Secure Lossless Aggregation for Smart Grid M2M Networks

A. Bartoli*, J. Hernández-Serrano†, M. Soriano*†, M. Dohler*, A. Kountouris‡ and D. Barthel‡
*Centre Tecnològic de Telecomunicacions de Catalunya (CTTC), Spain
†Universitat Politècnica de Catalunya (UPC), Spain
‡Orange, France Telecom, France

***Abstract*—Whilst security is generally perceived as an important constituent of communication systems, this paper offers a viable security-communication-tradeoff particularly tailored to Advanced Metering Infrastructures (AMIs) in Smart Grid systems. These systems, often composed of embedded nodes with highly constrained resources, require e.g. metering data to be delivered efficiently whilst neither jeopardizing communication nor security. Data aggregation is a natural choice in such settings, where the challenge is to facilitate per-hop as well as end-to-end security. The prime contribution of this paper is to propose a secure aggregation protocol that meets the requirements of Smart Grids, and to analyze its efficiency considering various system configurations as well as the impact of the wireless channel through packet error rates. Relying on analysis and corroborative simulations, unprecedented design guidelines are derived which determine the operational point beyond which aggregation is useful as well quantifying the superiority of our protocol w.r.t. non-aggregated solutions.**

I. Introduction

Ever since Obama's "National Broadband Plan" [1], Smart Grids have moved into the limelight. Said grid is connecting points of inference (e.g. water meter) and control (e.g. valve) with a smart decision engine (e.g. the utility's control center) by means of Machine-to-Machine (M2M) communication mechanisms.

At national level, advantages are that the efficiency and effectiveness of the system are significantly increased; thus, dependency on foreign natural resources, waste and usage at large is diminished. Advantages for resource suppliers, such as utility companies, are the ability of (near) real-time monitoring of the grid infrastructure; thus, faults and outages can be detected and attended to with minimum delay, energy can be traded at different tariffs, etc. The end-user profits as an optimized and (nearly) instantaneous bill can be delivered, failures can be detected and handled remotely by the utility or the user him/herself, appliances are used when energy is cheapest, etc.

As outlined in [1], wireless communication systems play an integral role in accomplishing this vision. This is realized by the typically used communication architecture, where embedded radios are installed at each sensor (meter) and actuator (valve); these communicate wirelessly between each other in a multihop fashion over a (typically) tree-like topology [2]; until a gateway is reached which could be a DSL line or a cellular interface.

Whilst these required wireless constituents are becoming increasingly ubiquitous, they suffer from some inherent shortcomings. Notably, the nodes are often short of resources (e.g. power supply, memory, processing power); the spectrum they use to communicate is considered to be scarce; the wireless channel itself is a source of uncertainty which leads to packet errors and thus re-transmissions; the wireless channel is broadcast by nature and thus prone to compromise in security.

The above advocates for a paradigm shift in designing wireless communication systems tailored to the needs of AMI in Smart Grids. A first step is to use data aggregation at each multihop node, which aggregates the received packets from its leaf nodes prior to forwarding the aggregated packet to its parent node. Given that one of the AMI's core requirement is to be able to obtain an exact reading from each node and also to be able to uniquely associate a node to the data, lossless aggregation must be used. Among the very few lossless techniques available, packet concatenation is a suitable solution which yields ease of use at notable performance gains.

In order to save energy in said data collection networks, aggregator nodes instead of retransmitting the raw received data forward the aggregated data by combining the packets (saving headers) or even removing redundant information. The use of aggregation avoids unnecessary energy consumption on multihop networks but introduces, among others, a new major risk: the aggregated packet contains big portions of the collected data and such data could be easily eavesdropped by a passive attacker or even forged or deleted by an active one. Consequently, many protocols have appeared in the literature to secure aggregation on multihop networks. These security protocols are commonly classified according to whether they are end-to-end or hop-by-hop secure aggregation schemes.

In end-to-end encryption schemes [3]–[5], collected data is secured at the source and the keys to decrypt and check this data are only shared between the originator (mainly a metering node) and the base station or gateway. As a result, the challenge is how the intermediate nodes do aggregation on data that they cannot decrypt. Aggregation on such solutions can be as simple as concatenation of encrypted data (saving packet headers) or more sophisticated provision of secure aggregation by using additive privacy homomorphism protocols [4]. However, with end-to-end encryption, the link layer is not protected at all and thus being accessible for an attacker. As a naive example, one could simply drain the radio

978-1-4244-6511-8/10/$26.00 ©2010 IEEE

by constantly sending packets which can only be identified as false once the entire reception and decryption has taken place.

Hop-by-hop aggregation protocols, such as [6]–[9], provide more efficient aggregation operations and protect the link layer and above. Nevertheless, since sensed data are revealed for the sake of aggregation at the aggregator nodes, hop-by-hop aggregation protocols are by design weaker in terms of confidentiality than end-to-end aggregation protocols. Combination of both protocols can be done under certain conditions and some proposals, such as [10], [11], have already tackled this in parts.

Summarizing, aggregating packets has a profound impact onto various aspects of the wireless communication system. First, per-hop and end-to-end security mechanisms need to be re-designed. Second, next-hop communication and security overhead is saved when only one longer instead of several shorter packets is transmitted. Third, the packet error rate (PER) and thus the average number of retransmission of a longer packet are generally larger than of a shorter packet. To the best of the authors' knowledge, a joint tradeoff of above security and communication paradigms has not been performed to date.

The aim of this paper is hence to quantify the performance gain of secure, lossless packet aggregation operating over a lossy channel. To this end, the paper is structured as follows. In Section II we discuss some background ideas of the scenario and then we focus on the security features of our protocol: end-to-end security and hop-by-hop security. Next we present the secure lossless aggregation process and the corresponding algorithm. In Section III, in order to justify the benefits of our studies, we assess the overhead utilized and the energy consumption with and without lossy channel of our protocol compared to traditional solutions or with solution without security requirements. Then, the protocol is analyzed clearly showing the bandwidth and energy gains, in addition to establishing a high level of security.

II. Lossless Aggregation Protocol

In this section we present a protocol for smart grid M2M networks that secures communications between a set of collector nodes or meters and a base station or gateway in an efficient and secure manner. The protocol is designed for a typical scenario depicted in Figure 1 where some metering nodes collect data which is reported to a gateway (base station) through a multihop network. As a result, there are three types of nodes and a base station: 1) metering nodes, that actually infer the data; 2) aggregator nodes, that collect data sensed by a set of metering nodes; and 3) routers that provide the necessary infrastructure to facilitate communication between involved nodes (notice that aggregator nodes are also routers); and 4) the base station itself. The aim of the protocol is to provide both end-to-end (between meters and gateway) and hop-by-hop (within every link) security whilst minimizing the traffic in the network.

End-to-end security is achieved by means of a shared secret between every meter and the gateway; hop-by-hop security is done at MAC/PHY layers by means of pairwise keys between every network node and its one-hop neighbors; lossless aggregation maximizes the use of links and thus minimizes network traffic. As we will show later in Section III, the proposed protocol not only avoids an extra cost for security but also reduces the overall cost of the process of sending the data. This is due to aggregation savings making up for or even exceeding the computational cost of the security operations.

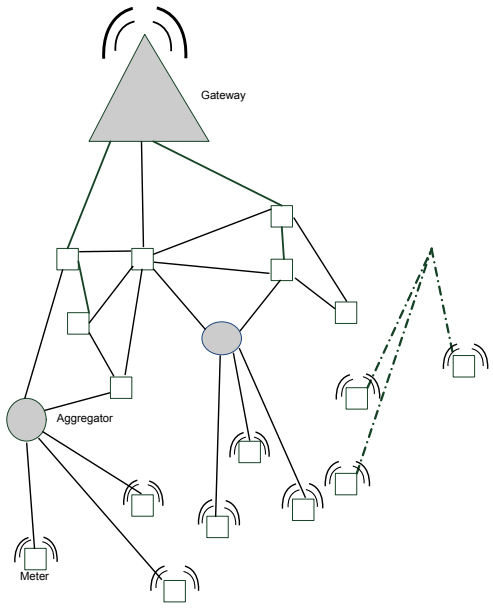

Fig. 1. Abstraction of the Smart Grid application scenario.

Subsequently, we outline the functioning of end-to-end security, hop-by-hop security and lossless aggregation. Thereupon, in Section III, we detail its application to IEEE 802.15.4 networks.

A. End-to-End Security

The aim of end-to-end security is to protect the data from unauthorized eavesdropping (confidentiality); to allow the destination to check the integrity of the received data and its freshness; and to unequivocally identify the source of such data (authentication). End-to-end security is achieved here as follows.

The metering node creates a packet with the sensed data as shown in Figure 2. The headers include: the source of data (addressing field), destination (gateway), a timestamp, a key identifier, a security control and the data length; the data is encrypted with the key shared with the gateway; and a *message integrity code* (MIC) is appended. Consequently, end-to-end security due to CIA (confidentiality, integrity and authentication) and freshness (because of the timestamp) are provided between the meter and the gateway.

Compared to non-secure protocols, the use of end-to-end security introduces some overhead (see OH_N in the implementation example in Figure 4); however, on the other hand, it allows the gateway to securely identify the source of the

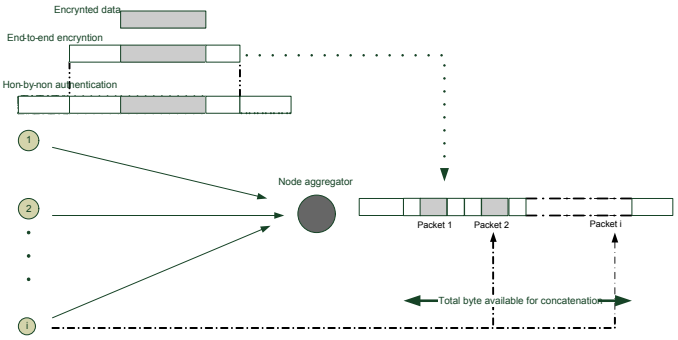

Fig. 2. The secure lossless aggregation process.

data and to detect any modification of this data along its path to the gateway.

In a typical implementation, the overhead OH_N related to the application layer in order to achieve end-to-end security is at least:

- An identifier of the key/s used for encrypting the data and creating the MIC. This identifier allows the gateway to find or derivate the keys for checking the MIC and decrypting the data. Once again, 1 byte should be enough in most cases.
- A security control that contains the security level and the key identifier mode.
- [1]A timestamp in order to guarantee freshness of collected data. Its length will be related to the amount of sent/received packets per time interval. Usually the timestamp is just related to the frame counter, the key counter or both.
- [1]The length of the encrypted data. The gateway will need this length in order to know how many bits to decrypt after this header. Typically the length is part of the frame control field.
- A MIC of the packet header and data. The MIC typically is 32, 64 or 128 bits long.

B. Hop-by-Hop Security

End-to-end security is checked at the final destination; however, before reaching the gateway, the packets must go through one or more wireless links that are by nature exposed to attackers. As a result, if no security is provided in order to restrict the access to the media, only the destination point will be able to detect altered, dropped or fake packets. This fact exposes the network to exhaustion attacks since those packets will waste precious energy at the intermediate nodes (routers). Consequently, hop-by-hop integrity, authentication and freshness should also be provided at PHY/MAC layers.

From above reasoning, the protocol requires the use of timestamps and MICs also at PHY/MAC layers. Then, compared to non-secure protocols, the use of hop-by-hop security introduces at least the following overhead:

[1]It will also be usually present in any non-secure scenario.

- A timestamp (it is not related to the timestamp at network layer) in order to guarantee freshness. Once again, the timestamp is often a frame counter, a key counter or a combination of both.
- An identifier of the key used for creating the MIC. This identifier allows the next hop to find or derive the keys for checking the MIC. Once again, 1 byte should be enough in most cases.
- A MIC of the frame header and payload. The MIC typically is 32, 64 or 128 bits long. Strictly speaking the frame integrity check sequence (e.g. a CRC) can be replaced by this MIC (for example when using TinyOS [12]), and thus the real overhead would be just the difference (if there is any) in size between the MIC and the check sentence.

C. Lossless Aggregation

Since collected/sensed data normally contains just a few bits of metering information, the payload of PHY/MAC packets between meters and their aggregator node is usually far from its maximum allowed or its optimal size. As a result, we propose to concatenate several packets into a single one at aggregator nodes. This concatenation or lossless aggregation reduces unnecessary overhead transmission (headers and MICs). The proposed aggregation process is illustrated in Figure 2 and its execution is detailed in Algorithm 1.

Algorithm 1 Secure lossless aggregation (at every aggregator node).

```
osize = 0
opacket_id = 0
createOutputPacket( opacket_id )
for every input packet do
  if checkMIC() == TRUE then
    mac_data = getPacketMacData()
    if osize + sizeOf( mac_data ) > P'_a then
      createMIC( opacket_id )
      sendPacket( opacket_id )
      opacket_id = opacket_id + 1;
      createOutputPacket( opacket_id )
      osize = 0
    end if
    aggregateInputPacketPayloadIntoOutputPacket(
    mac_data, opacket_id )
    osize = osize + sizeOf( mac_data )
    if last received packet OR timeout then
      createMIC( opacket_id )
      sendPacket( opacket_id )
    end if
  end if
end for
```

From the aggregator node to the gateway, intermediate nodes have only to check the MAC/PHY integrity/authentication of every received packet, and forward the packet with a new MIC and updated headers. Integrity,

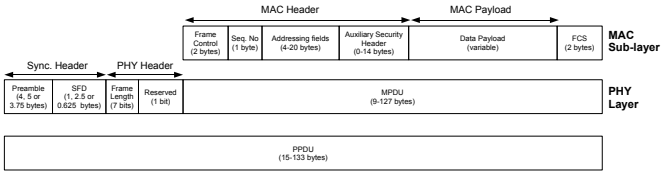

Fig. 3. The IEEE 802.15.4 frame.

authentication and freshness at PHY/MAC layers are therefore checked at every hop. The resulting packets at the aggregator will be made of the following fields:

- MAC/PHY header: that also includes the key identifier used for hop-by-hop security and timestamp.
- for $i = 1$ until $i = n$ with n the number of aggregated input packets at every output packet.
  - Network header of the ith meter's packet.
  - Encrypted data of the ith meter's packet.
  - MIC of the ith meter's packet.
- PHY/MAC MIC.

Summarizing, the aggregator receives the packets, checks their PHY/MAC MICs, combines as many packets as it can into every output packet by concatenation, calculates the MIC of the output packets and sends such packets to the next hop.

III. Performance Evaluation

In this section, we quantify the gains comparing systems with and without aggregation. In order to show the benefits of the protocol, we apply it to 802.15.4, which is the most extended wireless communications technology for remote metering. The 802.15.4 frame format is defined in Figure 3 [13]. As shown in Figure 4, we have defined a complete frame structure for said technology that not only optimizes energy and overhead but also provides a very high level of security.

The PHY headers in Figure 4 are 6 bytes long. We use short addressing (2 bytes per address) for MAC identification but standard addressing (4 bytes per address) for authenticating meters at the gateway. We use AES-CCM with 128 bit shared key between the source (meter) and the gateway for end-to-end security and, for this reason, the payload length is a multiple of the key length (128 bit = 16 bytes). If the payload is shorter than 16 bytes, the protocol pads it with nil bytes to facilitate encryption. As depicted in Figure 4, the minimum payload at MAC sub-layer is 36 bytes and the maximum is 108 bytes. That is to say that a maximum of 3 packets can be aggregated into one packet ($3 \cdot 36$ bytes $= 108$ bytes).

Table I presents the differences Δ in bytes transmitted at the aggregator node with and without lossless aggregation. The table just reflects the impact of aggregation at the output of an aggregator node. The real savings for the whole network grows linearly with the number of hops between the aggregator and the gateway and thus justifies even more the use of lossless aggregation.

The results in Table I are obtained from a varying number of meters N attached to an aggregator and two possible lengths of collected data, 16 and 32 bytes. As a result, since the total overhead (see Figure 4) is $OH_N + OH_M + OH_P = 20 + 19 + 6\text{bytes} = 45$ bytes, the size of the PHY packets generated by the meters P_m can be 61 or 77 bytes. Considering that the aggregator concatenates network packets and that the maximum PHY packet size is P_a, then the maximum number of aggregated packets at the output frame is $A = \left\lceil \frac{P_a - OH_P - OH_M}{P_m - OH_P - OH_N} \right\rceil$ and the total amount of packets at the aggregator output is $O = \lceil N/A \rceil$.

From above reasoning, assuming an error-free channel, the total amount of bytes at the output of the aggregator with and without aggregation as well as the Δ in bytes are obtained as per below expressions to yield the values in Table I.

$$bytes_a = N \cdot (P_m - OH_P - OH_M) + O \cdot (OH_P + OH_M)$$
$$bytes_{na} = N \cdot P_m$$
$$\Delta = Bytes_{na} - Bytes_a = (N - O) \cdot (OH_P + OH_M)$$

Table I clearly shows how the aggregation efficiency grows when the length of the collected data decreases. Since typically collected data in Smart Grid metering applications are just a few bits long, we can save up to a 27% of the bits transmitted at the output of the aggregator, a gain which is further pronounced if multiple hops are present. This gain in overhead translates directly in energy gains since the energy needed to accomplish proposed security is by orders of magnitude lower than the communication energy [14]; the respective results are thus not depicted here for brevity.

So far, a perfect and lossless communication medium has been assumed. However, to conduct a fairer energy expenditure analysis, we also take the PER at PHY and resulting retransmission attempts at MAC into account. Intuitively, longer aggregated packets incur larger errors, more retransmissions and thus larger energy expenditures, something usually not taken into account when aggregation is analyzed. To this end, we assume typical Smart Grid operating conditions, i.e. flat fading channel in frequency; block fading channel in time; and Rayleigh distributed in amplitude. Furthermore, we presume typically used embedded hardware (e.g. CC2500 radio) which relies on binary phase shift keying (BPSK) modulation; no channel coding (block/convolutional); retransmission policies for lost packets.

There are three simplifications worth mentioning here. First,

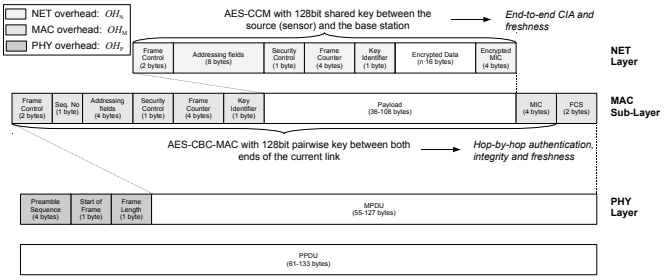

Fig. 4. The proposed aggregation packet format.

TABLE I
COMPARISON OF BYTES TRANSMITTED BY THE AGGREGATOR NODE WITH AND WITHOUT AGGREGATION.

N	collected data (bytes)	P_m (bytes)	$bytes_{na}$	$bytes_a$	Δ (bytes)	Δ (%)
2	16	61	122	97 ($O = 1$)	25	20.49%
3	16	61	183	133 ($O = 1$)	50	27.32%
19	16	61	1159	859 ($O = 7$)	300	25.88%
31	16	61	1891	1391 ($O = 11$)	500	26.44%
53	16	61	3233	2358 ($O = 18$)	875	27.06%
97	16	61	5917	4353 ($O = 33$)	1564	26.43%
2	32	77	154	129 ($O = 1$)	25	16.23%
3	32	77	231	206 ($O = 2$)	25	10.82%
19	32	77	1463	1238 ($O = 10$)	225	15.37%
31	32	77	2387	2012 ($O = 16$)	375	15.71%
53	32	77	4081	3431 ($O = 27$)	650	15.92%
97	32	77	7469	6269 ($O = 49$)	1200	16.06%

we have not considered shadowing which also typically occurs in AMI/Smart Grid settings; including shadowing however is a significantly complicated exposure and hence left for future work. Second, we have not considered higher modulation orders nor channel coding; the inclusion of these however is fairly straightforward [15] and hence not treated here. Third, the retransmission window is assumed to last as long as it takes to get the packet transmitted successfully; in reality, the number of retransmission attempts is limited which, however, clutters analysis and is hence also left for future work.

With above assumptions, and relying on the insights of [15], the average number of transmission attempts is:

$$\overline{N_{tx}}\,(N_b) = \frac{1}{1 - PER\,(\overline{\gamma}, N_b)}, \tag{1}$$

with

$$PER\,(\overline{\gamma}, N_b) \approx 1 - \exp\left[\frac{-4.25\log_{10}(N_b) + 2.2}{2\overline{\gamma}}\right], \tag{2}$$

where $\overline{\gamma}$ is the average signal-to-noise ratio (SNR) experienced and N_b the number of bits per packet (which is equal to the number of symbols due to BPSK). In this calculation, we also assume that the losses related to ACK messages are negligible compared to the much longer data packets. Finally, we also assume that all aggregating Smart Grid links suffer from the same SNR; whilst this is a simplifying assumption, a generalization to arbitrary SNRs is straightforward and thus omitted here.

This average number of transmissions impacts the energy count since the energy used for each transmission and reception now needs to be multiplied by $\overline{N_{tx}}$. This significantly impacts the results, as shown in Table II which presents the energy consumption of our protocol considering the lossy channel. Notably, for average channel SNRs below some 3.5dB, the longer aggregated packets force a lot of retransmissions which deteriorates the energy efficiency. Therefore, as illustrated in Figure 5, for SNR values lower than 3.5dB, the concatenation protocol is not recommended; whereas for SNRs above 3.5dB, the benefit of the proposed concatenation protocol is significant. Finally, in Figure 6, we have also plotted the gains in the case of multihop deployment; here, by increasing the number of hops, the energy savings increases. Concatenation facilitates energy savings and thus lifetime extension of 27% for each hop assuming 16 byte of data payload in every packet.

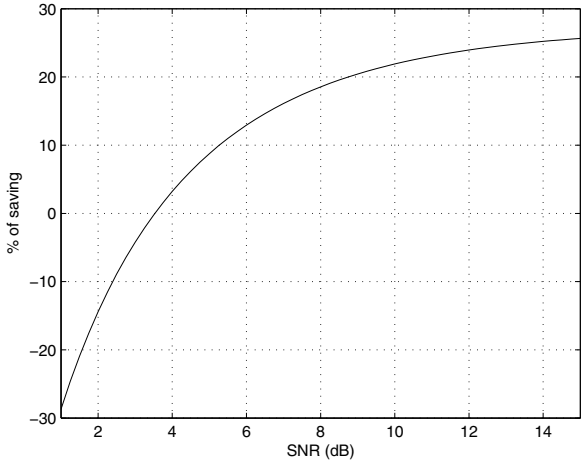

Fig. 5. Energy savings of aggregation w.r.t. no aggregation over lossy single-hop wireless channels.

IV. CONCLUDING REMARKS

Meters in Smart Grids aim to infer the meter information and deliver this information reliably and securely to the gateway, so that it can reach the utility companies. The inferred data however is often composed of a few bits which, if used in the context of standardized solutions with minimum and maximum packet lengths, yields high overheads and hence poor energy efficiency. Aggregation is hence a natural solution where, due to the need to identify each node and its associated inferred metering data, requires lossless aggregation mechanisms. Aggregation, however, poses extra challenges on per-hop and end-to-end security since aggregating nodes essentially need to access the information content. Design issues are further complicated by the fact that the wireless medium is lossy in nature.

TABLE II
ENERGY CONSUMPTION WITH A VARYING SNR.

N	$OH_P + OH_M$ (bytes)	collected data (bytes)	SNR (dB)	$\overline{N_{tx}}$ w/o aggr.	$\overline{N_{tx}}$ with aggr.	Total tx bytes w/o aggr.	Total tx bytes with aggr.	Energy consumption w/o aggr. (mJ)	Energy consumption with aggr. (mJ)
3	21	16	0	100.77	207	18440.9	27531	2743.9	4096.6
3	21	16	5	4.3	5.4	786.9	718.2	117	106.868
3	21	16	10	1.586	1.7	290.238	226.1	43.19	33.6436
3	21	16	15	1.157	1.1836	211.731	157.41	31.5	23.422
2	21	32	0	124.93	201	19239.22	26545	2862.76	3949.896
2	21	32	5	4.6	5.35	708.4	690.15	105.41	102.672
2	21	32	10	1.62	1.7	249.48	219.3	37.12	32.63
2	21	32	15	1.165	1.1826	179.41	152.478	26.696	22.6887

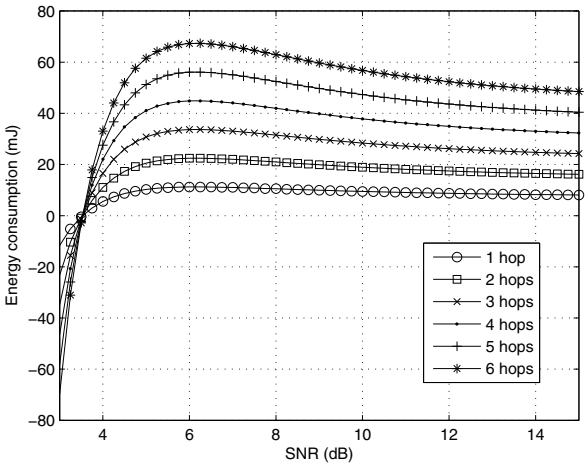

Fig. 6. Difference in energy expenditure between aggregation and no aggregation for a varying number of hops.

The aim of this paper was thus to propose a viable and readily deployable protocol which exhibits above features, as well as quantifying the performance gains assuming some realistic wireless channel and typically deployed hardware. Based on IEEE 802.15.4 standards settings, we have shown that the developed protocol is indeed secure providing per-hop authentication as well as end-to-end confidentiality, is reliable and energy efficient.

Notably, the developed protocol has been analyzed considering different scenarios (packet length, SNR, etc.) in order to evaluate its efficiency. The analysis of the SNR revealed that only in very noisy channels this solution is not useful because the high number of packet losses, and in consequence, the overall number of transmitted bytes (including the necessary retransmissions) is larger than when concatenation is not applied. We have shown that, due to the lossy nature of the wireless channel, a breakpoint occurs at about 3.5dB SNR before which aggregation is not recommended. These low SNRs typically occur in range-limited meter deployments. The analysis thus showed the optimal operating point beyond which data aggregation is useful as well quantifying the performance superiority with respect to non-aggregated solutions.

ACKNOWLEDGMENT

This work is supported by a France Telecom research contract on machine-to-machine (M2M) security.

REFERENCES

[1] (2009) National broadband plan. [Online]. Available: www.broadband.gov/download-plan

[2] C. Dugas, "Configuring and managing a large-scale monitoring network: solving real world challenges for ultra-low powered and long-range wireless mesh networks," in *Int. J. Network Mgmt*, 2005.

[3] C. Castelluccia, A. C.-F. Chan, E. Mykletun, and G. Tsudik, "Efficient and provably secure aggregation of encrypted data in wireless sensor networks," *ACM Trans. Sen. Netw.*, vol. 5, no. 3, pp. 1–36, 2009.

[4] S. Peter, K. Piotrowski, and P. Langendoerfer, "On concealed data aggregation for wireless sensor networks," 2005.

[5] E. Mykletun, J. Girao, and D. Westhoff, "Public key based cryptoschemes for data concealment in wireless sensor networks," in *In IEEE International Conference on Communications. ICC2006*, 2006.

[6] H. Chan, A. Perrig, and D. Song, "Secure hierarchical in-network aggregation in sensor networks," in *CCS '06: Proceedings of the 13th ACM conference on Computer and communications security*. New York, NY, USA: ACM, 2006, pp. 278–287.

[7] A. Mahimkar, "Securedav: A secure data aggregation and verification protocol for sensor networks," in *In Proceedings of the IEEE Global Telecommunications Conference*, 2004, pp. 2175–2179.

[8] B. Przydatek, D. Song, and A. Perrig, "Sia: secure information aggregation in sensor networks," in *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2003, pp. 255–265.

[9] Y. Yang, X. Wang, S. Zhu, and G. Cao, "Sdap: A secure hop-by-hop data aggregation protocol for sensor networks," *ACM Trans. Inf. Syst. Secur.*, vol. 11, no. 4, pp. 1–43, 2008.

[10] S. Ozdemir and Y. Xiao, "Secure data aggregation in wireless sensor networks: A comprehensive overview," *Computer Networks*, vol. 53, no. 12, pp. 2022 – 2037, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/B6VRG-4VXB88W-1/2/19d5ff6af92871bd6aff85ac5de4ae8d

[11] E. Mlaih and S. A. Aly, "Secure hop-by-hop aggregation of end-to-end concealed data in wireless sensor networks," *CoRR*, vol. abs/0803.3448, 2008.

[12] C. Karlof, N. Sastry, and D. Wagner, "Tinysec: a link layer security architecture for wireless sensor networks," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*. New York, NY, USA: ACM Press, 2004, pp. 162–175. [Online]. Available: http://dx.doi.org/10.1145/1031495.1031515

[13] (2010) The ieee 802.15.4 standard association website. [Online]. Available: http://web.archive.org/web/20080224051703/standards.ieee.org/getieee802/download/802.15.4-2006.pdf

[14] A. Wander, N. Gura, H. Eberle, V. Gupta, and S. Shantz, "Energy analysis of public-key cryptography for wireless sensor networks," in *PERCOM*, 2005.

[15] R. Zhang, "Analysis of energy-delay performance in multi-hop wireless sensor networks," PhD Thesis, 2009.