Trust and Energy-aware Routing Protocol for Wireless Sensor Networks

Laura Gheorghe, Răzvan Rughiniș, Nicolae Țăpuș
Department of Computer Science and Engineering
University Politehnica of Bucharest
Bucharest, Romania
{laura.gheorghe, razvan.rughinis, ntapus}@cs.pub.ro

Abstract - **Wireless Sensor Networks are composed of resource-constrained devices and are used in critical monitoring and tracking applications. Therefore, routing protocols for such networks should take into consideration the trustworthiness of and the energy available on sensor nodes. We developed TER - Trust and Energy-aware Routing protocol, a location-based, trust and energy-aware, routing protocol for Wireless Sensor Networks. The protocol uses distance, trust and energy as metrics when choosing the best path towards the destination. The protocol can be easily extended to support other metrics. We implemented our protocol in TinyOS and tested it in several test configurations. We determined experimentally that TER provides traffic load and energy balancing while building trustworthy paths.**

Keywords - Wireless Sensor Networks, routing protocol, trust, energy, security

I. INTRODUCTION

Wireless Sensor Networks (WSNs) represent an innovative technology used for monitoring specific environments. A WSN is composed of tens to thousands of sensor nodes, which are low-power, low-cost, small, resource-constrained devices. The sensor nodes collaborate in order to detect events that take place in the monitored environment and send relevant data to one or more base stations [1].

WSNs are used in critical applications like military surveillance, homeland security and medical monitoring, and, in these cases, protecting the network against malicious attacks is crucial. However, WSNs have unique characteristics: wireless transmission medium, limited resources available on sensor nodes, hostile environment, ad-hoc deployment, unreliable communication, and unattended operation. Therefore, protocols for critical sensor networks should be designed with security in mind, while taking into consideration their specific constraints and challenges.

For large sensor networks, multi-hop communication is more energy-efficient than single-hop communication. A routing protocol is used for assuring packet delivery and most network traffic has a many-to-one pattern because all nodes send data packets towards the base station.

Most routing protocols for sensor networks use a single metric to determine the best path to destination. Some use two metrics such as location and energy [2], [3], location and trust [4], or trust and link quality [5]. We identify a need for a routing framework that can be easily extended to support any metric.

In this paper, we propose a trust and energy-aware, location-based routing protocol called Trust and Energy-aware Routing (TER) protocol. TER uses trust values, energy levels and location information in order to determine the best paths towards a destination. The protocol achieves balancing of traffic load and energy, and generates trustworthy paths when taking into consideration all proposed metrics. Other metrics can be easily integrated in the protocol.

The protocol relies on the trust values provided by the Adaptive Trust Management Protocol (ATMP), which computes trust based on intrusion detection techniques [6]. However, TER can also use trust and reputation data provided by other trust management mechanisms.

The rest of the paper is structured as follows: Section II presents related work, Section III describes the protocol design, Section IV includes implementation details, Section V presents the experimental evaluation, and Section VI discusses conclusions and future work.

II. RELATED WORK

Based on the network structure, routing in Wireless Sensor Networks can be classified in flat-based, hierarchical-based and location-based routing [7]. Based on protocol operation, routing protocols can be classified in multi-path based, query-based, negotiation-based, QoS-based and coherent-based routing protocols.

Location-based routing protocols compute routing paths based on the location of nodes. Well known location-based protocols are: Geographic Adaptive Fidelity (GAF) [2] and Geographic and Energy Aware Routing (GEAR) [3].

Geographic Adaptive Fidelity (GAF) is an energy-aware, location-based routing algorithm [2]. Location information is used by each node to associate itself to a virtual grid. Nodes in the same grid square are equivalent in regard to packet forwarding and take turns in sleeping and being awake in order to load balance energy consumption. GAF relies on an underlying ad hoc routing protocol.

Geographic and Energy Aware Routing (GEAR) is an energy-aware and location based routing protocol [3]. The protocol selects the neighbor using an energy-aware and geographically informed algorithm to forward the packet towards the target region. Then, it uses a recursive geographical forwarding technique for disseminating the packet in the target region.

Two relevant routing protocols, which take into consideration trust values when determining the path to the

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-203-5

destination are elaborated in TRAP [4] and Zahariadis et al. [5].

TRAP is a trust-aware routing protocol, which represents a component of µRACER routing solution for Wireless Sensor Networks [4]. Each node considers the communication context when choosing the next hop. The communication context includes the past behavior of neighbor nodes and the quality of the links between the local node and the neighbors.

Zahariadis et al. propose the integration of a trust model with a location-based routing protocol [5]. A metric is computed using the distance of the neighbor node to the destination and the trust in the neighbor node. Therefore, the metric is maximized for trustworthy neighbors closer to the destination.

Our protocol is a location-based routing protocol, because it uses the location of neighbor nodes for determining the best path towards the destination. However, TER also considers trust and energy when determining the best next hop. In addition, the protocol uses trust values to determine whether to forward packets from specific nodes.

III. Protocol Design

In Wireless Sensor Networks, most network traffic is upstream traffic, with a many-to-one communication pattern because all packets must reach the base station. In this paper, we develop a method for performing trustworthy routing of upstream traffic.

Trust and Energy-aware Routing (TER) is a trust and energy-aware, location-based routing protocol for Wireless Sensor Networks. The trust values are obtained from Adaptive Trust Management Protocol (ATMP), which computes them based on intrusion detection techniques [6]. We use an extended version of ATMP, which delivers energy and location data along with the trust associations.

TER includes two phases: setup and forwarding. In the first phase, the best next hop towards the base station is selected by taking into consideration several factors, such as trust, energy and location. In the second phase, the packets generated by trustworthy nodes are forwarded using the selected next hop.

A. Assumptions and Notations

A WSN can be represented as a graph, like in Formulas 1, 2 and 3, where N_i are vertices which represent nodes in the sensor network and $\{N_i, N_j\}$ are edges which represent that two sensor nodes can communicate with each other directly.

$$WSN = (V, E) \quad (1)$$

$$V = \bigcup N_i \quad (2)$$

$$E = \bigcup \{N_i, N_j\} \quad (3)$$

The set of neighbors of a node is represented in Formula 4, where N_i is the local node and N_j is a neighbor node.

$$\mathrm{NB}(\mathrm{N_i}) = \{\bigcup N_j \mid N_j \in V \wedge \{N_i, N_j\} \in E\} \quad (4)$$

The sensor network may be placed in a harsh environment and operate unattended. An attacker may have physical access to the nodes and can compromise them.

We assume that each node knows its location and how much energy it has consumed at any moment. The localization algorithm or technology used for obtaining the location is out of scope for this paper.

We also assume that the Base Station (BS) has a fixed location. Each node knows the location of the BS. This information is distributed to all sensor nodes, during network initialization, along with the shared keys.

The TER assumes that ATMP is extended to send energy and location information along with the trust associations. Therefore, ATMP periodically sends update packets containing the trust associations, the consumed energy and location of the local node (the node sending the updates).

The trust associations (TA) are represented in Formula 5. It includes associations between the neighbors of the local node (n_i) and direct trust values (T_i) [6].

$$TA = [(n_1, T_1), (n_2, T_2), ..., (n_p, T_p)] \quad (5)$$

The update packet (UP) is represented in Formula 6, where E_l is the energy consumed by the local node and (x_l and y_l) are the coordinates of the local node.

$$UP = [TA, E_l, (x_l, y_l)] \quad (6)$$

ATMP takes the trust associations received from multiple neighbors and computes a final trust value. This value has a historical component (T_{old}), a direct (T_d) and an indirect component (T_i), as in Formula 7. The weights are allocated in regard to Formula 8. The final trust (T_{new}) is used in TER when computing the cost.

$$T_{new} = \alpha T_{old} + \beta T_d + \sum_{i=1}^{p} \gamma_i T_i \quad (7)$$

$$\alpha + \beta + \sum_{i=1}^{p} \gamma_i = 1 \quad (8)$$

A node is considered suspicious, if it has a trust value lower than a certain limit (SL), as in Formula 9.

$$Suspicious(N) = \begin{cases} 1 & if\ T(N) \geq SL \\ 0 & if\ T(N) < SL \end{cases} \quad (9)$$

The update packets are authenticated using a broadcast authentication mechanism such as µTESLA [8] in order to prevent malicious updates.

We assume that the parameters of TER and ATMP (weights, limits) can be modified during run-time through generic reconfiguration mechanisms.

The Setup Phase if performed periodically in order to update the costs. The period depends on the number of nodes, topology, mobility, application and security requirements. A large, dense network with mobile nodes, or a network exposed to threats should execute the Setup phase more often.

B. Setup Phase

In the Setup Phase, each node computes a cost for each of its neighbors. The neighbor with the lowest cost is subsequently chosen as the next hop on the route to the BS.

The cost takes into consideration the trust value provided by ATMP, the energy level available on the neighbor node, the distance from the local node to the neighbor node, and the distance from the neighbor node to the base station.

The cost for a neighbor node N is computed using Formula 10, where DT is the degree of distrust in the neighbor node N normalized by the largest distrust among all neighbors, E is the consumed energy of node N normalized by the largest consumed energy among all neighbors, DN represents the distance from the local node to node N normalized by the largest distance, DB represents the distance between N and the BS, normalized by the largest distance, and weights are allocated in regard to Formula 11.

$$C(N) = \alpha DT(N) + \beta E(N) + \gamma DN(N) + \delta DB(N) \quad (10)$$

$$\alpha + \beta + \gamma + \delta = 1 \quad (11)$$

The distrust (dt) is computed from the trust value generated by the ATMP (Formula 7) in regard to a specific neighbor, using Formula 12. In this formula, T signifies the trust value and MaxTrust is the maximum value for the trust parameter. The normalized value of distrust (DT) is computed and used in Formula 10.

$$dt(N) = MaxTrust - T(N) \quad (12)$$

The distance to a neighbor (dn) is computed using the coordinates of the local node and the ones of the neighbor node, as in Formula 13, where x_L and y_L are the coordinates of the local node, x_N and y_N are the coordinates of the neighbor node N. The distance is normalized (DN) and used when computing the cost.

$$dn(N) = \sqrt{(x_L - x_N)^2 + (y_L - y_N)^2} \quad (13)$$

In the same manner, the distance from the neighbor node and the BS (db) are computed, using Formula 14, where x_B and y_B are the coordinated of the BS, x_N and y_N are the coordinates of the neighbor node N. The normalized value (DB) is used when computing the cost.

$$db(N) = \sqrt{(x_N - x_B)^2 + (y_N - y_B)^2} \quad (14)$$

In the Setup Phase, the node computes the cost for each neighbor and chooses the neighbor with the lowest cost as next hop towards the BS. Formula 15 represents the next hop, where N_j is the neighbor node with the minimum cost, and nb is the number of neighbors.

$$\begin{aligned} NH(N_i) = \{ N_j \mid N_j \in NB(N_i) \wedge \\ C(N_j) = \min\{C(N_1), C(N_2), .., C(N_{nb})\}\} \end{aligned} \quad (15)$$

C. Forwarding Phase

In the Forwarding Phase, the node receives packets and forwards them towards the base station only if they are trustworthy. The trustworthiness of a packet is determined using Formula 16, where T is the trust in source node SN, TL is the minimum allowable trust limit and MAC is the Message Authentication Node.

$$Trustworthiness(P) = \begin{cases} 1 & if\ T(SN) \geq TL \wedge MAC(P)\ is\ valid \\ 0 & if\ T(SN) < TL \vee MAC(P)\ is\ invalid \end{cases} \quad (16)$$

If the packet cannot be authenticated (MAC) or if the source node has a trust value lower than the trust limit (TL), the packet is considered untrustworthy.

D. Design considerations

Most applications that use WSNs do not require reliable delivery. The use of acknowledgement considerably increases energy consumption. Therefore, we do not include an acknowledgement mechanism in TER. However, if the application does not tolerate packet loss, acknowledgements are easy to integrate with our protocol.

Duplicate detection is necessary in the case of routing loops. However, in order to detect duplicates, information about each packet has to be stored on the nodes. This has a considerable impact on memory usage. If the application requires duplicate detection, TER can be easily extended to support such feature.

IV. IMPLEMENTATION

The protocol has been developed in TinyOS [9], within a layer in the communication stack, between the Active Message and the Application layers. A nesC [10] component has been used for implementing the two phases of TER.

Because TinyOS is an event-based operating system, code is executed only when an event takes place. We have three types of events in TER: receive trust, location and energy data from ATMP, trigger timer, and receive packet. The flow of operations for the three types of events is represented in Figure 1.

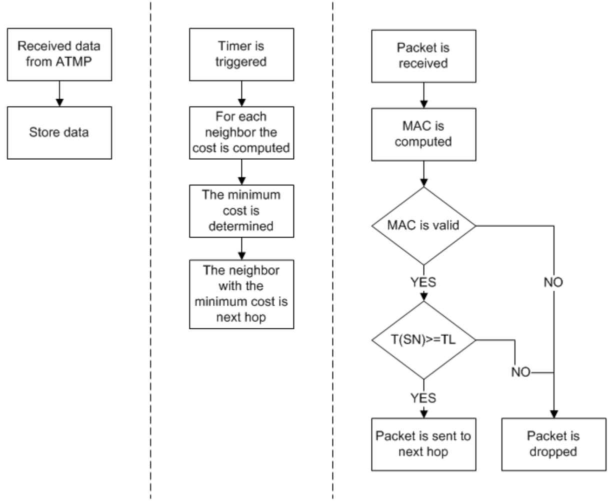
Received data from ATMP
Store data
Timer is triggered
For each neighbor the cost is computed
The minimum cost is determined
The neighbor with the minimum cost is next hop
Packet is received
MAC is computed
MAC is valid
YES
NO
T(SN)>=TL
NO
YES
Packet is sent to next hop
Packet is dropped

Figure 1. TER workflow

The TER component communicates with the ATMP component through an interface, in order to receive trust, energy and location information regarding the neighbor nodes. The ATMP component sends the data through a nesC event when it has obtained trust, energy and location information. The information is stored by the TER component.

A timer is used for periodically computing the cost using the information received from the ATMP component, according to Formula 10. The component determines the neighbor with the lowest cost and stores the identifier of the neighbor as next hop.

When a packet is received, the first step is to validate the MAC. If the MAC is invalid, the packet is considered untrustworthy and discarded. If the MAC is valid, the trust value for the source node is verified. If the trust value is below a certain accepted limit, the packet is considered untrustworthy and dropped. If the MAC is valid and the trust is above the accepted limit, the packet is forwarded trough the next hop.

V. Experimental Evaluation

The protocol has been evaluated experimentally using TOSSIM, a simulator for TinyOS [11]. TOSSIM captures the behavior of a large number of nodes at network bit granularity. Therefore, it is a reliable tool for evaluating the behavior of TER enabled nodes in different test cases.

We want to test our protocol in a realistic environment, in order to make sure it operates properly even in harsh conditions. We have therefore used TOSSIM to model an environment with interferences and signal attenuation, which causes a considerable amount of packet loss (30%) specific to harsh environments. The probability of packet loss is increased with the number of hops between source and destination. Therefore, longer paths cause a lower delivery rate.

The test scenario involves a network topology of 10 nodes and the Base Station (Node 0), as in Figure 2. For the analysis, we isolate the flow of packets generated by Node 7 destined to the Base Station.

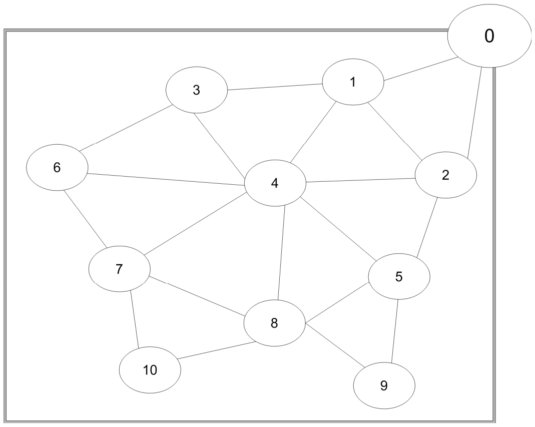

Figure 2. Scenario Topology

We analyze the behavior of TER in different test configurations - with different values for weights α, β, γ and δ. Table 1 includes the analyzed configurations and the values for the considered weights.

TABLE I. Test Configurations

	α	β	γ	δ
	Trust	Energy	Node-neighbor	Neighbor-BS
Configuration 1	0	0	0	1
Configuration 2	1	0	0	0
Configuration 3	0.3	0.3	0.1	0.3
Configuration 4	0.4	0.3	0.1	0.2
Configuration 5	0.3	0.4	0.1	0.2

We vary the weights from Formula 10 in order to determine the best routing behavior for the proposed scenario. This behavior is evaluated in regard to the number of packets routed through suspicious nodes and to energy consumption.

In Configuration 1, only the distance from the neighbor to the destination is considered, therefore the neighbor which is closer to the destination has the lowest cost. In Configuration 2, only trust is considered: the most trustworthy neighbor has the lowest cost. The first 2 configurations serve as benchmarks in order to determine the influence of a single metric on the packet flow.

The next 3 configurations take in consideration all the proposed metrics and they can be used to determine the most appropriate routing behavior for a specific situation. In Configuration 3, trust, energy and distance to BS have equal weights while the distance to neighbor has a lower weight. In Configuration 4, trust has highest weight, then energy and distance to BS, while the distance to neighbor has the lowest weight. Configuration 5 is similar to Configuration 4 but the energy has the highest weight.

We evaluate the routing behavior of sensor nodes by considering a particular scenario with two suspicious nodes: the trust in Node 4 is 60%, the trust in Node 10 is 40%, the Trust Limit is 50%, and the Suspicious Limit is 80%. This implies that Node 10 sends untrustworthy packets, which will not be forwarded by other nodes.

For each configuration specified in Table 1, we evaluate the routing behavior when delivering a large number of packets generated by Node 7 and destined to the BS. We ran each test 20 times and computed the average values for routed packets and energy consumption, for each considered configuration.

A. Routed Packets

A way of evaluating routing behavior is through the number of packets routed by each node. From the results, we can determine which are the most used paths for each configuration, and whether the suspicious nodes are effectively avoided.

The number of routed packets per node, in each configuration, is represented in Figure 3. An average number of 370 packets are sent by the source node 7, as it can be observed in the figure.

For Configuration 1, all packets take the route [7, 4, 1, 0]. This is the best path when taking in consideration the distance between the neighbor and the destination. An average number of 334.6 packets are delivered through suspicious node 4 (all packets which are not lost on the link between Node 7 and Node 4), but no packet is delivered through suspicious node 10. For this specific topology, the algorithm chooses an efficient path but routes through an untrustworthy node.

In Configuration 2, all packets follow the route [7, 6, 3, 1, 0], the first best path when taking into consideration the trust values. No packet is delivered through the suspicious nodes. For this specific topology, all packets, which are not lost during transmission, reach the base station. This is because the algorithm chooses the first trustworthy next-hop which happens to be placed in the direction of the base station. In other topologies, it is possible that the algorithm does not pick a neighbor in the right direction; in such a case, the paths would be longer and more packets would be lost during transmission.

Because of the greedy algorithm implemented by TER, trust or energy cannot be used as single metric when computing the cost. Therefore, it is better to use these metrics in combination with location.

In Configuration 3, the traffic load is more balanced. The paths that are used for packet delivery are: [7, 8, 5, 2, 0], [7, 6, 4, 1, 0], [7, 6, 3, 1, 0], [7, 8, 9, 5, 2, 0]. The average number of packets delivered through node 4 is 0.65, which is very low. No packets are sent through suspicious node 10. The configuration provides a very good load balancing, as it uses 4 paths to the destination and a very small number of packets are delivered through suspicious nodes.

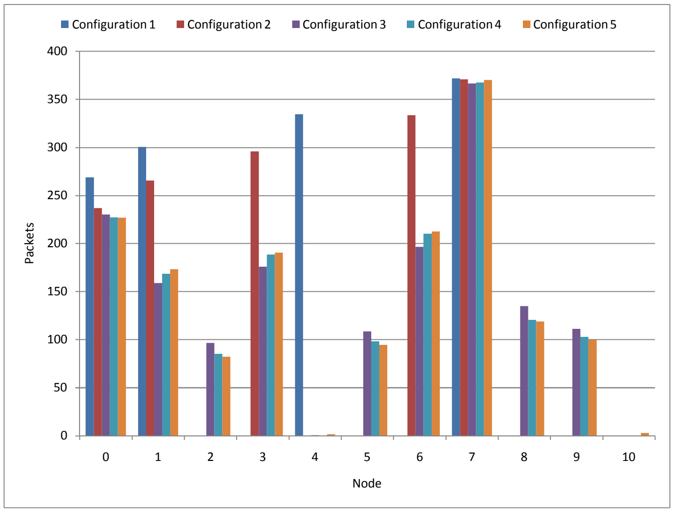

Figure 3. Routed Packets per Node

In Configuration 4, packets are delivered through paths: [7, 8, 5, 2, 0], [7, 6, 3, 1, 0], [7, 8, 9, 5, 2, 0]. No packet is delivered through suspicious nodes 4 and 10. A good load balancing is assured in this configuration and suspicious nodes are avoided.

In Configuration 5, several paths are used for packet delivery: [7, 8, 5, 2, 0], [7, 6, 3, 1, 0], [7, 8, 9, 5, 2, 0]. Some routing loops are generated: [7, 10, 8], [7, 4, 5, 9, 8]. The average number of routed packets through node 4 is 1.65 and through node 10 is 3. The configuration has a good load balancing but it may produce routing loops and a small number of packets are delivered through suspicious nodes.

When analyzing the packets' paths, we determine that Configurations 3 and 4 are the best for this scenario because they have good load balancing, do not create routing loops, and avoid suspicious nodes.

B. *Energy consumption*

Another way of evaluating routing behavior is the energy consumed while routing data packets. We wish to determine whether energy consumption is well balanced between the nodes. The energy metric has an important role in balancing consumption. Without the energy metric, the packets would take the same path and deplete the energy of the nodes on that path.

We evaluate the energy consumption necessary for routing 300 packets generated by Node 7 and destined to the BS. The energy consumed with routing data packets towards the destination, on every node, in each configuration, is represented in Figure 4. The values are represented in Joules. A sensor node has two alkaline AA batteries, each with 9360 J energy. Most energy is consumed with sending and receiving packets. Amiri determined experimentally that a byte sent or received by CC2420 radio consumes 0.12mJ [12].

From Figure 4, we can determine whether energy consumption is balanced between the nodes and if suspicious nodes have been avoided.

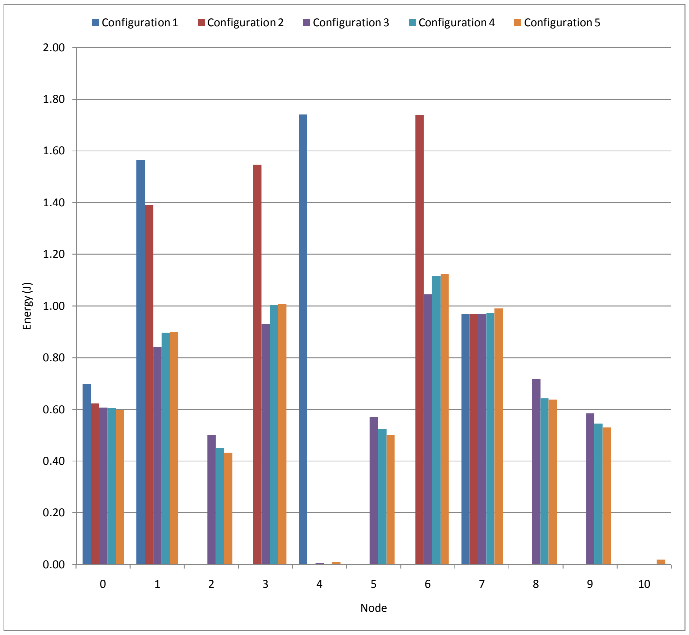

Figure 4. Energy Consumption per Node

In Configuration 1, the most energy is consumed on the suspicious node 4. Energy consumption is not well balanced, as nodes 2, 3, 5, 6, 8 and 9 have no energy consumption due to packet delivery. Node 7 and node 0 have lower energy consumption than nodes 4 and 1 because they either only transmit or only receive data packets. The energy consumption on nodes 4 and 1 doubles because they transmit and receive packets. Overall, the configuration does not have a balanced energy consumption and routes through suspicious nodes.

In Configuration 2, the suspicious node is avoided, but energy consumption is still not so well balanced, because nodes 2, 5, 8 and 9 are not delivering any packets. The energy consumption drops from 1.74 J on node 4, to 1.55 J on node 3 and to 1.39 J on node 1 because of packet loss. Packets are lost during transmission, so less packets are routed by the subsequent nodes.

In Configurations 3 and 4, energy consumption is well balanced in the network and there is no energy consumption on the suspicious nodes. Configuration 5 is also well balanced and has low energy consumption on the suspicious nodes. These 3 configurations are the best from the point of view of balancing energy consumption due to data packet delivery.

The total energy, consumed on all nodes while delivering 300 data packets generated by Node 7, is represented in Figure 5. The least energy is consumed in Configuration 1 because the protocol determines the shortest path to the destination. Similar energy consumptions have been determined for configurations 3, 4, and 5, which try to determine the shortest path while avoiding suspicious nodes and balancing energy consumption. Configuration 2 has lower energy consumption but it is not well balanced throughout the network.

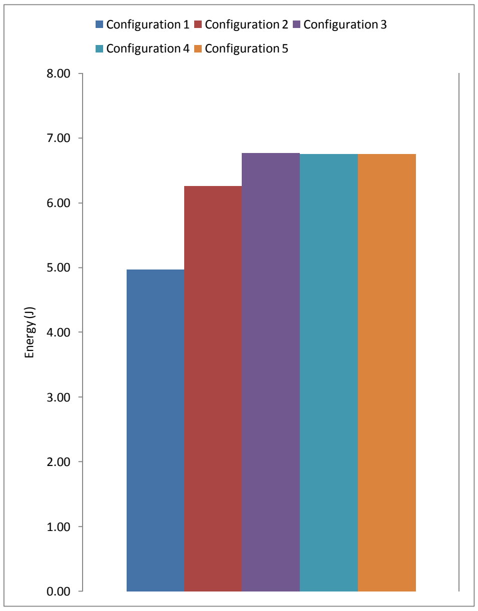

Figure 5. Total Energy Consumption

The energy metric imposes an energy cost but at the same time it allows for a good balancing of energy consumption (see Figure 4), which is an important aspect for Wireless Sensor Networks. The energy metric is particularly important if there is no redundancy among nodes concerning the transmitted information, and therefore we aim to avoid the energy depletion of the nodes which may be preferred by the routing protocol due to their position.

C. Discussion

Configuration 1 does not take into consideration neither energy nor trust, including only the distance from the neighbor to the base station. Therefore, it may route packets through nodes which are untrustworthy or have low energy. It only guarantees that it chooses the shortest path towards the destination, as it was observed experimentally. The shortest path consumes the least total energy on sensor nodes but it does not avoid nodes with low power. If the nodes are not mobile, the path is used until some of the nodes die and another path has to be chosen. On the long term, this strategy may determine the partitioning the network. This configuration does not provide load balancing of traffic, is not trustworthy and does not have a balanced energy consumption.

Although Configuration 2 generates trustworthy paths, these paths can be long and inefficient in some cases because the algorithm does not take location into consideration. The only guarantee is that it chooses trustworthy paths. If nodes are not mobile and if the trust values do not change, the algorithm chooses the same path and it consumes all nodes' energy on the path. It does not provide load balancing, it does not guarantee that an efficient path is chosen, and it does not balance energy consumption.

Configuration 3 has a very good load balancing of network traffic, delivers a small number of packets through suspicious nodes and balances energy consumption.

Configuration 4 performs load balancing of network traffic, selects trustworthy and short paths, and balances energy consumption on sensor nodes. Trust has a greater weight and this explains the minimum amount of packets routed through suspicious nodes.

Configuration 5 performs load balancing for network traffic, balances energy consumption, but routes through suspicious nodes, and generates routing loops.

The last three configurations have similar total energy consumption, provide load balancing of traffic, balancing of energy consumption, and they avoid suspicious nodes. From these configurations, Configuration 3 is preferable insofar it has the best load balancing of network traffic and Configuration 4 is preferable insofar it has the minimum number of packets delivered through suspicious nodes.

VI. CONCLUSION AND FUTURE WORK

Wireless Sensor Networks that are used for deploying critical applications such as military surveillance or medical monitoring should provide a high level of security and trustworthiness. Therefore, routing protocols for WSNs should to be designed with security in mind, taking into account multiple metrics that support network availability.

We developed Trust and Energy-aware Routing protocol, which is a location-based, trust and energy-aware routing protocol for sensor networks. The protocol is based on the

Adaptive Trust Management Protocol, which computes trust values based on node behavior.

The protocol uses several metrics: trust values, energy levels, the distance between the local and the neighbor node and the distance between the neighbor node and the destination. These metrics may have different weights when computing the cost of routing a packet through a specific neighbor. The cost is computed based on the metrics and their weights. The neighbor with the lowest cost is chosen as next hop towards the base station.

Trust and Energy-aware Routing protocol has two phases: the Setup and the Forwarding phase. In the Setup phase, the next hop is determined, and in the Forwarding phase, the packets generated by a trustworthy source are forwarded and the others are dropped.

We have implemented the protocol in TinyOS and we have evaluated it experimentally using TOSSIM, comparing 5 protocol configurations for the same scenario. Each configuration has different weights for the trust, energy and distance metrics. For each configuration, the routing behavior has been examined in regard to the paths and packets routed through each node, the consumed energy, and the effectiveness of packet delivery.

Traffic load and energy balancing are very important in Wireless Sensor Networks. In relation to other routing protocols, TER achieves a good balancing of load and energy and generates trustworthy paths, when taking into consideration all proposed metrics: trust, energy and distance.

As future work, we plan to extend the protocol to include other metrics, such as link quality, and to support adaptive weights, allowing, for example, the weight for energy to increase over time. Other extensions we want to implement are duplicate detection and acknowledgements.

We also want to integrate our protocol with another trust mechanism. In addition, we wish to evaluate the protocol in a larger, real-world network.

ACKNOWLEDGEMENTS

This work has been partially funded by the European Commission under grant agreement FP7-ICT-258280 TWISNet project, partially by Google Inc., University Relations and partially by the Sectoral Operational Programme Human Resources Development 2007-2013 of the Romanian Ministry of Labour, Family and Social Protection through the Financial Agreement POSDRU/88/1.5/S/60203.

REFERENCES

[1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," Computer Networks, 2008, vol. 52, no. 12, pp. 2292-2330.

[2] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed Energy Conservation for Ad Hoc Routing," in ACM/IEEE International Conference on Mobile Computing and Networking, 2001, pp. 70-84.

[3] Y. Yu, D. Estrin, and R. Govindan, "Geographical and Energy-Aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks," 2001.

[4] A. Rezgui and M. Eltoweissy, "μRACER: A Reliable Adaptive Service-Driven Efficient Routing Protocol Suite for Sensor-Actuator Networks," IEEE Transactions on Parallel and Distributed Systems, 2009, vol. 20, no. 5, pp. 607-622.

[5] T. Zahariadis, P. Trakadas, H. Leligou, P. Karkazis, and S. Voliotis, "Implementing a Trust-Aware Routing Protocol in Wireless Sensor Nodes," Developments in E-systems Engineering, 2010, pp. 47-52.

[6] L. Gheorghe, R. Rughiniş, R. Deaconescu, and N. Ţăpuş, "Adaptive Trust Management Protocol Based on Fault Detection for Wireless Sensor Networks," in The Second International Conferences on Advanced Service Computing, 2010, pp. 216-221.

[7] J. Al-Karaki and A. Kamal, "Routing techniques in wireless sensor networks: a survey," IEEE wireless communications, 2004, pp. 1-37.

[8] M. Luk, A. Perrig, and B. Whillock, "Seven cardinal properties of sensor network broadcast authentication," in Proceedings of the fourth ACM workshop on Security of ad hoc and sensor networks - SASN '06, 2006, pp. 1-10.

[9] P. Levis et al., "TinyOS: An operating system for sensor networks," Ambient Intelligence, 2004, pp. 115-148.

[10] D. Gay, P. Levis, R. Von Behren, M. Welsh, E. Brewer, and D. Culler, "The nesC language: A holistic approach to networked embedded systems," in Proceedings of the ACM SIGPLAN 2003 conference on Programming language design and implementation, 2003, vol. 35, no. 11, pp. 1-11.

[11] P. Levis, N. Lee, M. Welsh, and D. Culler, "TOSSIM: accurate and scalable simulation of entire TinyOS applications," in Proceedings of the first international conference on Embedded networked sensor systems - SenSys '03, 2003, pp. 126-137.

[12] M. Amiri, "Wireless sensor networks: Evaluation of power consumption and lifetime bounds," LAP LAMBERT Academic Publishing, 2011, pp. 1-60.