A Novel Algorithm for Securing Data Aggregation in Wireless Sensor Networks

Haythem Hayouni, PhD Student, Univ Carthage, Sup'Com Security Lab, Tunis, Tunisia

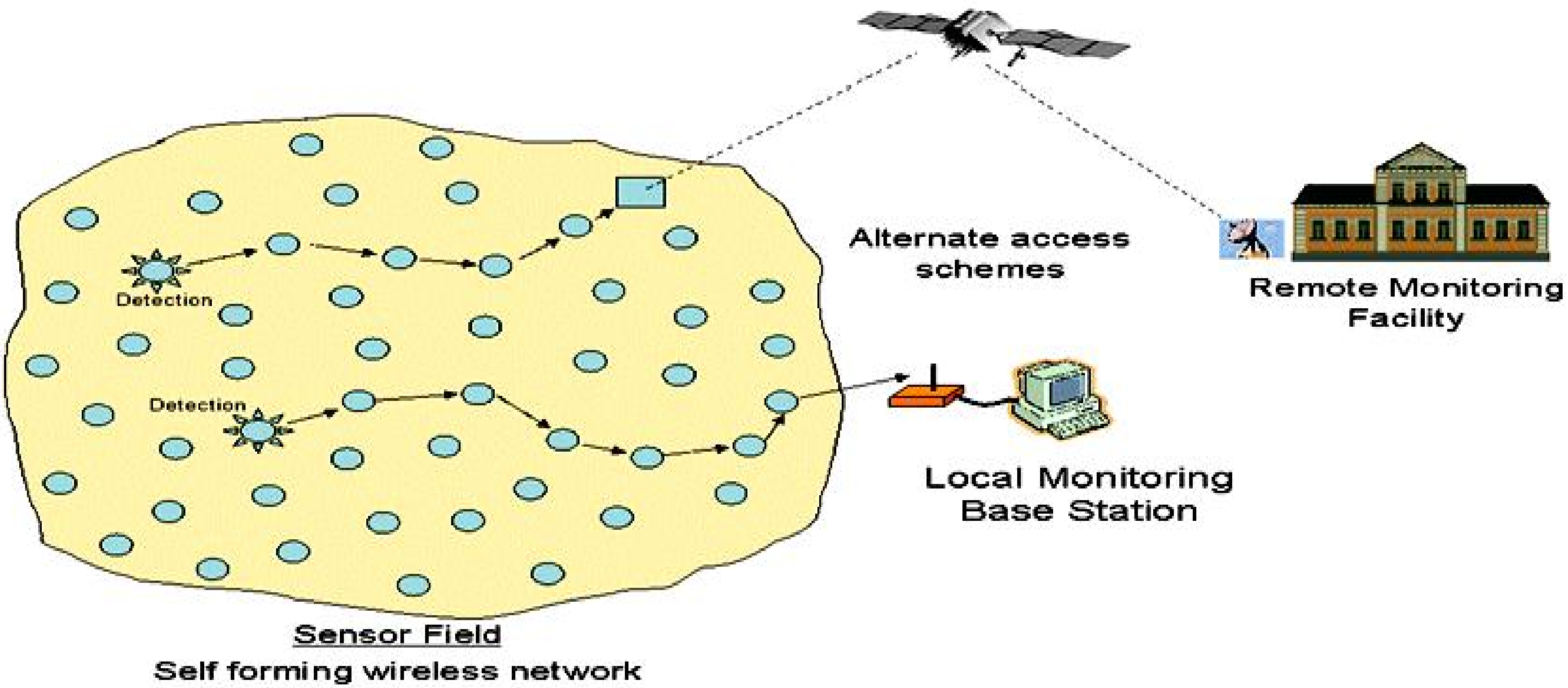
Mohamed Hamdi, Univ Carthage, Sup'Com, Security Lab, Tunis, Tunisia

AD HOC NOW 2016 JULY 04-06

SUP'COM
Higher School of Communication of Tunis

Wireless Sensor Network (WSN)

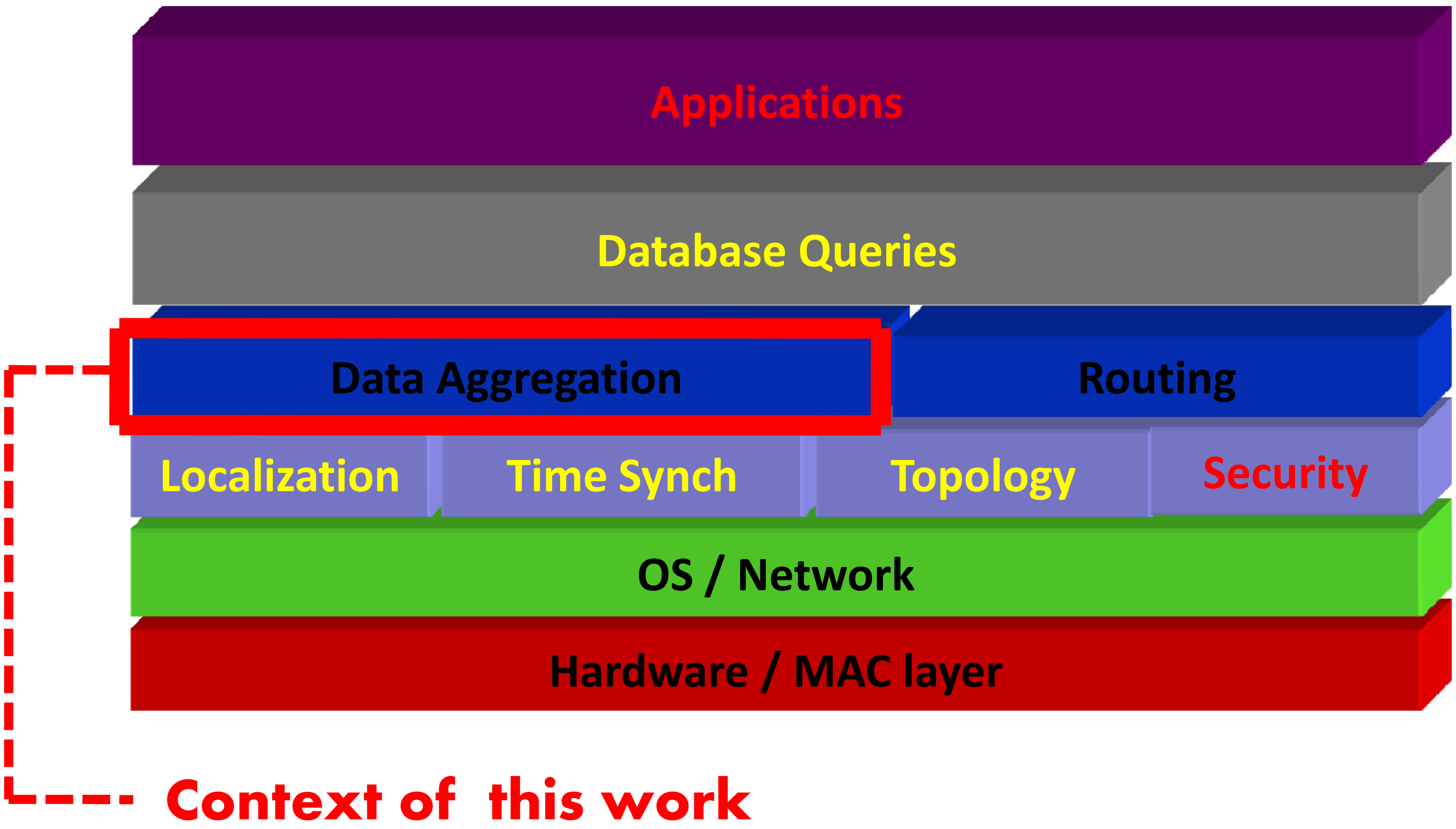
- Highly distributed network which consists of many low-cost sensor nodes and a base station (or sink) that gathers the observed data for processing.



Characteristics of Sensor Networks

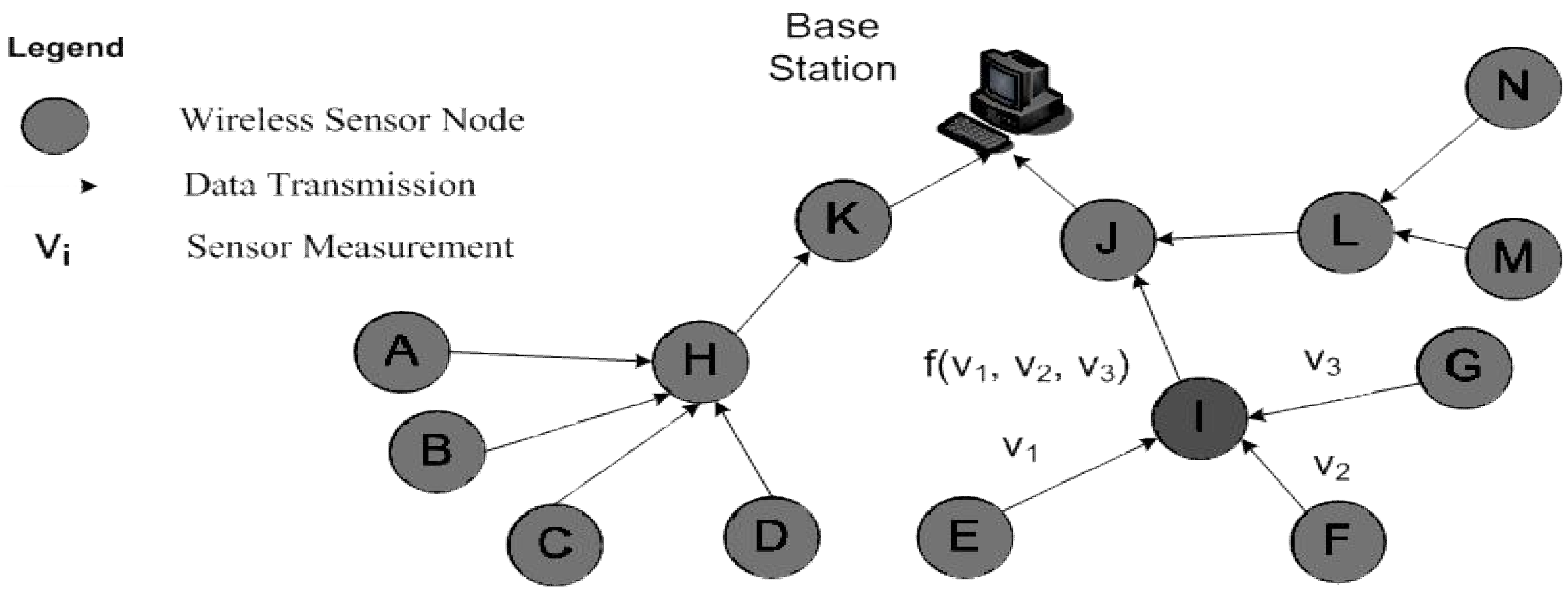
- Low power and limited energy supply
- Simple CPU (low speed, small memory)
- Low network speed, unreliable data link, small transmission range
- Large deployment and dynamic replacement
- No infrastructure and self & dynamic organization

Layered Structure of Sensor Networks

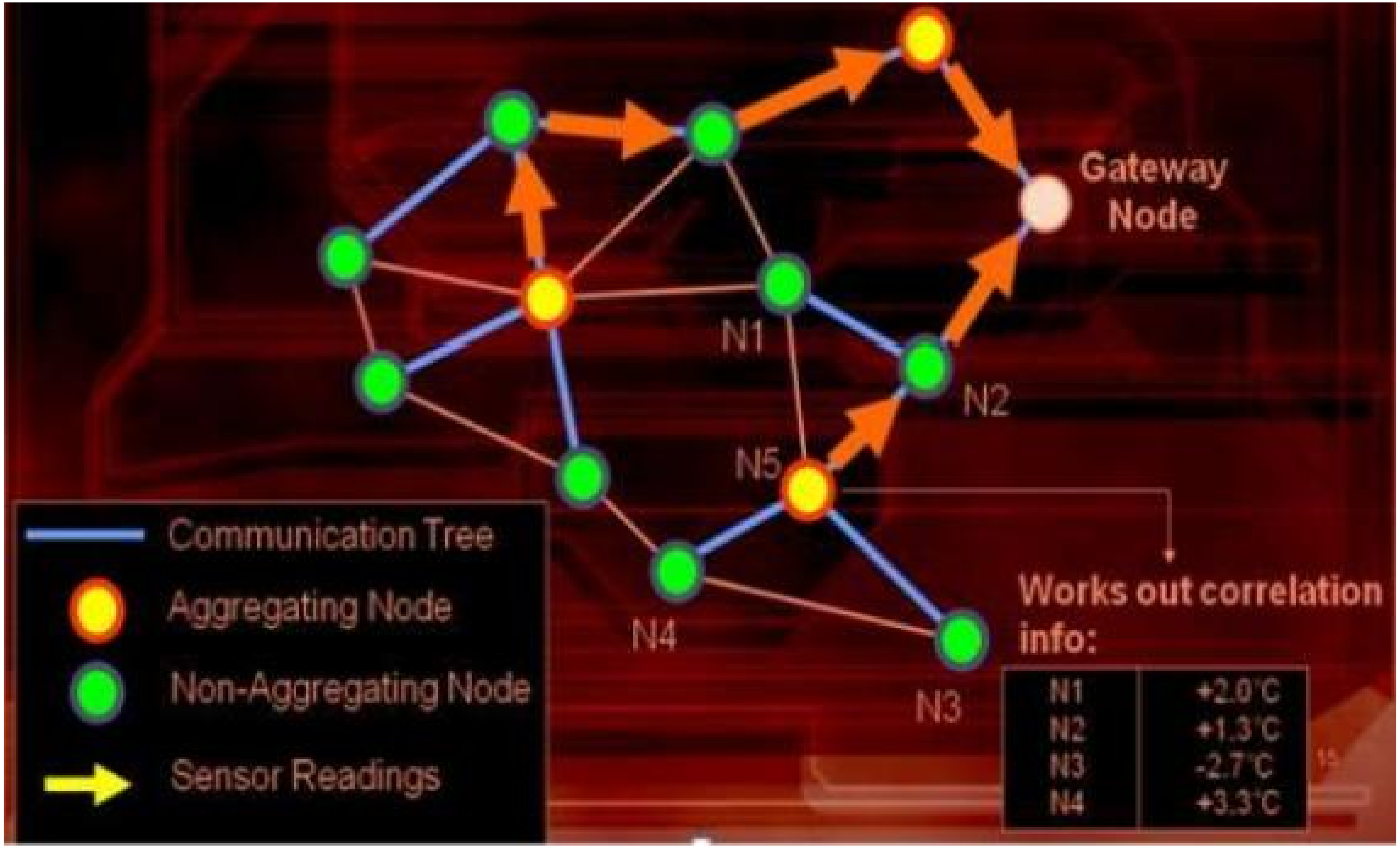


What is Aggregation?

- Aims to reduce power consumption in WSN.
- Aims to combine and summarize the data packets of several nodes so that amount of data transmission is reduced.
- Data aggregation reduces the number of transmissions thereby improving the bandwidth and energy utilization in WSN.

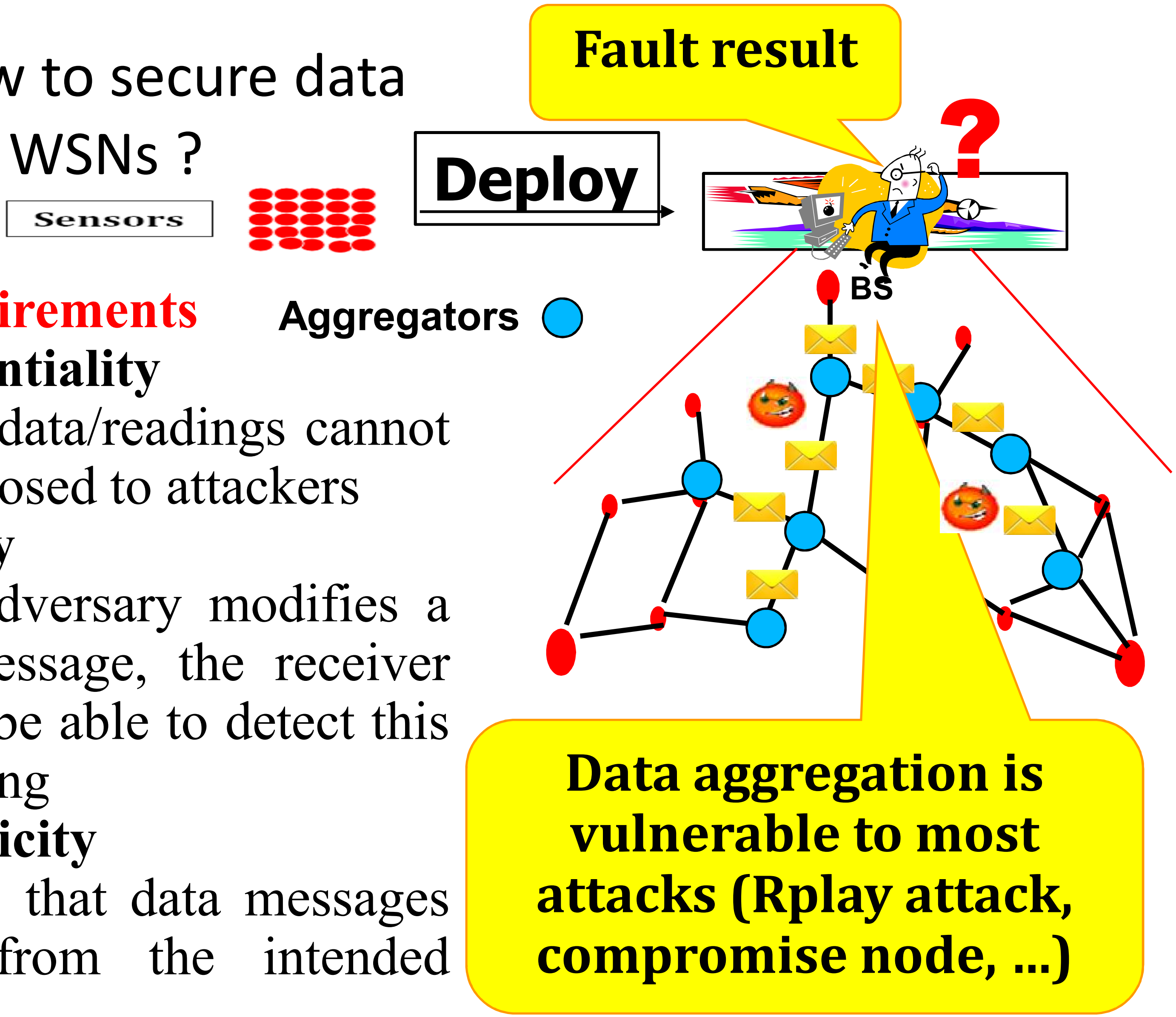


What is Aggregation?



Data aggregation in WSN: Problem

Problem: how to secure data aggregation in WSNs ?



Security Requirements

- **Confidentiality**
  Sensor data/readings cannot be disclosed to attackers
- **Integrity**
  If an adversary modifies a data message, the receiver should be able to detect this tampering
- **Authenticity**
  Ensures that data messages come from the intended sender

Secure data aggregation in WSN: Related work

iPDA: ***He, W., Nguyen, H., Liu, X., Nahrstedt, N., Abdelzaher, T.: iPDA: an integrityprotecting private data aggregation scheme for wireless sensor networks. (2008)***

Authors present a reliable data aggregation forwarding scheme for WSNs, based on cluster formation. However, the integrity of personal data is not provided since the attacker can easily modify data between nodes and the aggregator.

EIRDA: ***Engouang, T.D., Yun, L.: Aggregate over multi-hop homomorphic encrypted data in wireless sensor networks. (2013)***

Authors propose an efficient aggregation of encrypted data, based on homomorphic hash while providing security of aggregated data.

AMAC : ***Parmar, K., Jinwala, D.C.: Aggregate MAC based authentication for secure data aggregation in wireless sensor networks. (2014)***

Authors propose an efficient data aggregation solution based Message Authentication Code (MAC) to provide authentication. This solution use Aggregate MAC (AMAC) to reduce the transmission cost incurred by MAC.

Contributions

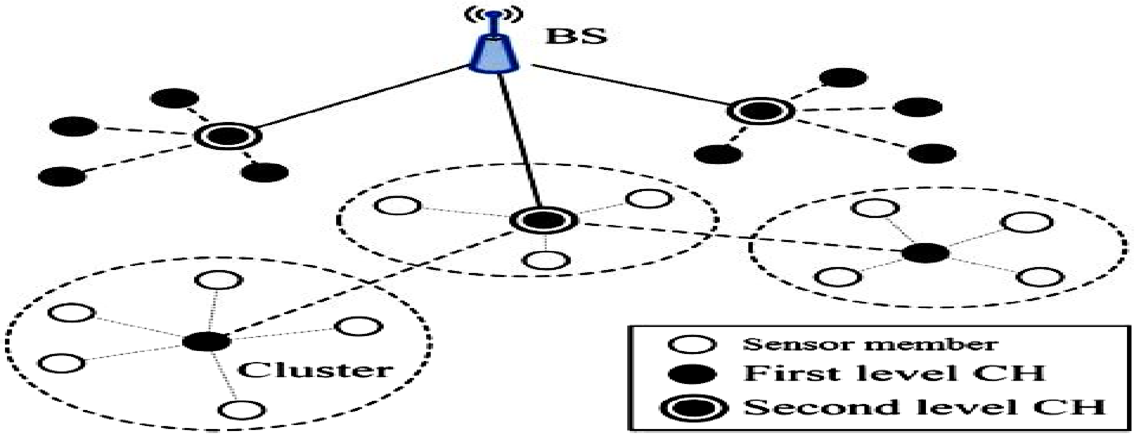
- **Sensor nodes deployment**
  - **Tree formatting phase**
  - **Key generation**

- **Propose an algorithm to secure data aggregation named E-SHE)**
  - **Data aggregation based on homomorphic primitives.**
  - **Homomorphic MACs**
  - **Verification process**
  - **Preserve data privacy and check data integrity**

- **Our results** : less communication and computation overheads, high data transmission efficiency, and less energy to prolong network lifetime.
  - **Verification by simulation**

Background and Assumptions

- **System Architecture**



- **Attack Model**

  To describe the security of our algorithm, we define four attacks:
  - Unauthorized aggregation
  - Malleability
  - Node compromise:
  - Replay attack

Background and Assumptions

- **Homomorphic Encryption**

HE allows direct addition and multiplication of ciphertexts. Let m_1 and m_2 be two plaintexts and let $\otimes, \times$ be the homomorphic operations on the ciphertexts and plaintexts, respectively; we have

$$\text{Enc}(m_1) \otimes \text{Enc}(m_2) = \text{Enc}(m_1 \times m_2),$$ where $\text{Enc}(m)$ represents the ciphertext of m.

Contribution: Secure Data Aggregation Based on Homomorphic Primitives

E-SHE contains four process: **Key Generation**, **Sign-Encrypt, Aggregate** and **Verify**.

Key Generation

Given $\alpha \in Z$, the tuple (q_1, q_2, q_3, E) is generated, where q_1, q_2, q_3 are large primes, E is the set of elliptic curve points. Then, take three points (g_1, g_2, g_3) randomly from E. Compute points $P = q_2 q_3 * g_1$, $Q = q_1 q_3 * g_2$, and $H = q_1 q_2 * g_3$. Such that the order of P, Q and H is q_1, q_2, and q_3 respectively. The generated encryption key Y is:

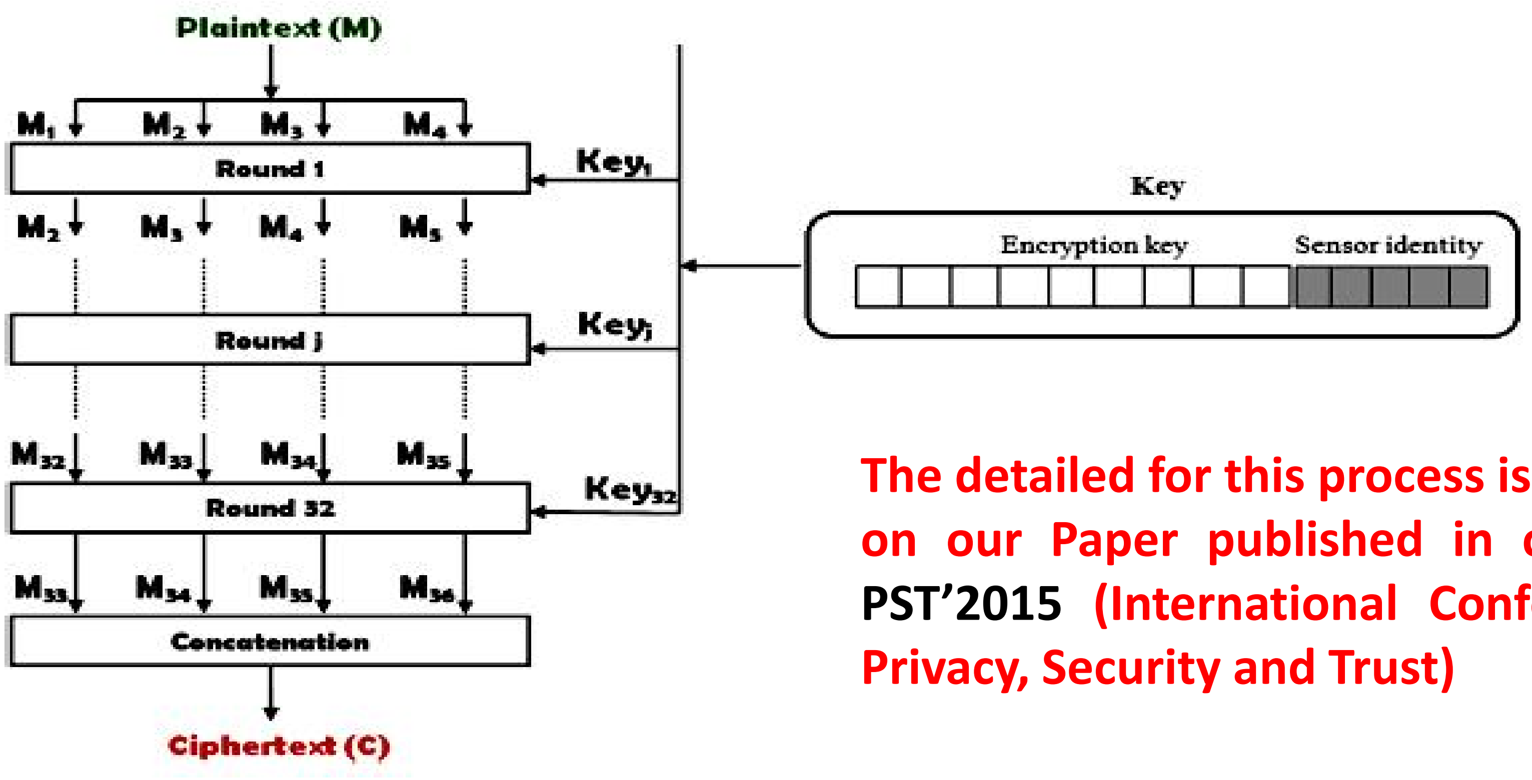
$$Y = (E, P, Q, H)$$

After, the **sensor identity** is added to the key **Y** to form a final key **Key**.

Sign-Encrypt

Encryption step for our algorithm is based on 32 rounds unbalanced Feistel network . The generated **ciphertext** is defined as follow:

$$msg = Concat(M_{33}, M_{34}, M_{35}, M_{36})$$



The detailed for this process is presented on our Paper published in conference PST'2015 (International Conference on Privacy, Security and Trust)

After, the sensor computes the signature t_i of (Key_i, id_i, msg_i). Our contribution presented as follow:

To formally our schemes, message msg is formed as s segments of l bits. Let $r = 2^l$, then the message space is F_r^s. All contributors and verifiers share one global MAC key that consists of (key_1, key_2). Let K_1 and K_2 denote the key spaces of key_1 and key_2 respectively, and $\mathcal{I}$ denote the space of node identities. Two pseudo random functions are required: $Rd_1 : K_1 \rightarrow F_r^s$ and Rd_2: $(K_2 \times \mathcal{I}) \rightarrow F_r$.

t_i is computed as follow:

$$a = Rd_1(key_1)$$

$$b_i = Rd_2(key_2, id_i)$$

$$t_i = a.msg_i + b_i$$

At the end, the sensor node i sends the couple $(E(msg_i), t_i)$ to aggregator.

Aggregate

The aggregator aggregates $((msg_1, t_1, w_1), ..., (m_j, t_j, w_j))$ as follow:

(i): Aggregated Ciphertext

$$msg' = \sum_{i=1}^{j} w_i.msg_i \in F_r^s$$

(ii): Aggregated MAC

$$t' = \sum_{i=1}^{j} w_i.t_i \in F_r$$

At the end, the aggregator sends the aggregated result (msg', t') to the base station BS.

Verify

The Base station (BS) decrypts the aggregate result using its private key, where it needs to inverse the mapping from the point on the elliptic curve to the aggregate result.

While the BS receives (msg'_i, t'_i) from $Aggregator_i$, it can recover and verify each sensing data as follows:

- BS obtains the m_i by decrypting msg'_i.
- BS verify (Key$_i$, id_i, m_i, t'_i):

$$a = Rd_1(key_1) \in F_r^s$$

$$b = \sum_{i=1}^{j}(w_i.Rd_2(key_2, id_i)) \in F_r$$

If $a.m_i + b = t'_i$ then the result is accepted, else the result is refused.

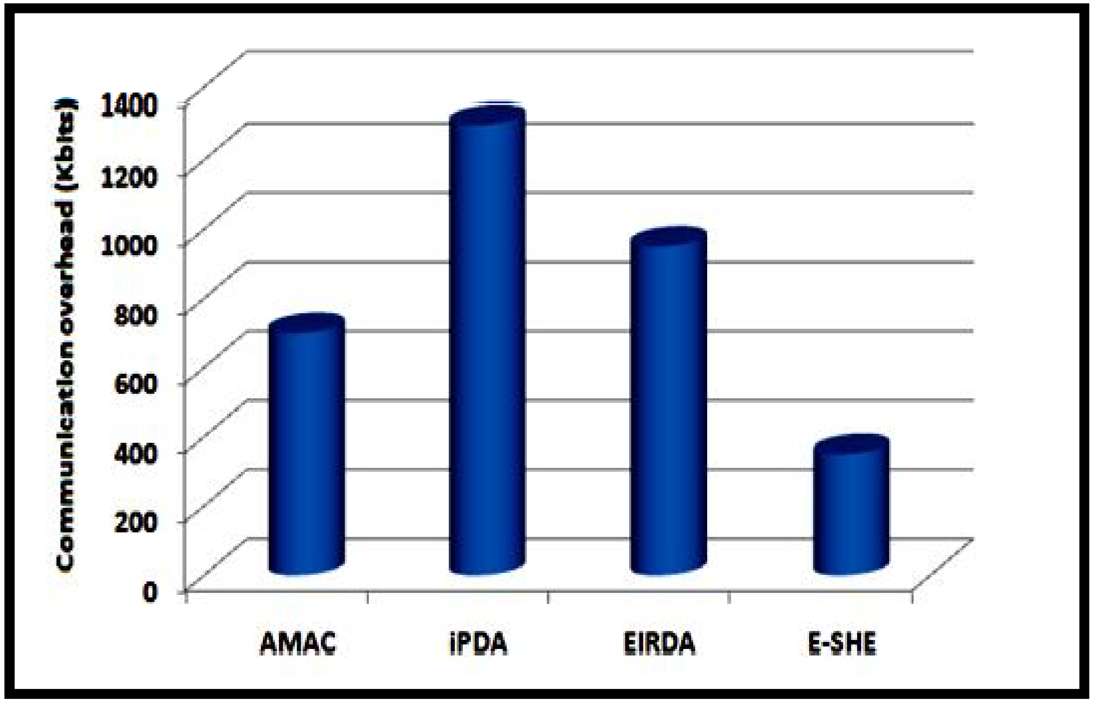
E-SHE: Simulation and performance evaluation

Simulation parameters.

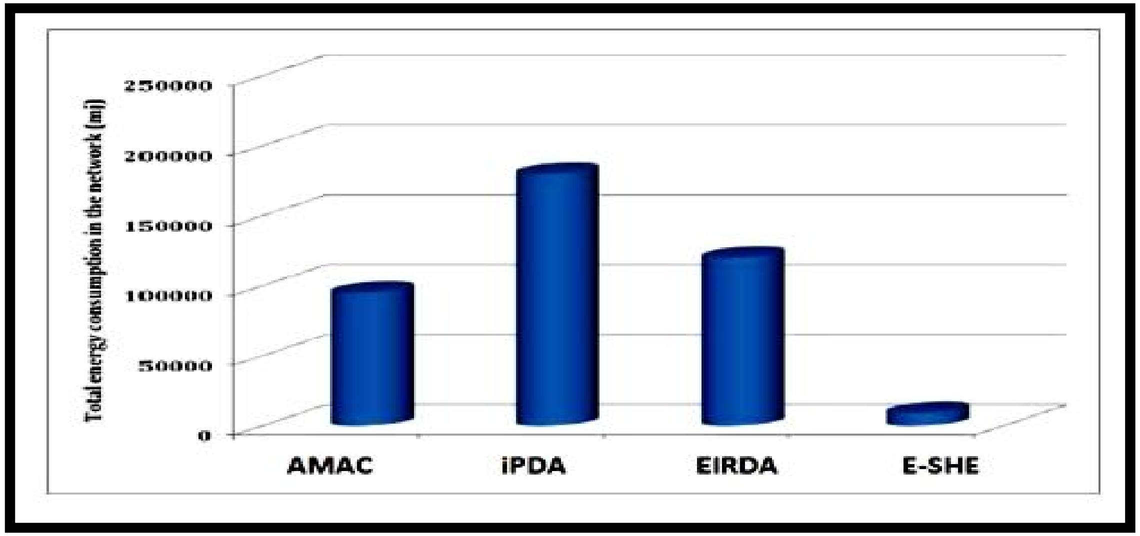
Parameter	Value
Number of sensor nodes	20–300
Transmission range	30 m
Area size	400 m × 400 m
Transmit power	0.720 mw
Receiving power	0.405 mw
Initial energy	6.3 J
Packet size	45
Noise floor	−105 dB
Simulation time	500 s

-- We use TinyOS 2.0 simulator (TOSSIM)

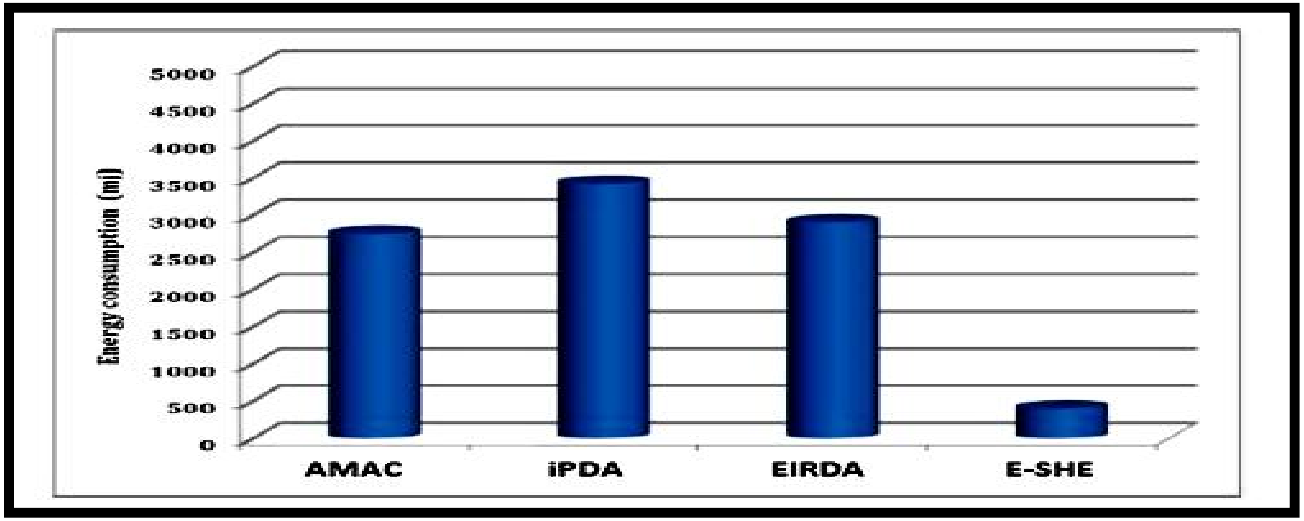
Communication overhead



Total energy consumption in the network



Energy consumption by sensor nodes



Conclusion and perspectives

- Aggregation can provide many benefits.
- Many different protocols exist with different types of goals in mind.
- We propose an efficient algorithm for securing data aggregation based on homomorphic MACs.
- Our algorithm requires less communication and computation overheads than previously known methods and can effectively preserve data privacy, check data integrity, and consuming less energy to prolong network lifetime.
- At present, our algorithm is applied to the secure aggregation scheme for SUM queries only. Further research will be to design a secure data aggregation scheme which can cover a wide range of exact aggregate queries

Thanks