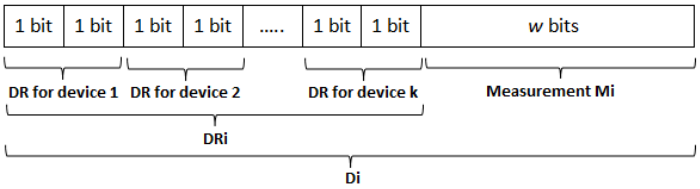

Fig.2. Data format.

By using (r_{ij} , Y_{ij}) and HKDF, each device generates two keys K_{ij1} and K_{ij2} to encrypt and sign the data D_{ij}, namely, $K_{ij} = HKDF(Y_{ij}||r_{ij}Y||Y, N_{ij})$ where $K_{ij} = K_{ij1}|| K_{ij2}$. More specifically, the device does the following:

- Encode D_{ij} into $e_{ij} = D_{ij}||0^r$, where $r=\lambda*(i-1)$.
- $K_{ij} = HKDF(Y_{ij} || r_{ij}Y ||Y, N_{ij})$, $K_{ij} = K_{ij1}|| K_{ij2}$
- Encrypt e_{ij} to $C_{ij} = e_{ij} + K_{ij1} \bmod M$
- Compute MAC on C_{ij} : MAC_{ij} = HMAC (C_{ij}, K_{ij2})
- Send C_{ij} and MAC_{ij} to CH_j

Once received, the data is aggregated at CH_j. The later combines all the ciphertexts and all MACs received. More specifically, the device does the following:

- Compute $C_{agg} = \sum_{i=1..n}^{j} C_{ij} \bmod M$
- Compute $MAC_{agg} = \oplus MAC_{ij}$
- Send C_{agg} and MAC_{agg} to BS or the nearest CH

C. Verification Phase

The 1st verification (MAC Validation):

Once all data received by BS i.e. corresponding to the aggregate of each cluster, the BS invokes the decryption and MAC verification processes:

- Compute all currents K_{ij}
- Compute $e_{agg} = C_{agg} - \sum_{i=1...L}^{j} K_{ij1} \bmod M$
- Decode (e_{agg}, λ): $D_{ij} = e[(i-1)*\lambda, \lambda*i-1]$
- For each D_{ij} compute MAC_{ij}
- Verify the aggregated MAC_{agg}

The 2nd Verification (Attack detection and final validation):

From each D_{ij}, the BS retrieves the measurement m_{ij} and DR_{ij} of S_{ij}. Also, according to the neighbor's list l_{ij} of node S_{ij}, the BS can retrieve from DR_{ij} all DR_k of S_{ij}, see Fig. 2.

Then, the BS compute the *Threat Level (TL)* related to each node_k and CH_j. The *TL*s of CH_j and node_k (IDS_k) are computed as shown in Eq. (5)

$$TL=(TL_{\text{CH}_j}, TL_{\text{node}_k}) \quad (5)$$

where $TL_{\text{CH}_j} = \alpha_4.\bar{C} - \beta_2.C$ and $TL_{\text{node}_k} = \alpha_4.\bar{A} - \beta_2.A.C$ (or $\bar{C}$) is the number of IDSs that confirm (or do not confirm) the forwards of alerts by CH_j, agree (or do not agree) that CH_j does not receive an alert to forward and the CH_j does not exhibit (or exhibits) a malicious behavior. A (or $\bar{A}$)is the number of IDSs that agree (or do not agree) on the detection provided by IDS_k. $\alpha_4+\beta_4=1$

TL_{CH_j} and TL_{node_k} of an attacker that occurred respectively at CH and IDS levels, could be modeled by y_2 and z_2. Therefore, the BS detects the CH and IDS as attackers when Eq (6) and Eq (7) are met, respectively.

$$TL_{\text{CH}_j} > \min_{S_2} \psi_2(y_1, y_2, y_3) \quad (6)$$

$$TL_{\text{node}_k} > \min_{S_3} \psi_3(z_1, z_2, z_3) \quad (7)$$

IV. Security Analysis And Performance

In this section, we analyze the security and evaluate the performance of the proposed protocol in terms of the computation complexity and communication overhead and also accuracy detection.

A. Security Analysis

In what follow, we analyze the security of the scheme against A_e and A_i previously described. In the scheme, the data D is enciphered with a provably secure encryption [12], so even if the communication is eavesdropped, A_e cannot comprehend the content. Also, the encrypted data is signed with an unforgeable MAC protocol, HMAC. So, A_e cannot alter the packet in order to stimulate the final decision. In fact, it cannot forge a valid MAC tag for the ciphertext. Furthermore, the nonce N_{ij} used in HKDF ensures the different keys for every transmission, so A_e cannot replay valid packets already transmitted. In other words, after MAC_{agg} validation in *the first verification*, the BS can be certain that the data aggregation is secured against A_e.

In order to be validated, the data sent to the BS must pass through *the second verification*, where the IDS is involved. The IDSs rely on certain detection policies (e.g., signature and anomaly based detection techniques) to identify the internal attack A_i with a high accuracy as explained [14]. However, these IDSs that are activated at node_k and CH_j could provide false decisions, and hence the detection and false positive rates will be decreased and increased, respectively. To address this issue the BS analyzes the *Threat Levels (TLs)* related to each node_k and CH_j with NE points, as shown in Eq (6) and Eq (7). The node_k and CH_j are detected as attackers when these equations hold.

B. Accuracy detection

To analyze the accuracy detection, we compute the attacks detection rate – the false positive rate (defined also as false alert). We varied the number of attackers (defined in Section II) from 5% to 30% of overall nodes and compute the accuracy detection of the proposed security system. According to Fig 3, it's apparent that, our system has the ability to secure the aggregated data efficiently against the internal and external threats. This result is achieved thanks to the combination between a robust cryptography mechanism and an accurate security game. Indeed a wise choice of two level of security can prevent the occurrence of cyber threats, even when the number of this latter increase, as shown in Fig 3.

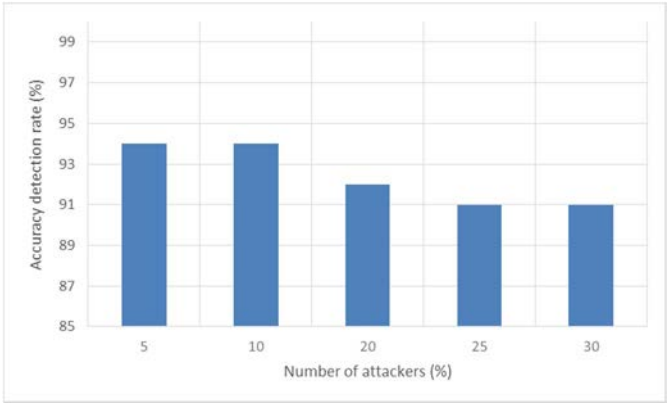

Fig.3. Accuracy detection

C. Computation Overhead Analysis

The scheme is implemented in TinyOS using TinyECC Library [15] with the standard curve SECP160R1. For HKDF and signatures, we use the HMAC provided in the library that considers SHA-1 as hash function. We consider MicaZ sensor platform for our results. The curve point generation takes 1.31s. For encryption, we consider 20 bytes the size of the modulus. Each device needs 0.07s to encrypt and sign the data and the homomorphic aggregation performed by aggregator nodes takes about 0.002s. The IDSs that are activated at $node_k$ and CH_j, rely on a lightweight detection technique to identify the misbehavior of resource-constrained devices. This technique is based on a rule based detection approach. So, the computation overhead at devices is negligible. Our cyber security game is launched at a powerful node, BS, since it requires a certain computation overhead to detect with a high accuracy the malicious attacks at $node_k$ and CH_j levels.

D. Communication Overhead Analysis

First of all, the complexity of communication is *O(1)* for CH and non-CH nodes i.e. for each sensor device deployed. The total communication cost i.e. the number of transmitted packets of the scheme is *N*, where *N* is the number of participating nodes in the whole network. In aggregation phase, each device sends a packet of 30 bytes (20 bytes ciphertext and 10 bytes for truncated HMAC output). To give a sense to this, simulation are conducted using TOSSIM-CC2420 [16], by taking into account the energy consumption model for MicaZ presented in [17]. We compare with RCDA [7]. The reason to choose this work for comparison is that it provides an end-to-end security and proposes a mechanism to identify the malicious node. 100 sensor devices with 8 clusters are considered in the simulation, while a simple TDMA-based clustering algorithm to transmit (every 30s) data to the BS. Each device encrypts 12 bits $D_{ij} = DR_{ij} \,||\, m_{ij}$. DR_{ij} represents the 2 bits node's reports about its k neighbors (assume k=4), So, 8 bits for the reports and 4 bits are sufficient for m_{ij}. Note that a larger modulus can be considered with a larger D_{ij}. In our simulation, the aggregator is responsible of 11 to 12 nodes, so, the 20 bytes modulus is sufficient to homomorphically aggregate the corresponding D_i.

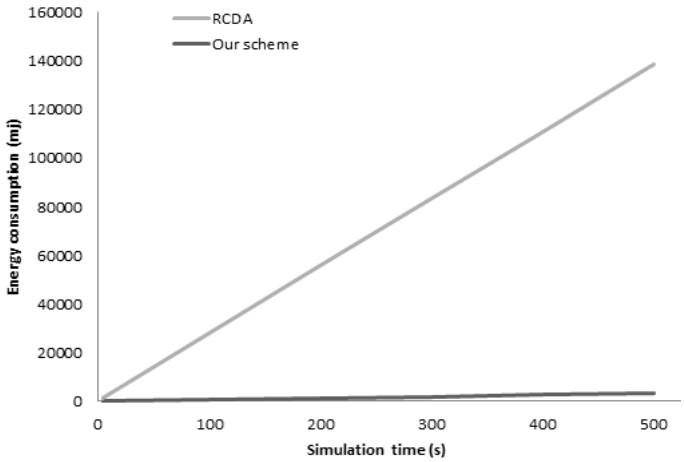

Fig. 4. Energy consumption

In Fig. 4, we show that our scheme provides a great reduction of energy consumption compared with RCDA. This gain can be explained by the fact that much lesser computation overhead is incurred in our scheme due to the use of symmetric primitives. Consequently, for the same level of security provided, the network lifetime is hugely improved. Also, in the case of attacker's presence, RCDA notifies the corresponding aggregators in order to detect the malicious node. So, an additional overhead and time are incurred. The IDS proposed in this paper allows the BS to directly detect the malicious node after the second verification even in the presence of A_i.

V. Related work

There is an extensive research on secure data aggregation in WSNs. Existing works are designed for different security requirements. Przydatek et al. [4] focus on data integrity by using an *interactive proof* session in which the base station interacts with aggregator node in order to verify the correctness of the aggregation result. This scheme focuses on aggregation of plain data and *the stealthy attack*, an attack where the adversary aims to force the base station to accept a deceiving aggregation result. In [5], the authors propose a secure scheme in which the data confidentiality is provided end-to-end. Through efficient hop-by-hop verification, their scheme allows early detection of attacks such as false data injection and impersonation, thus reducing the need to rely entirely on sink node for verification. However, attacks caused by internal adversaries such as selective forwarding cannot be detected by their scheme. In [6-7], the authors focus on end-to-end integrity and end-to-end confidentiality. In [6], the authors propose an efficient scheme in which they employ stateful public key encryption in order to provide an end-to-end security. The solution does not impose any bound on the aggregation function's nature (Maximum, Minimum, Average, etc.). Even if the verification is performed at BS, their algorithm cannot detect an internal attack. In [7], the authors use El Gamal encryption and aggregate signatures based on bilinear maps. Each packet includes an encryption and a signature on data. The scheme incurs an important computation and communication overhead due to the use of identity-based signature. Furthermore, the scheme only prevents external attacks.

The IDS technique has the ability to detect the internal attack, while the encryption techniques prevent an external threat to launch cyber-attacks. In [14] and [18-19], the authors combine between these two techniques to leverage the advantage of each one of them. Specifically, in [14] the authors use the encryption and authentication protocols to prevent the unauthorized sensor node to overhear the information exchanged between its neighbors nodes. To detect the Denial of Service (DoS) such as grey hole attacks, the IDS based on a signature detection technique is used to identify the internal attackers. In [18], the authors integrate an authentication and IDS techniques to secure the cluster based network. The purpose of this work is to reduce the sizes of session keys and intrusion data exchanged between the IDS nodes, while ensuring a tradeoff between the attack detection and energy consumption. According to the simulation results of these works [14][18], the authors prove that the internal and external cyber threats are efficiently detected. However launching

simultaneously these two techniques can degrade the performance of the network since a high computation overhead could be generated. Therefore, to overcome this issue, in [19] a lightweight security mechanism is proposed for sensors network, where the cryptography technique is executed only when an external node is suspected to launch an attack. Despite, a low communication overhead that this framework requires to protect the sensors, an important number of external attacks are not detected.

The optimal activation of IDS' monitoring could be useful in dealing the security issues for resource-constrained devices. Since the goal of this security approach is to ensure a certain tradeoff between internal attacks detection and low computation (and communication) overheads. The works [20] and [21] were pioneer that dealt this security tradeoff for IoT network. Specifically in [20], the activation of a monitoring and detection process is done locally, i.e., there is no cooperation between the neighbors IDSs and base station to determine which IDS should be activated. This optimal activation is done with a help of Ising model, which relies on graph theory. In [21] an optimal activation of anomaly detection technique is proposed. The goal of this work is to use a heavy algorithm such as neuronal network for anomaly detection in low resource nodes. The activation of anomaly detection technique is done when cyber-attacks exhibits a new misbehavior. This is achieved with a help of non-cooperative game. The weakness of these works is that the authors do not take into account the fact that the attackers can launch an external threat to take control of a legitimate sensor node.

VI. CONCLUSION

In this paper, a novel security protocol of data aggregation is proposed for resource-constrained devices. The scheme is a hybridization of an intrusion detection system and a cryptographic solution. Indeed, data needs to be verified twice by the BS to be validated. To the best of our knowledge, it is the first work which employs IDS with homomorphic encryption for constrained devices. The major advantage is the time-efficient detection. In fact, results and analysis presented in this paper show, on one hand, the efficiency of the scheme, and on the other hand, the very high level of security, where after verification the malicious device is directly and instantly identified even in the case of internal adversaries. Furthermore, the proposed scheme allows the sink node to calculate any aggregation function on sensors data. This property is very important since it is needed to serve a wide range of resource-constrained applications. In future work, we aim to extend our work to support nodes mobility.

REFERENCES

[1] Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, 17(4), 2347-2376.

[2] Keoh, Sye Loong, Sandeep S. Kumar, and Hannes Tschofenig. "Securing the internet of things: A standardization perspective." *IEEE Internet of Things Journal* 1.3 (2014): 265-275.

[3] Peter, Steffen, Dirk Westhoff, and Claude Castelluccia. "A survey on the encryption of convergecast traffic with in-network processing." *IEEE Transactions on Dependable and Secure Computing* 7.1 (2010): 20-34.

[4] Przydatek, B., Song, D., & Perrig, A. SIA: Secure information aggregation in sensor networks. Proceedings of the 1st international conference on Embedded networked sensor systems. November 2003. ACM, L.A, California, 2003; 255- 265.

[5] Merad Boudia OR, Senouci SM, Feham M. Secure and efficient verification for data aggregation in wireless sensor networks. *Int J Network Mgmt*. 2017;e2000.

[6] Merad Boudia OR, Senouci SM, Feham M. A novel secure aggregation scheme for wireless sensor networks using stateful public key cryptography. *Ad Hoc Networks*, *32*, 98-113. 2015

[7] Chen, Chien-Ming, et al. "RCDA: recoverable concealed data aggregation for data integrity in wireless sensor networks." *IEEE Transactions on parallel and distributed systems* 23.4 (2012): 727-734.

[8] Butun, Ismail, Salvatore D. Morgera, and Ravi Sankar. "A survey of intrusion detection systems in wireless sensor networks." *IEEE Communications Surveys & Tutorials* 16.1 (2014): 266-282.

[9] Kott, Alexander, Ananthram Swami, and Bruce J. West. "The Internet of Battle Things." *Computer* 49.12 (2016): 70-75.

[10] Fontaine and Galand. A Survey of Homomorphic Encryption for Nonspecialists. *EURASIP Journal on Information Security*, volume 2007, 1-15.

[11] C. Gentry, Fully homomorphic encryption using ideal lattices, *Symposium on the Theory of Computing (STOC)*, 2009, pp. 169-178.

[12] Castelluccia, C., Chan, A. C., Mykletun, E., & Tsudik, G. Efficient and provably secure aggregation of encrypted data in wireless sensor networks. *ACM TOSN*, 5(3), 20, 2009.

[13] Q. Wu, S. Shiva, S. Roy, C. Ellis, V. Datla, "On modeling and simulationof game theory-based defense mechanisms against DoS and DDoSattacks", Proceedings of *the 2010 Spring Simulation Multiconference*, Orlando, Florida, USA, 2010.

[14] S. Raza, L. Wallgren, T. Voigt, "SVELTE: Real-time intrusion detection in the Internet of Things", *Ad Hoc Networks*, Vol 11, Issue 8, 2013, pp. 2661-2674.

[15] TinyECC (Version 2.0) http://discovery.csc.ncsu.edu/software/TinyECC/

[16] P. Levis, N. Lee, M. Welsh, and D. Culler. TOSSIM: Accurate and scalable simulation of entire tinyos applications. In Proceedings of the First ACM Conference on Embedded Networked Sensor Systems (SenSys 2003). ACM Press, November 2003.

[17] G. De Meulenaer, D. Gosset, O.X. Standaert, O. Pereira, On the energy cost of communication and cryptography in wireless sensor networks, *Proceeding of IEEE WIMOB*, October 2008, Avignon, pp. 580–585.

[18] W.T. Su, K.M. Chang, Y.H. Kuo, "eHIP: An energy-efficient hybrid intrusion prohibition system for cluster-based wireless sensor networks", *Computer Networks* 51(4), 2007, pp. 1151-1168.

[19] H.Sedjelmaci, S.M. Senouci, "A lightweight hybrid security framework for wireless sensor networks", IEEE ICC, Sydney, Australia, 2014, pp. 3636-3641.

[20] Y. Ponomarchuk, S.W. Seo, "Optimal activation of intrusion detection agents for wireless sensor networks", *Fourth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*, Florence, Italy, 2010

[21] H. Sedjelmaci, SM. Senouci, T. Taleb, "An accurate security game for low-resource IoT devices", *IEEE Transactions on Vehicular Technology*, 2017