Data Aggregation Using Homomorphic Encryption in Wireless Sensor Networks

T.D. Ramotsoela
Department of Electrical, Electronic and Computer Engineering
University of Pretoria
Tshwane, South Africa
u10350749@tuks.co.za

G.P. Hancke
Department of Computer Science
City University of Hong Kong
Kowloon, Hong Kong
gp.hancke@cityu.edu.hk

Abstract—**Secure information aggregation using homomorphic encryption in wireless sensor networks allows data to be aggregated without having to decrypt the packets. While data aggregation provides a means to reduce network traffic, homomorphic encryption increases the size of the packets and this could negatively affect system performance. This is because energy consumption of the nodes is directly proportional to the amount of data transferred. In this paper, the effect of this increased packet size was investigated for the Domingo-Ferrer encryption scheme and compared to a symmetric encryption scheme. It was found that the symmetric encryption scheme outperforms the homomorphic encryption scheme for smaller networks, but as the network size grows, homomorphic encryption starts outperforming symmetric encryption. It was also found that the homomorphic encryption scheme does not significantly reduce the performance of plaintext aggregation.**

Keywords-Aggregation; homomorphic encryption; network security; wireless sensor networks

I. Introduction

Wireless sensor networks (WSNs) have become increasingly popular in many applications such as environment monitoring [1] and law enforcement [2]. The networks consist of a number of cheap sensor nodes which consist of a sensor, a processor, and a power source [3] together with a sink and a back-end platform [4]. The sensor changes depending on the specific application the sensor node is used in but the processor is usually a simple processor with low computational power. The power source is usually a limited power supply such as a commercial battery.

These resource constraints mean that the efficiency of data transfer is paramount in these applications so any protocols must be design with these constraints in mind [5], [6]. This is because the energy consumption of the nodes is directly proportional to the amount of data transmitted [3]. One of the methods that can be used to reduce traffic in the network is called data aggregation. This process involves combining data coming from different sources enroute [7]. Aggregation however becomes a problem when security is an issue in the system [8]. This is because each node would have to decrypt each packet, aggregate the data, and then encrypt the result before sending it to the next hop. Secure information aggregation in WSNs is a growing field and is especially important for growing number of Internet-of-Things applications [9], [10]. One of the proposed solutions is called homomorphic encryption. Homomorphic encryption is an encryption scheme that allows data processing on encrypted data as opposed to plaintext [11].

This paper investigates the use of homomorphic encryption in data aggregation for WSNs. In using homomorphic encryption, the data can be aggregated without having to decrypt each incoming packet because the encrypted data can be aggregated directly. The resulting ciphertext is however usually much larger than the original plaintext [12]. Since the energy consumption of the nodes is directly proportional to the amount of data transmitted, it can be deduced that the larger the volume of network traffic is, the poorer the system performance will be. This is because an increase in packet size depletes the power sources of the nodes faster than if encryption was not used. The main objective of the investigation is to determine whether this increase in packet size has a significant effect on system performance compared to traditional encryption mechanisms.

The rest of the paper is organised as follows. Section II gives a general overview of WSNs and their security and section III describes the experimental setup. In section IV the results of the experiment are presented and they are discussed in section V. Finally, the paper is concluded in section VI.

II. Background

A. Wireless Sensor Networks

Wireless sensor networks can be defined as "a network of devices, denoted as nodes, which can sense the environment and communicate the information gathered from the monitored field (e.g. an area or volume) through wireless links" [13]. They were proposed by the military of the United States of America in the 1970's [14]. It was not until the end of the 20th century that wireless sensor networks started becoming popular in applications not related to the military.

978-1-4799-7755-0/15/$31.00 ©2015 IEEE

This is largely attributed to the advances in the related fields, such as microelectronics and telecommunications.

One of the key features of WSNs is minimising the power consumption in the nodes [15]. This, along with functions such as managing network protocols as well as interfacing the sensing and communicating units is the responsibility of the processing unit. In addition to the power constraints, another limitation of WSNs is the computational power of the processing unit [2]. The networks themselves are also limited in terms of bandwidth. This is what makes the efficiency of data transfer so important in these applications.

B. The Security of Wireless Sensor Networks

The three key objectives of network security are identified as confidentiality, integrity and availability [16]. Confidentiality means that only authorised parties are allowed to access data while integrity means that no unauthorised parties should be allowed to modify the data. When talking about availability, the implication is that the system/data should be available when required. These three concepts combine to form what is referred to as the CIA triad.

The order of importance in smart grids is availability, followed by integrity and then confidentiality as opposed to normal IT networks where the order is reversed [17]. Smart grids use sensor networks to perform their tasks; therefore, by extension the same can also be said for WSNs. This shows that, in most cases, normal network security policies cannot be applied to WSNs in their original format because the priorities are reversed.

The characteristics of WSNs that make them vulnerable to security attacks are [18]: (1) they are openly accessible to everyone, (2) security isn't designed into the protocols, (3) they have limited resources so the protective measures that can be implemented are limited, and (4) they are usually deployed in hostile environments. These characteristics make it difficult to protect WSNs in comparison to computer networks.

The availability of the system has already been noted as being the most important security objective of WSNs when considering the CIA triad. Denial of service (DoS) attacks attempt to compromise the availability of a system [19]. While confidentiality and integrity are assessed using a binary scale (i.e. they have either been maintained or compromised), availability is a bit more difficult to classify.

Denial of service is also not necessarily the result of an attack, a fault in the system could also affect system performance. So it is important to classify what is or is not acceptable with regards to availability. The designers have to determine a threshold, like the minimum data throughput the system requires to perform its task effectively. Anything falling outside the bound could then be considered denial of service and corrective measures should then be taken. WSNs use a layered architecture and DoS attacks can occur in any of those layers [20]. This makes protecting the system against these types of attacks more challenging.

Integrity is almost as important as availability because a loss in integrity can also lead to a loss of availability [21]. In network security, digital signatures are used as a countermeasure to integrity failure. A digital signature consists of a file, a demonstration that the file has not been altered, the identity of the signer, and verification that the signature is authentic [22]. Public-key cryptography is used as a means to verify the sender's identity and ensure that the message digest has not been modified in transit. Public-key cryptography is however too computationally expensive for WSN applications. Instead, symmetric cryptography is used to verify integrity. A popular symmetric encryption scheme used in WSN applications is called µTesla. It uses a key chain of symmetric keys instead of public-key cryptography [23].

The common physical attacks on integrity include node compromise and replication [24]. When considering message related attacks, attackers usually attempt to alter a message they captured in transit or replay an old unaltered message. The latter is referred to as a replay attack and is not as harmless as it may seem. Replay attacks could be used to shut down the system by flooding the network with seemingly legitimate messages [19]. By doing this, the resources of the network could become exhausted. The last kind of common attack is where the attacker sends a message falsely portraying it as one from a legitimate node in the WSN. This kind of attack is called a counterfeiting attack [24].

As already mentioned, the least important security feature of WSNs is confidentiality. Confidentiality is however still a major concern and loss of confidentiality might lead to catastrophic consequences for the users [25]. The common confidentiality attacks on WSNs are eavesdropping, node tampering and node replication [24]. The main issue with trying to protect the confidentiality of WSNs is that they have very limited resources. While public key cryptography has been shown to be feasible for use in WSNs, they are still too computationally expensive [20]. Symmetric key cryptography is more efficient but their key management schemes are far from ideal.

C. Secure Information Aggregation in WSNs

In WSNs, there is a sink node that collects the information from the other nodes in the network [26]. Packets from each node can be sent independently via the shortest path to the sink node. This process is referred to as an address-centric protocol and it is used in computer networks [7]. Using this protocol, each node acts as a passive router to packets sent from other nodes. This means that it does not perform any operations on the data before it transmits it to the next hop. While this protocol is efficient in computer networks such as the Internet, the same cannot be said for applications such as wireless sensor networks and smart meter communication networks. This is because the mentioned networks use application specific data and the entire process can be optimised by aggregating the data [7].

Using data aggregation, the data is pre-processed at each node before it is transmitted to the next hop [26]. In this way, the routing occurs along a reversed multicast tree with the sink being the root node. This process is referred to as a data-centric protocol and the tree is called an aggregation tree [7]. Using data aggregation in this setting has a number of advantages such as reducing the number of transmissions and getting rid of redundancy. One drawback of using aggregation is the latency caused by the processing and possible buffering of data at each node.

When the security of the system is a concern, encryption can be used to protect the data. The problem with this is that most in-network aggregation schemes assume that all the sensor nodes are trusted [27]. This is a problem because they are allowed to view the data that passes through them from other nodes (e.g. they decrypt the data in order to perform the aggregation). While this could acceptable for some applications, it might not be the case for others such as smart metering systems. Another issue is that each node suffers significant overhead as a result of having to decrypt, aggregate, and then re-encrypting the result before transmission. Homomorphic encryption can be used instead of conventional end-to-end encryption to alleviate these problems [27].

Homomorphic encryption is an encryption scheme that allows data processing on encrypted data as opposed to plaintext [11]. The implication of this is that each intermediate meter does not need to decrypt the data in order to perform the aggregation task. An important security feature of this scheme is that for a given encryption key, each plaintext can be encrypted into a number of different ciphertexts [28]. This means that plaintext is shorter than the resulting ciphertext and this difference in length should be chosen to be small as possible depending on the application. Having different ciphertexts for the same plaintext makes this algorithm resistant to dictionary attacks [12].

The operations that can be performed using the homomorphic encryption scheme are multiplication and addition. A fully homomorphic encryption scheme is one that can perform both types operations on encrypted data [28]. Fully homomorphic encryption schemes are not yet efficient in practice so there are very few applications that implement them. In Somewhat fully homomorphic encryption schemes, a limited number of multiplication operations are allowed while there is no limit to the number of additions allowed [29]. Their overhead is however still too high for direct implementation in practical applications [28]. Schemes that can perform only one type of operation at a time are called either additive or multiplicative [12]. These are the preferred schemes in practice because their overhead is acceptable in most applications [28].

III. Experimental Setup

The simulation was done using Network Simulator 2 (NS2), which is the accepted method for academic WSN simulation [30]. The comparative end-to-end encryption scheme used in this experiment is the RC4 encryption algorithm which is popular in WSN applications [20]. Although RC4 has been proven to be vulnerable to security attacks, and not really recommended in modern systems, many applications of small and portable devices still use it because of its speed and efficiency [31], e.g. keystream used to create ciphertext can be created very quickly and could be pre-computed. These applications normally use variants of the algorithm that have improved security features. Despite its security issues, RC4 is a strong competitor for our study as we are looking to compare the network overhead involved, of which RC4 introduces a minimal amount, i.e. data packets of any length can be encrypted without additional padding as would be the case for block ciphers like the Advanced Encryption Standard (AES).

The homomorphic encryption scheme that will be used is the Domingo-Ferrer encryption algorithm [32]. It was used in the popular secure aggregation scheme that proposes using concealed data aggregation along a reverse multicast tree [3]. For the purposes of this experiment, the aggregation tree construction will be ignored and the primary focus will be the network traffic.

A. Domingo-Ferrer Encryption

The Domingo-Ferrer encryption scheme is an asymmetric homomorphic encryption scheme. As already mentioned, asymmetric cryptography is generally too computationally expensive for WSN applications. The authors in [3] however found that limiting the size of the security parameters makes this scheme feasible for practical implementation. This happens at the expense of the security of the scheme, but it was found that it still provides an appropriate level of security.

The public parameters are a large integer g which is 10^{200} or larger and a positive integer d which should be greater than two [32]. The large integer g should have many small divisors and also many integers smaller than it that can be inverted modulo g. The first limitation proposed in [3] is that d should not be greater than 4 and should include the lower bound 2. The second limitation is that g should not be greater than 2^{32}. The secret parameters are a positive integer $r \in Z_g$ (which should be chosen such that $r^{-1} mod\ g$ exists) and a positive integer g' such that $log_{g'}g$ is a secret security parameter [32]. The secret key of the scheme is thus (r, g').

To encrypt a number $m \in Z_{g'}$, d random numbers (s_1 to s_d) should be generated such that $m = \sum_{j=1}^{d} s_j\ mod\ g'$ and $s_j \in Z_g$. The ciphertext is then found using equation 1 below.

$$E(m) = (s_1 r\ mod\ g, s_2 r^2\ mod\ g, \dots, s_d r^d\ mod\ g) \qquad (1)$$

To decrypt the ciphertext, the j^{th} coordinate is computed by $r^{-j} mod$ g to retrieve $s_j\ mod\ g$. The plaintext is then found using equation 2 below.

$$D(E(m)) = \sum_{j=1}^{d} s_j\ mod\ g' \qquad (2)$$

The addition and subtraction operations are done componentwise while the multiplication operation is done by cross multiplying the components in Z_g like polynomials. The division operation is not supported by this scheme [32].

B. *Network Topology*

The network topology was chosen such that a direct comparison between the data-centric and address-centric protocols can be made. It was also chosen such that the aggregation can be done without having to construct the aggregation tree. The nodes were clustered into groups of four that were all within range of one another. One of the nodes in each cluster was placed within range of a node in another cluster. This will be referred to as the boundary node.

Four network sizes were considered in this experiment: 12, 24, 36 and 48 nodes. These networks had 3, 6, 9 and 12 node clusters respectively. This network configuration was able to simulate what happens as the distance from the sink (in terms of number of hops) increases so it was not necessary to implement larger networks. Using this network configuration, it was also possible to deduce how the network would behave under different circumstances without having to change the network topology.

The network grows vertically as the number of clusters increases. The boundary node of the last cluster is taken to be the sink of the network. It is assumed that the sink does not send any packets but only receives them. For example, in the 12 node network, the boundary node of the 3rd cluster is the sink. For the 24 node network, the boundary node of the 6th cluster is the sink and that of the 3rd cluster is considered just a normal boundary node. From this point forward, the boundary node of the nth cluster will be referred to as boundary node n. The previous explanation is important because while boundary node 3 does not send any packets in the 12 node network, it does in the 24 node network. This distinction will be important in the results section.

C. *Functionality*

For address-centric routing, messages are sent to the sink via the shortest path. For data-centric routing, each node sends its packet to the boundary node in its cluster. The boundary node then aggregates the data of all its children with its own and sends the result to the in range node of the next cluster. This node then aggregates that data with its own and sends it to its own boundary node like all its siblings. The process continues until the sink receives data from all its children.

The following scenarios were implemented and compared. Raw end-to-end data (no aggregation, no encryption), encrypted end-to-end data (no aggregation, confidentiality), raw aggregate data (aggregation, no encryption), and encrypted aggregate data (HE aggregation).

The network key of the rc4 encryption algorithm is of no significance since it does not affect the size of the packet. The security parameters of the Domingo-Ferrer encryption do affect the size of the packet so they will be mentioned here. The size of the large integer g was chosen to be 2^{32}, which is the largest it can be as explained previously. The value of g′ does not affect the packet size, but it was chosen as 2^{16} for this experiment. The value of r, which primarily depends on g, was chosen to be 30027. For the experiment, the packet data was 3 bytes long before encryption and a header of 17 bytes was assumed. This is the approximate size of the IEEE 802.15.4 overhead [33].

IV. Results

A. *Address-centric Routing*

TABLE I shows the results of the NS2 simulation for the address-centric routing. The first column (B-node) indicates the boundary node. It is followed by the plaintext, RC4 and distributed (explained later) columns respectively. Each of the results columns has the number of bytes received and transmitted by each boundary node.

TABLE I. Boundary nodes traffic in address-centric routing

	plaintext		rc4		distributed	
B-node	Rec	Trans	Rec	Trans	Rec	Trans
12 nodes						
1	60	80	60	80	0	20
2	140	160	140	160	20	40
sink	220	0	220	0	220	0
24 nodes						
3	220	240	220	240	40	60
4	300	320	300	320	60	80
5	380	400	380	400	80	100
sink	460	0	460	0	460	0
36 nodes						
6	460	480	460	480	100	120
7	540	560	540	560	120	140
8	620	640	620	640	140	160
sink	700	0	700	0	700	0
48 nodes						
9	700	720	700	720	160	180
10	780	800	780	800	180	200
11	860	880	860	880	200	220
sink	940	0	940	0	940	0

It is now important to take the earlier discussion about the different boundary nodes into consideration. The row named 12 nodes marks the beginning of the 12 node network which has 2 boundary nodes and a sink node. The row named 24 nodes marks the beginning of the 24 node network and the end of the 12 node network. It has 5 boundary nodes and a sink node. It was explained earlier that it is assumed that a sink node does not send any data but a boundary node does. So when analysing the results of the 24 node network, the sink node of the 12 node network is ignored and boundary node 3 is considered instead. So the network traffic of boundary nodes 1 and 2 are identical for both networks. However the sink node in the 12 node network does not send any data, but it is a boundary node in the 24 node network which does send data. So in summary, the nodes of the 12 node network are 1, 2 and sink. In the 24 node network, the nodes are 1, 2, 3, 4, 5 and sink. In the 36 node network, the nodes are 1, 2, 3, 4, 5, 6, 7, 8 and sink. By extension, the same is done for the 48 node network.

The column named distributed was not the result of an NS2 simulation. By studying the network behaviour of the simulated configuration, the behaviour of a best case scenario for the topology was deduced. This was done because the simulated network used the worst case scenario where the shortest path to the sink is only one route. In the simulated network, only the boundary node of a cluster is within range with only one other node in the next cluster. This meant that all the data of a particular cluster had to pass through the boundary node to get to the next cluster.

In the best case scenario, it is assumed that all the nodes in the cluster can communicate with all the nodes in the next cluster. It is assumed that this communication happens in a distributed fashion such that all the nodes in a cluster receive and transmit the same amount of data. For example, the nodes in cluster 1 send their packets to different nodes in cluster 2. The nodes in cluster 2 then send the packets of cluster 1 and their own packets to different node in cluster 3 and so on. For the 12 node network, each node in cluster 2 would be able to directly communicate with the sink node. So in this case, all the packets are sent directly to the sink node. This means that the nodes in the same cluster as the sink don't receive any packets from other clusters.

B. *Data-centric Routing*

TABLE II shows the results of the NS2 simulation for the data-centric routing. The first column (B-node) indicates the boundary node. It is followed by the plaintext, and the Domingo-Ferrer encryption scheme columns respectively. The network was simulated with each of the 3 possible values for *d*. Each of the results columns has the number of bytes received and transmitted by each boundary node. The structure of the table is the same as for the address-centric protocol.

C. *Combined Results*

The energy consumption of the radio is of the same order of magnitude whether it is receiving or transmitting data [34]. This means that a more accurate measure of system performance looks at the net traffic through a node. TABLE III shows the combined results of all the net traffic though each boundary for the 48 node network. These results are graphed in Figure 1. The figure excludes the non-distributed results for address centric routing because its performance is far worse than the other scenarios. It also excludes the results of the sink which will be discussed from the table.

V. DISCUSSION OF RESULTS

A. *Address-centric Routing*

From TABLE I it is clear that the RC4 encryption does not increase the packet size which is one of the reasons why it is so popular in WSN application. Address-centric routing in the simulated case however has very poor results. In this case, there is only one route for the shortest path and the boundary nodes receive and transmit vast amounts of data. As the distance from the sink node increases, the boundary nodes closer to the sink will deplete their power sources very quickly.

TABLE II. BOUNDARY NODES TRAFFIC IN DATA AGGREGATION

	Plaintext		HE (d=2)		HE (d=3)		HE (d=4)	
B-node	Rec	Trans	Rec	Trans	Rec	Trans	Rec	Trans
12 nodes								
1	60	20	111	37	141	47	171	57
2	60	20	111	37	141	47	171	57
sink	60	0	111	0	141	0	171	0
24 nodes								
3	60	21	111	37	141	47	171	57
4	61	21	111	37	141	47	171	57
5	61	21	111	37	141	47	171	57
sink	61	0	111	0	141	0	171	0
36 nodes								
6	61	21	111	37	141	47	171	57
7	61	21	111	37	141	47	171	57
8	61	21	111	37	141	47	171	57
sink	61	0	111	0	141	0	171	0
48 nodes								
9	61	21	111	37	141	47	171	57
10	61	21	111	37	141	47	171	57
11	61	21	111	37	141	47	171	57
sink	61	0	111	0	141	0	171	0

TABLE III. NET TRAFFIC THROUGH BOUNDARY NODES

	Address		Aggregation			
B-node	RC4	Dist	Plain	HE (d=2)	HE (d=3)	HE (d=4)
1	140	20	80	148	188	228
2	300	60	80	148	188	228
3	460	100	80	148	188	228
4	620	140	80	148	188	228
5	780	180	80	148	188	228
6	940	220	80	148	188	228
7	1100	260	80	148	188	228
8	1260	300	80	148	188	228
9	1420	340	80	148	188	228
10	1580	380	80	148	188	228
11	1740	420	80	148	188	228
sink	940	940	61	111	141	171

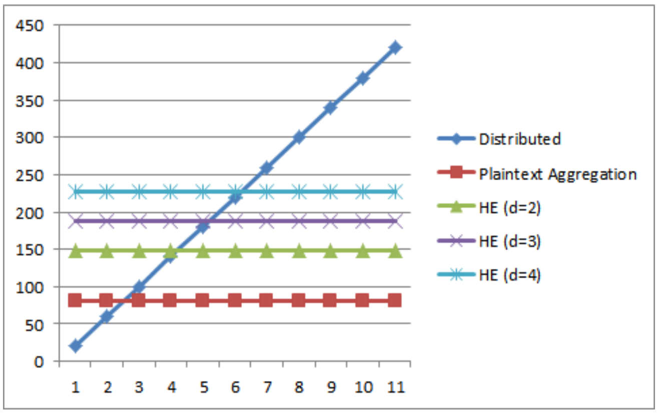

Figure 1. Net traffic through boundary nodes

When the load is distributed among all the nodes in a cluster, the results are far better in that the boundary node reduces its energy consumption by a factor of c, where c is the number of nodes in a cluster. This however coincides with an increase in energy consumption for all the other nodes in the cluster. This is more acceptable though since more nodes last longer, meaning the system as a whole would function longer. It can however be seen that the nodes closer to the sink still deplete their power sources much fast than those further away. The nodes in cluster 1, for example, will last approximately 21 times longer than those in cluster 11 assuming they use the same power source.

Looking at the smaller networks though, the results are much better. In the 12 node network, the furthest node from the sink is only 2 hops away and the intermediary nodes only last 3 times longer than the furthest nodes. In the 24 node network the furthest nodes are 5 hops from the sink and they last 9 times longer than those that are only 1 hop from the sink.

It is clear from the results that when using address-centric routing, the number of nodes furthest from the sink should be as small as possible. It is also important for the nodes to have multiple paths to the sink so as to distribute the load across as many nodes as possible. Not only does this help with network congestion, but also with the energy consumption of the nodes.

B. *Data-centric Routing*

Looking at TABLE II, the benefits of using aggregation are immediately visible when looking at the plaintext results. Each cluster in the network consumes almost the exact same amount of energy. The minor difference between boundary node 2 and 3 are due to an increase in size of the aggregate data. It increases from 3 bytes to four bytes which is a 1.25% increase of the net traffic amount so it can be considered negligible. The nodes that don’t perform any aggregation tasks (i.e. the leaf nodes) will last approximately 4 times longer than the aggregating nodes. This is comparable to address-centric routing when the furthest nodes are only 2 hops away from the sink. An important thing to note is that its performance doesn’t change as the distance from the sink increases. The amount of data in the network is also significantly reduced so it aids with network congestion.

Looking at the Domingo-Ferrer results, when d = 2, the nodes have a net traffic that is approximately 1.85 times larger than if encryption was not used. The net traffic is approximately 2.35 and 2.85 larger for d=3 and d=4 respectively. So when using this scheme under the specified security parameters, the energy consumption of the nodes will, in the worst case, be increased by a factor of 3 and the network traffic will be increased by the same amount. From the results it is clear that while this scheme does affect system performance, it will be acceptable for most applications when security is a concern.

C. *Combined Results*

From TABLE III, it is clear that the performance of the simulated address-centric protocol is far worse than all the aggregation scenarios. It is for this reason that only the distributed results will be considered. Looking at both TABLE III and Figure 1, the 12 node network outperforms even the plaintext aggregation. In this scenario the furthest node from the sink is only two hops away. Increasing the number of nodes in each cluster would not affect its performance but it would reduce the performance of aggregation. The only performance benefit aggregation has in this case is network congestion. This sink will still receive $(n-1) \times 20$ bytes of data in address-centric routing, where n is the number of nodes in the network (including the sink). In aggregation however, the sink will only receive approximately $k \times 20$ bytes of data, where k is the number of children it has. Network congestion can also become a problem in aggregation, but it can be alleviated by limiting the number of children each aggregator node can have [12].

For the 24 node network, the results are still comparable. Even though the nodes in cluster 5 have a net traffic that is 2.25 times larger than the aggregation nodes, the rest of the clusters have similar or better results than the aggregating nodes. In this case the furthest node is only five hops away from the sink. There will be benefits to using either aggregation or address-centric routing depending on the network constraints and the number of nodes in the network. So in this case it is very application specific. As the furthest nodes get further away from the sink however, the benefits of using aggregation start outweighing those of address-centric routing.

Comparing the Domingo-Ferrer encryption scheme to the address-centric protocol, it is not until the 36 node network that the HE scheme starts outperforming it for d=2. In this scenario, the furthest node from the sink is 8 hops away. It is in this network that the average net traffic of the nodes in the address-centric protocol becomes larger than that of the aggregating nodes. This happens at the 40 and 44 node networks for d=3 and d=4 respectively. This is when the furthest nodes from the sink are 9 and 10 hops away respectively. There will again here be performance benefits to using either aggregation or address-centric routing. Network traffic through the nodes closest to the sink is still a major concern for the address-centric protocol. It can also be seen that as the furthest node gets further away from the sink, aggregation becomes the superior choice.

In summary, the results of the simulated address-centric protocol were far worse than those of any of the aggregation schemes. The best case results were more comparable to the aggregation schemes, outperforming them for smaller networks. The two address-centric scenarios are however two extremes, one is extremely efficient and the other one's performance is quite awful. In reality though, one would normally get a network that is in between the two extremes. So the network topology and routing algorithms play a vital role in network performance. The results of this experiment are however able to show a general trend in the performances of the schemes. For smaller networks, where the furthest node is not too far from sink, using RC4 is generally better than Domingo-Ferrer. As the network size grows and with it, the distance between the sink and the furthest nodes, Domingo-Ferrer starts outperforming RC4. Where the performances are comparable, factors such as network topology, size and congestion should be taken into considering when choosing between the schemes. It is also important to note that the Domingo-Ferrer nodes will, in the worst case, increase energy consumption by a factor of only 3 when compared to plaintext aggregation. This means that it does not significantly reduce the performance of the plaintext aggregation.

VI. Conclusion

Wireless sensor networks have become increasingly popular in many applications such as environment monitoring and law enforcement. Data aggregation is a method used to reduce network traffic but cannot be used together with conventional encryption schemes because it is not secure and introduces extra overhead. Homomorphic encryption is an encryption scheme that allows data processing on encrypted data as opposed to plaintext. It has the benefit that each intermediate node does not have to decrypt each packet, but the resulting ciphertext is usually much larger than the original plaintext. This could negatively affect system performance because the energy consumption of each node is directly proportional to the amount of data it transmits.

It was found that for smaller networks, where the furthest node is not too far from sink, using RC4 is generally better than Domingo-Ferrer. As the network size grows and with it, the distance between the sink and the furthest nodes, Domingo-Ferrer starts outperforming RC4. Where the performances are comparable, factors such as network topology, size and congestion should be taken into considering when choosing between the schemes. It was also found that the Domingo-Ferrer encryption scheme does not significantly reduce the performance of plaintext aggregation. This means that the Domingo-Ferrer encryption scheme is feasible for WSN applications when considering its effect on network traffic.

References

[1] A Kumar and G Hancke, "Energy Efficient Environment Monitoring System Based on the IEEE 802.15.4 Standard for Low Cost Requirements," *IEEE Sensors Journal*, vol. 14, no. 8, pp. 2557-2566, March 2014.

[2] C Castelluccia, E Mykletun, and G Tsudik, "Efficient Aggregation of encrypted data in Wireless Sensor Networks," in *Mobile and Ubiquitous Systems: Networking and Services Conference*, San Diego, CA, USA , 2005, pp. 109-117.

[3] J Girao, D Westhoff, and M Schneider, "CDA: Concealed Data Aggregation for Reverse Multicast Traffic in Wireless Sensor Networks," in *IEEE International Conference on Communications*, 2005, pp. 3044 - 3049.

[4] R Fisher, L Ledwaba, G Hancke, and C Kruger, "Open Hardware: A Role to Play in Wireless Sensor Networks?," *MDPI Sensors*, vol. 15, no. 3, pp. 6818-6844, 2015.

[5] S Chinnapen-Rimer and G Hancke, "Actor coordination using info-gap decision theory in wireless sensor and actor networks," *Inderscience International Journal of Sensor Networks*, vol. 10, no. 4, pp. 177-191, October 2011.

[6] A Abu-Mahfouz and G Hancke, "An Efficient Distributed Localization Algorithm for Wireless Sensor Networks: Based on Smart References Selection Criteria," *Inderscience International Journal of Sensor Networks*, vol. 13, no. 2, pp. 94-111, May 2013.

[7] B Krishnamachari, D Estrin, and S Wicker, "The impact of data aggregation in wireless sensor networks," in *Proceedings of the 22nd International Conference on Distributed Computing Systems*, 2002, pp. 575-578.

[8] C Castelluccia, E Mykletun, and G Tsudik, "Efficient Aggregation of encrypted data in Wireless Sensor Networks," in *Mobile and Ubiquitous Systems: Networking and Services Conference*, 2005, pp. 109-117.

[9] G Hancke, K Markantonakis, and K Mayes, "Security Challenges for User-Oriented RFID Applications within the 'Internet of Things," *Journal of Internet Technology*, vol. 11, no. 3, May 2010.

[10] C Kruger and G Hancke, "Implementing the Internet of Things vision in industrial wireless sensor networks," in *IEEE International Conference on Industrial Informatics (INDIN)*, 2014, pp. 627-632.

[11] N Saputro and K Akkaya, "Performance Evaluation of Smart Grid Data Aggregation via Homomorphic Encryption," in *IEEE Wireless Communication and Networking Conference (WCNC)*, Shanghai, 2012, pp. 2945-2950.

[12] F Li, B Luo, and P Liu, "Secure Information Aggregation for Smart Grids Using Homomorphic Encryption," in *First IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Gaithersburg, MD, 2010, pp. 327-332.

[13] C Buratti, A Conti, D Dardari, and R Verdone, "An Overview on Wireless Sensor Networks Technology and Evolution," *Sensors*, vol. 9, no. 9, pp. 6869-6896, Aug 2009.

[14] W Jiang, H Jin, C Yu, and C Liu, "Introduction and Overview of Wireless Sensor Networks," in *Handbook of Research on Developments and Trends in Wireless Sensor Networks: From Principle to Practice.*: Information Science Reference, 2010, ch. 1, pp. 1-19.

[15] C Townsend, S Arms, and Inc. Microstrain, "Wireless Sensor Networks: Principles and Applications," in *Sensor Technology Handbook*.: Newnes, 2004, ch. 22, pp. 575-589.

[16] William Stalling, "Intoduction," in *Network Security Essentials*.: Peason Education Inc, 2011, pp. 15-40.

[17] J Liu, Y Xiao, S Li, W Liang, and C. Chen, "Cyber Security and Privacy Issues in Smart Grids," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 4, pp. 981-997, Jan 2012.

[18] Y Zhou, Y Fang, and Y Zhang, "Securing wireless sensor networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 3, pp. 6-28, Sep 2008.

[19] C Pfleeger and S Pfleeger, "I can't get no satisfaction," in *Analyzing Computer Security*. Michigan: Peason Education International, 2011, pp.

596-657.

[20] Y Wang, G Attebury, and B Ramamurthy, "A survey of security issues in wireless sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 8, no. 2, pp. 2-23, Feb 2007.

[21] Y Yan, Y Qian, H Sharif, and D Tipper, "A Survey on Cyber Security for Smart Grid Communications ," *Communications Surveys & Tutorials, IEEE* , vol. 14, no. 4, pp. 998 - 1010, January 2012.

[22] C Pfleeger and S Pfleeger, "Not All as It Seems," in *Analyzing Computer Security*. Michigan: Peason Education International, 2011, pp. 520-570.

[23] A Perrig, R Szewczyk, J Tygar, V Wen, and D Culler, "SPINS: Security protocols for sensor networks," *Wireless Networks*, vol. 8, no. 5, pp. 521-534, Sep 2002.

[24] T Bin, O Xi, L Dong, L Shoushan, and Y Yixian, "Study of Attacks and Countermeasures in Wireless Sensor Networks," *Advances in Information Sciences and Service Sciences*, vol. 8, no. 4, pp. 311-320, May 2012.

[25] H Khurana, M Hadley, Ning L, and D Frincke, "Smart-grid security issues," *Security & Privacy, IEEE*, vol. 8, no. 1, pp. 81-85, January 2010.

[26] M Ding, X Cheng, and G Xue, "Aggregation tree construction in sensor networks ," in *IEEE 58th Vehicular Technology Conference*, 2003, pp. 2168 - 2172.

[27] H Chan, A Perrig, and D Song, "Secure hierarchical in-network aggregation in sensor networks," in *ACM conference on Computer and communications security*, Alaxandria, VA, 2006, pp. 278 - 287.

[28] C Aguilar-Melchor, S Fau, F Fontaine, G Gogniat, and R Sirdey, "Recent Advances in homomorphic encryption," *IEEE SIGNAL PROCESSING MAGAZINE*, vol. 30, no. 2, pp. 108-117, Mar 2013.

[29] C Aguilar-Melchor, P Gaborit, and J Herranz, "Additively homomorphic encryption with d-operand multiplications," in *CRYPTO 2010: International Cryptology Conference*, 2010, pp. 138–154.

[30] A Abu-Mahfouz and G Hancke, "ns-2 extension to simulate localization system in wireless sensor networks," in *IEEE AFRICON*, 2011, pp. 1-7.

[31] C Pu and W Chung, "Group key update method for improving RC4 stream cipher in wireless sensor networks," in *International Conference on Convergence Information Technology*, Gyongju, 2007, pp. 1366-1371.

[32] J Domingo-Ferrer, "A Provably Secure Additive and Multiplicative Privacy Homomorphism," in *Proceeding ISC '02 Proceedings of the 5th International Conference on Information Security*, London, UK, 2002, pp. 471-483.

[33] T Burchfield, S Venkatesan, and D Weiner, "Maximizing throughput in ZigBee wireless networks through analysis, simulations and implementations," in *Proc. Int. Workshop Localized Algor. Protocols WSNs*, 2009, pp. 15-29.

[34] G Anastasi, M Conti, M Di Francesco, and A Passarella, "Energy conservation in wireless sensor networks: A survey," *Ad Hoc Networks*, vol. 7, no. 3, pp. 537–568, May 2009.