SecRoute: End-to-End Secure Communications for Wireless Ad-hoc Networks

Protecting the Railway infrastructure, as a case study on Cyber-Physical Systems

George Hatzivasilis, Ioannis Papaefstathiou
Dept. of Electrical & Computer Engineering
Technical University of Crete
Chania, Crete, Greece
gchatzivasilis@isc.tuc.gr, ygp@mhl.tuc.gr

Konstantinos Fysarakis, Ioannis Askoxylakis
Institute of Computer Science
Foundation for Research and Technology Hellas
Heraklion, Crete, Greece
{kfysarakis. asko}@ics.forth.gr

***Abstract*—Railways constitute a main means of mass transportation, used by public, private, and military entities to traverse long distances every day. Railway control software must collect spatial information and effectively manage these systems. Wireless sensor networks (WSNs) are an attractive solution to cover the area along-side the railway routes. In-carriage WSNs are also studied in cases of dangerous cargo transportation. The secure communication of all these devices becomes important as successful attacks can harm the railway's business operation or cause serious injuries and deaths. This paper presents SecRoute – an end-to-end secure communications scheme for wireless ad hoc networks. The scheme implements mechanisms for cryptographic communication, trusted-based routing, and policy-based access control. SecRoute and alternative schemes are modelled on the NS-2 network simulator and a comparative analysis is conducted, indicating that the proposed scheme provides enhanced protection. A proof of concept of SecRoute is deployed on real embedded platforms and exhibits good overall performance, demonstrating that attacks on the route and carriage WSNs are effectively countered.**

Keywords— Cyber-physical systems; CPS; Internet of Things; IoT; access control; secure routing; trust; WSN; embedded devices;

I. INTRODUCTION

The evolution of the Internet of Things (IoT) motivates the deployment of intelligent cyber-physical systems (CPS) that utilize wireless networking. Railway settings are such intelligent systems that include urban, industrial and military railways. Wireless sensor networks (WSNs) can cover large operational area, processing ambient information, and, thus, an attractive solution in the context of such CPS applications. In wireless ad-hoc networks each entity relies on its neighbors to carry its messages and communicate with all participants. Several setups can be deployed, ranging from small to large scale installations that interact with environmental parameters [1]. Popular applications include, e.g. railways, smart cars, and intelligent buildings.

Due to the open medium and the dynamic entry of new nodes, routing protocols establish trust relationships to avoid malicious nodes. Reputation- and trust-based schemes are used in wireless networking to provide secure routing functionality [2]. Reputation is formed by a node's past behavior and reveals its cooperativeness; a node with high reputation is considered trustworthy and legitimate nodes rely on trustworthy entities to accomplish communication tasks. Low reputation reveals selfish or malicious entities; legitimate nodes avoid disreputable entities and do not serve their traffic. Authentication is also important to avoid impersonation attacks. The mainstream cryptographic solutions that are based on asymmetric cryptography require significant resources and are not always applicable in the WSN domain. Broadcast authentication protocols [3] are based on symmetric cryptography and, thus, are more appropriate for sensors and embedded systems. Moreover, authorization issues must be catered for. Authorization based on policies is efficiently implemented on embedded devices [4] and eases the operational management of a system.

Motivated by the above, this work introduces SecRoute – a novel system for secure routing and authorization in wireless ad-hoc networks, designed around the intricacies of CPS applications. The proposed system consists of three main components: (a) A cryptographic service that enforces authentication, message integrity, and confidentiality; (b) An efficient trust-based routing scheme that protects the communication against ad-hoc routing attacks; (c) A policy-based access control framework that provides authorization. As a proof of concept, SecRoute is applied on a railway scenario to safeguard two distinct WSNs: the first outdoor network monitors the railway's routes, while the second network is located inside a train's carriage that transports dangerous cargo. A preliminary version of this approach was implemented and demonstrated in the European project nSHIELD.

This work is organized as follows: In section 2, we refer to attacks on routing protocols and trust systems. In section 3, we indicate related studies on trust schemes for secure routing, and in section 4, we present our approach; SecRoute. In section 5, we demonstrate and analyze the security and performance of the system in Network Simulator 2 (NS-2) in comparison with the related schemes. In section 6, SecRoute is applied on embedded devices that implement a railway monitoring application, while section 7 features the concluding remarks.

II. ATTACKS & COUNTERMEASURES

Routing protocols can fail to protect legitimate nodes against selfish and malicious activity. Selfish entities avoid consuming their resources to serve other participants, or have excessively more demands than others, discouraging legitimate entities from acting in a trustworthy manner too. Malicious entities modify or stop the transmitted data; thereby, legitimate entities cannot communicate and abandon the network. Trust systems are integrated with routing protocols as a defense mechanism, but the trust schemes can also become the target of more sophisticated attacks.

A. Attacks on routing

A review of routing attacks in mobile ad hoc networks is presented in [6]. In *flooding*, an attacker exhausts the resources of the network by sending many requests in a short period of time. In *blackhole* and *link spoofing*, an attacker advertises fake routing information. In *colluding misrelay*, a pair of neighboring attackers collude and avoid participating in a route. In *wormhole*, a pair of attackers, which communicate through a private high speed network, collude to record packets at one location and replay them at another location of the network.

SecRoute prevents flooding attacks by rating the excessive load added by a participant. Blackhole and link spoofing attacks are countered as SecRoute detects and negatively ranks the misuse and discarding of a packet. If this rank goes beyond a threshold, the entity is considered malicious and is excluded from routing. For colluding misrelay, as the attackers do not participate in routes as intermediate nodes, they do not gain positive ratings for forwarding traffic. Thus, they are not able to make requests either. SecRoute forces the nodes to at least partially cooperate to avoid isolation. Trust schemes in general do not deal with wormhole attacks. Defense mechanisms figure out the true distance between two nodes. Nevertheless, such mechanisms can be integrated into a trust scheme.

B. Attacks on trust systems

Trust systems provide protection against the main routing attacks, but the trust evaluation process is then becoming the new target of attacks. The European Network and Information Security Agency (ENISA) investigated attacks on trust systems [7]. *Identity-based* attacks are related with impersonation and identity theft. *Ballot-based* attacks exploit the rating mechanisms to increase or decrease the trust and reputation of an entity. *Social-based* attacks are related with the social behavior of the participants; an attacker can follow a conflicting behavior towards different nodes to make them lower their trust values for each other. *Topology-based* attacks consider the topology of the network and the components of the system that manage trust.

SecRoute deploys broadcast authentication that prevent identity-based attacks. Ballot-based vulnerabilities are countered by a robust reasoning process that estimates the trust value of an examined node. The recommendation operation, where nodes can make notifications about other participants, block the social-based threats. The impact of topology-based attacks is limited, as the falsely accused legitimate nodes re-enter the network after some time-interval.

III. RELATED SCHEMES

The Reputation based Framework for Sensor Networks (RFSN) [8] uses a beta Bayesian formulation to implement fading. The node's reputation evaluates both its routing cooperativeness and the reported sensed variables (e.g. temperature). Tools from statistics are integrated to capture occasional failure. Direct interaction is considered more important and indirect knowledge is processed by a belief discounting mechanism to prevent attacks by recommenders. For node authentication, RFSN uses μ-TESLA [8] broadcast authentication.

Ariadne [9] protects the network against malicious behavior. The scheme relies on feedback about which packets were successfully delivered to select the most efficient path to a destination. If a small fraction of packets is eventually delivered, the path will be avoided in future interactions. The scheme does not report malicious nodes to avoid badmouthing. Node authentication is based on TESLA.

The Cooperative Secure routing protocol based on ARAN (CSRAN) [10] uses trust and reputation to protect the routing operation upon the routing protocol ARAN [11]. For the reputation calculation, a Bayesian procedure performs reputation fading. If a node infers that the next hop in the route is malicious, it will automatically re-route the communication from that point. Thus, the network's performance is retained as fewer failures occur. However, the re-routing process can be exploited by attackers that arbitrarily re-route traffic to specific nodes. A node may also send first hand reputation values as recommendations to neighboring nodes. ARAN integrates routing authentication based on certificates.

The Secure Resilient Reputation-based Routing (SR3) [12] combines a reinforced random walk algorithm with reputation. The node count under this random walk is higher than in the rest of the approaches. If a valid acknowledgement is received, the communication is considered successful and the reputation on the path nodes is increased. No indirect knowledge is processed and the reputation values are maintained in a FIFO finite list, accomplishing fading. Lightweight cryptography is integrated for confidentiality, integrity and authentication.

SecRoute adopts efficient and state-of-the-art features for trust. It also enforces authentication and integrity checks as the rest schemes. Nonetheless, after successfully authenticating a node and validating the fair use of the network, authorization is required to perform the requested activity. The combination of the three services is a novel approach to provide an effective overall protection.

IV. THE SECROUTE SCHEME

A. Cryptographic Service

As the network nodes start communicating, the need to authenticate the sender of an incoming message becomes imperative. Broadcast authentication can overcome the computational (i.e. asymmetric cryptography) and operational

(e.g. key distribution) issues and the Timed Efficient Stream Loss Tolerant Authentication (TESLA) broadcast authentication protocol [3] is a common choice. It is based on loose time synchronization between the sender and the receiver. TESLA achieves the security properties of asymmetric cryptography by using keyed Message Authentication Code (MAC) functions. The sender attaches a MAC to each packet where the key is known only to itself. The receiver stores the packet without being able to authenticate it at that moment; shortly after, the sender discloses the key and the receiver can then authenticate the packet. TESLA is efficient, with low communication and computation overhead, while it scales to large networks with many nodes and tolerates packet loss, as demonstrated in the original paper.

For the purposes of this work, the Ultra-Lightweight Cryptographic Library (ULCL) [13] was used to implement TESLA, providing node authentication and message integrity. In the used testbed (presented in following sections), packet data sizes vary between 28 to 364 bytes. SHA-256 is applied for TELSA's MAC computations, taking 0.29μs to 3.8μs, respectively, to verify a message on a BeagleBone embedded device (ARM Cortex-A8 500-700MHz CPU, 256MB RAM, Linux OS). If a key distribution method is deployed, SecRoute can also achieve message confidentiality by encrypting the data segment of a packet; for this, AES-256 is used, taking 0.4μs to 5.2μs to encrypt the data.

The authentication for RFSN and Ariadne is similar. CSRAN uses certificates. RSA is utilized for the asymmetric encryption. On a Mobile Pentium III (Intel PIII 750/600 MHz CPU, 128MB RAM, Linux OS), the additional overhead for the digital signature generation at the sender required 8.5ms and the verification operation at the receiver took 0.5ms. A transmitted message of SR3 contains the hash of a random nonce encrypted with the message. If we utilize the same assumptions and primitives as for SecRoute, the cryptographic processing of a message only takes from 0.79μs to 5.5μs [11], depending on data size.

The broadcast authentication operation of RFSN, Ariadne and SecRoute is efficient with low added overhead. CSRAN and SR3 require a key distribution mechanism to provide the cryptographic service that further increases the computational resources. All three approaches provide security proofs.

B. Trust-based Routing

After authenticating the transmitted messages, nodes' participation in network is rated to protect the system against routing attacks. We implement a trust-based scheme for the network layer defense. The constant coefficients that are reported below are based on a previous security analysis conducted by the authors under the same assumptions [14].

A node continuously receives new pieces of knowledge either from its direct interaction with its neighbors or from the indirect notifications of other participants. Two relevant evaluation operations are implemented, for direct and indirect knowledge respectively. The evaluation outcome updates the reputation and trust values of an examined node. If the reputation or trust levels of a node decrease, notifications are automatically sent. At first the node is categorized as suspicious. A warning message is sent to the misbehaving node. If the node continues misbehaving, it is categorized as malicious. The legitimate node informs the rest of the participants. An entity can also counsel its trusted nodes by requesting notifications about a specific node.

For direct knowledge, grading determines the reputation value of an examined transaction. Our scheme's gradual grading policy assigns *1* to success and *-4* to failure. Gradual grading promptly detects misbehavior and efficiently counters threats where malicious entities gain high reputation to perform more successful attacks later. However, failures can still occur during normal operation due to traffic congestion. Congestion windows [15] is a fault-tolerance mechanism that protects the nodes' reputation during periods of traffic congestion. Then, the grading policy becomes tolerant and assigns a smaller negative value for a small number of failures. The reputation calculation determines the reputation value of an examined entity for a specific operation. The reputation fading maintains a small history of the latest grading values; values are weighted with a smaller value as time progresses and, thus, reputation fades, indicating that most recent values are more important. SecRoute implements the beta distribution that is described in [10] for fading, while the reputation values are statistically normalized prior to fading. The evaluation scope of direct knowledge indicates which nodes are rated after the transaction evaluation, but as it is not always feasible to detect a misbehaving node, SecRoute evaluates the performance of the whole path, similarly to the rest of the related schemes. Legitimates nodes retain their reputation by their participation on legitimate paths. Malicious nodes are eventually detected as they keep filling negative ranks.

The trust value of a node is the aggregation of the relevant reputation values for the three operations of forwarding, routing, and making recommendations. Forwarding is commonly considered the most important operation for the network's performance and is the most frequent one, thus has higher weight. The direct trust value is calculated as the weighted summation of the three reputation values (*60%* forwarding, *20%* routing, *20%* making recommendations). For indirect knowledge, notifications can contain positive or negative counsels (the trust value of the recommender). However, positive recommendations can be exploited by ballot-staffing attacks while negative notifications are usually exploited for badmouthing. The solution is a robust indirect knowledge evaluation which considers direct interaction as the more significant parameter for determining the trust level. The indirect trust value for a node is the weighted summation of the relevant recommendations. The deviation test checks if the received notifications deviate significantly from the node's direct opinion. If the deviation is more than *30%*, the notification is discarded and the recommender is ranked negatively. Moreover, notifications that are sent by trusted nodes gain higher weight (*+20%*). The final total trust for a node is the weighted summation of the direct and indirect trust values. As direct interaction is more important, it gains a weight of *80%* while the indirect trust is weighted with *20%*.

To categorize a node, SecRoute applies the following thresholds: *40* for trusted nodes, *30* for legitimate nodes, *-10* for suspicious nodes, and *-20* for malicious nodes. The

punishment for the malicious participant is the termination of packet forwarding for traffic that it is originated by this node and its exclusion from the routing operation. However, legitimate entities can be also categorized as malicious due to occasional malfunctioning. Thus, re-entrance strategies allow a punished node to re-enter the network with a default value after some time. For SecRoute, the initial and default neutral trust value for a newly entered entity is *0*. Finally, the path selection process indicates the criteria for deciding which candidate path to choose during the routing; SecRoute selects the shortest of the more well-reputed paths.

C. Policy-Based Access Control

After securing the routing operation and enhancing the correct transmission of requests and responses, there is the need to provide authorization functionality. As nodes receive requests they must decide if the requested actions can be made.

A Policy-Based Access Control (PBAC) framework [4][5] manages direct access requests to the resources of an embedded device, based on a predefined set of rules and policies. The solution is based on the eXtensible Access control Markup Language (XACML) policies and the Device Profile for Web Services (DPWS) standard for device discovery and message exchange. The framework consists of the following four components. The *Policy Enforcement Point (PEP)* performs access control by making decision requests and enforcing authorization decisions. Typically, this component runs on any node with accessible resources and services. The *Policy Decision Point (PDP)* evaluates requests against applicable policies and renders an authorization decision. The component lays on a trusted node with sufficient resources to parse applicable policies and make the decisions. The *Policy Information Point (PIP)* and *Policy Administrator Point (PAP)* act as a source of attribute values and are used for creating and managing policies or policy sets.

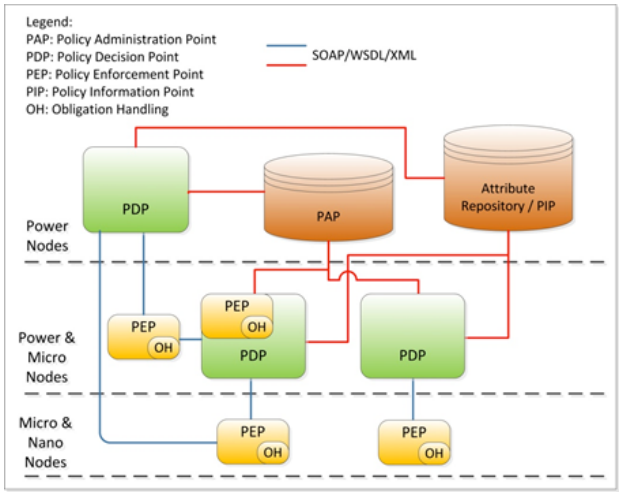

Fig. 1. The PBAC architecture.

A node might include one or more of these functional components, depending on its computational capabilities. The component interconnection is illustrated in Figure 1. By combining the above technologies, PBAC enables the fine-grained, policy-based control of resources from remote locations (e.g. cameras, sensors, and control stations). Based on the active policy, the framework allows for accessing the provided resources (e.g. video stream or sensor data), updating settings or even receiving alerts (e.g. in case of emergency, like blocked routes or train crashes).

V. Simulation Analysis

The trust schemes of SecRoute, RFSN, Ariadne, CSRAN, and SR3 are modelled in the NS-2 to analyze their routing behavior. SecRoute, RFSN, and Ariadne are integrated with the Dynamic Source Routing (DSR) protocol. A WSN with 50 nodes over an area of 350m × 350m is modelled. The IEEE 802.11 Distributed Coordination Function (DCF) is applied for the Media Access Control (MAC) layer and the two-ray ground reflection model is chosen for propagation. Nodes have a raw bandwidth of 2Mbps with a physical radio range of 100m. Each experiment consists of two phases that last 1 minute. At initialization, the nodes start with default trust values. At evaluation, three attack scenarios are monitored, measuring performance and security. In both phases, 10 sender nodes on one end of the WSN send 1kB data with Constant Bit Rate (CBR) to 10 receiver nodes at the other end of the WSN. Malicious nodes are introduced in the WSN to perform three types of attacks. In each attack, 5 experiments are performed per system, assuming 10, 20, 30, 40, and 50 percent of malicious nodes respectively. Each experiment is executed 10 times and the average metrics are reported. At each iteration, the malicious nodes are assigned randomly. For the first evaluated system, the attackers of each iteration are recorded to evaluate the same setting in all cases. Figure 2 depicts the evaluation results (analyzed in the paragraphs below).

In the first attack scenario, the malicious nodes exhibit conflicting behavior to confuse the trust mechanisms and then perform blackhole attacks, which is their main goal. Thus, at the initialization, the malicious nodes cooperate to gain legitimate trust levels, while at the evaluation phase, they start discarding traffic. The performance of each system is measured by the delivery ratio as the percentage of the packets that were sent successfully (Figure 2.A). The higher the value, the better.

The evaluation of DSR reveals the effectiveness of the attack on an unprotected system. Performance worsens as the malicious nodes increase. Ariadne also performs poorly due to the simple evaluation process and slow adaptability. SR3 detects malfunctioning faster, but the lack of recommendations leads each of source nodes to come across almost every malicious node. CSRAN, RFSN, and SecRoute discover the attackers even faster, due to the reputation fading feature. CSRAN performs well, as the legitimate nodes that detect the blackholes automatically re-route communications. However, no indirect knowledge is used and the malicious nodes continue malfunctioning until detected by all neighboring nodes. SecRoute is slightly better than RFSN, achieving the best results due to the more robust reasoning for evaluating direct knowledge and making recommendations.

In the second attack, the malicious nodes collude and exploit the recommendation processes; the goal is to take control of the path selection and include compromised nodes into it. The malicious nodes cooperate at the initialization

phase to gain high trust values for the three evaluated operations. Then, at the evaluation phase, they perform ballot stuffing attacks (advertising good recommendations for other malicious nodes) and badmouthing attacks (advertising bad recommendations for the legitimate nodes).

The success of the ballot-based attacks is assessed by the recommendation types assigned by the evaluated system. CSRAN, RFSN and SecRoute utilize negative and positive recommendations. Ariadne and SR3 are based on acknowledgements to verify a successful transmission. In that cases, the attackers drop the acknowledgement in paths with more that 50% of legitimate nodes. Thus, a high rate of legitimate nodes is falsely accused, while paths with many malicious nodes appear reliable. DSR is not evaluated in this scenario, as recommendations are not used in it. The systems are assessed based on the link-spoofing effect (Figure 2.B). It is calculated as the percentage of the total transactions that were routed through paths that contain at least one attacker because of the ballot-based attacks. The lower the value, the better.

Ariadne and SR3 provide little protection, while CSRAN performs well. RFSN is the best among the examined previously-proposed systems. SecRoute implements an even more robust recommendation evaluation mechanism and rates the recommenders against direct interaction. It achieves the best result, as it stops the link-spoofing attack once the bad recommenders are detected and punished.

In the final case, the malicious nodes take advantage of congestion periods and the WSN's trusted topology to make legitimate nodes unavailable via flooding. The attackers cooperate at initialization and then perform a flooding attack to every legitimate node that is detected as overloaded or is significant for the topology (e.g. highly trusted nodes). A flooding attack is a burst of data (10kB) sent from one attacker to another colluding node through the targeted legitimate nodes. The goal is to overburden these legitimate nodes and make them misbehave, harming their reputation. To investigate the ability of the systems to counter the attack, we measure the number of the executed flooding attacks (Figure 2.C). A high volume reveals that many nodes were found vulnerable.

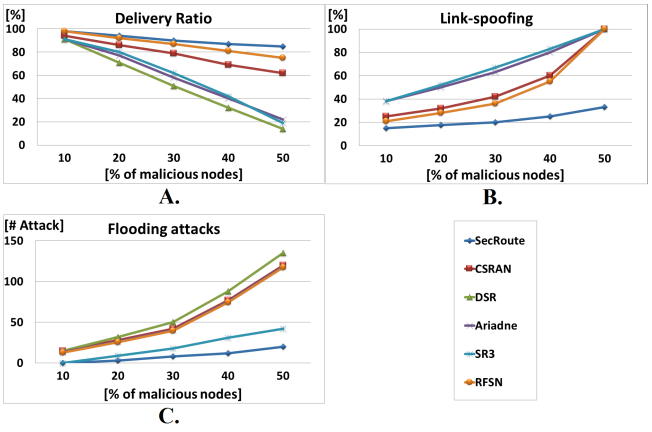

Fig. 2. Evaluation results for the three attack scenarios.

The most flooding attacks were performed under DSR, Ariadne, CSRAN and RFSN. For DSR, there were some paths that were rendered unavailable and falsely considered as broken. Ariadne, CSRAN and RFSN exhibit the most false accusations. In the cases of SR3 and SecRoute, mostly topology-based attacks were performed, as the load-balancing mechanisms mitigated the overloaded nodes and the relevant attacks. SecRoute exploits the fault-tolerance mechanism to detect the flooding as malicious activity even in congested periods, thus mitigating the false accusations. With the periodic re-entrance, nodes that are falsely accused enter the network after some time. SecRoute demonstrated the best behavior against this attack type.

VI. Proof-of-Concept - Railway CPS Protection

A SecRoute proof-of-concept is deployed on real embedded devices considering the intricacies of Railway CPS infrastructures, and its overhead compared to the pure DSR protocol. The DSR-UU implementation of the routing is extended and deployed on BeagleBone devices. Two setups are deployed. For in-carriage communication, a network of nine BeagleBone devices is created and connected wirelessly via USB Wi-Fi modules. Each device is equipped with environmental sensors. Moreover, the device at the carriage's entrance is equipped with infrared sensors and the device at the exit is equipped with a magnetic contact sensor. All nine devices use SecRoute. A tenth device controls a smart camera, and all ten devices feature the PBAC's PEP component. The devices transmit information to a base station, which maintains the authorization policies and the rest of PBAC's components, and controls the network. For outdoor route protection, a similar WSN setup is deployed. All devices are connected to a power supply, perform SecRoute and are equipped with weather sensors and a smart camera. In this scenario, the devices are emulated to be placed on the carriage departure, the line, the passenger's station and tunnels or bridges along the route and transmit real-time data to a security control center.

In both cases, the devices collect environmental parameters and transmit them to a processing center (laptop with Wi-Fi capability). The processing center deploys an application where the user can gain access and manage the system. We measure the overhead that is added to the DSR-UU protocol by the SecRoute during the normal network operation in terms of executable code size (kBs in ROM), average memory requirements (RAM consumption in bytes), and processing delay (ms of CPU time). Measurements were taken over a 24-hour operation of each system. Table 1 summarizes the average resource consumption of SecRoute, DSR-UU and the integrated SecRoute_DSR-UU. As is evident from the tests, the reputation calculation is the most resource-consuming component, due to the history of past values that is maintained for each evaluated operation. This feature proportionally affects the resource consumption of the trust metric. The added overhead for authentication, routing, forwarding and policy check is low. Average network latency is low, at 0.6 seconds. The integrated SecRoute_DSR-UU requires around 50% more memory and 70% more computational resources than DSR-UU; however, the additional overhead can be considered acceptable for the combination of enhanced security and load-balancing behavior.

TABLE I. RESOURCE CONSUMPTION OF SECROUTE

Component	ROM (kB)	RAM (kB)	CPU (ms)
Cryptographic service			
Authentication	3.7	2.22	0.0020
Encryption	25.0	10.41	0.0028
Authenticated encryption	28.7	12.63	0.0048
Trust scheme			
Direct Trust	5.6	4899.00	677.52
Reputation evaluation	20.0	1621.00	108.97
Indirect Trust	30.0	185.00	37.90
Total Trust	2.9	45756.00	9.48
Accept route request	2.0	0.00	104.23
Suitable Route Selection	15.0	40.00	33.17
PBAC policy check	24.7	36.00	7.50
Total SecRoute	210.4	46000.00	1652.50
DSR-UU	310.0	90000.00	2300.00
SecRoute_DSR-UU	520.4	136000.00	3952.50

The system can also be configured dynamically at runtime, to adjust to performance and security goals that are described in the active policy. The cryptographic service supports three communication settings: plaintext, authenticated, and authenticated encryption. The trust scheme offers two evaluation settings: direct trust, and direct and indirect trust. To retain resources at normal operation, the system can use authenticated communication and direct trust. Then, when the system detects a cyber-attack, it can inform the network nodes to increase their security level. A security policy orders a specific set of actions, such as using authenticated communication with both direct and indirect knowledge. The WSNs then adopt the policy, become stricter with misbehaving nodes and isolate the malicious ones. When the emergency is over, the system can return back to the normal (initial) configuration.

For the in-carriage WSN, we emulate the cases where a node is malfunctioning due to low battery and a malicious node that performs a blackhole attack. The first node is protected when the low energy level is detected and avoided from routing. The administrator is informed respectively. When the issue is resolved the node trust level is restored. The malicious node is detected once the attack rate passes a threshold and the node is excluded from routing. For the on-route WSN, we emulate the link-spoofing attack and the flooding attack against congested or topology significant nodes. SecRoute successfully detects both attacks and counters the attackers.

VII. CONCLUSION

This paper presented SecRoute – an end-to-end trust-based system for secure routing and authorization. Authentication is based on TESLA. A trust-based scheme protects the network against routing attacks. Authorization is enforced by a policy-based access control framework. Furthermore, the system can be configured dynamically to adjust security and performance metrics at runtime. SecRoute was evaluated using the NS-2 simulator, integrated with the routing protocol DSR. In comparison to four alternative systems, SecRoute's components are more efficient and provide higher protection. The system was also deployed on real embedded devices, as part of a Railway CPS scenario. SecRoute enhances the network's protection, with an acceptable performance overhead for the target devices, validating the feasibility of our approach. As a future work, we will extend the reasoning process of SecRoute with mechanisms to defend against jamming attacks, as well as the inclusion of trusted execution elements [16], to protect against malicious entities with physical access to devices; both feasible attacks, considering wide and unprotected deployments such as a railway infrastructure.

ACKNOWLEDGMENT

This work was supported by the General Secretarial Research and Technology (G.S.R.T.), Hellas under the Artemis JU research program nSHIELD, Grant Agreement No: 269317, and has also received funding from the European Union's Horizon 2020 research and innovation programme SHARCS under grant agreement No. 644571.

REFERENCES

[1] S. K. Singh, M. P. Singh, and D. K. Singh "Routing protocols in wireless sensor netwroks – a survey," Int. J. of Computer Science & Engineering Survey, vol. 1, no 2, pp. 63-83, November 2010 .

[2] G. Hatzivasilis and C. Manifavas "Building trust in ad hoc distributed resource-sharing networks using reputation-based systems,". 16th Panhellenic Conference on Informatics (PCI 2012), 2012, pp. 416-421.

[3] A. Perrig et al., "The TESLA Broadcast Authentication Protocol," CryptoBytes, vol. 5, issue 2, 2002, pp. 2-13.

[4] K. Fysarakis et al., "XSACd—Cross-domain resource sharing and access control for smart environments," Futur. Gener. Comput. Syst., Jun. 2016.

[5] K. Rantos, K. Fysarakis, C. Manifavas, and I. G. Askoxylakis, "Policy-Controlled Authenticated Access to LLN-Connected Healthcare Resources," IEEE Syst. J., pp. 1–11, 2015.

[6] M. Ngadi, R. H. Khokhar, and S. Mandala "A review of current routing attacks in mobile ad-hoc networks," International Journal of Computer Science and Security, vol. 2, issue 3, 2008, pp. 18-29.

[7] E. Carrara and G. Hogben "ENISA: Reputation-based System: a security analysis," European Network and Information Security Agency (ENISA) Position Paper, No.2, October 2007.

[8] S. Ganeriwal, L. Balzano, and M. Srivastava "Reputation-based framework for high integrity sensor networks," 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks, October 2004, pp. 66-77.

[9] Y.-C. Hu et al., "Ariadne: A secure on-demand routing protocol for ad hoc networks," Wireless networks, vol. 11, issue 1-2, 2005, 21-38.

[10] Y. Zhang, L. Xu, and X. Wang "A cooperative secure routing protocol based on reputation system for ad hoc networks," Journal of Communications, vol. 3, no. 6, pp. 43-50, November 2008.

[11] K. Sanzgiri et al., "A secure routing protocol for ad hoc networks," 10th IEEE International Conference on Network Protocols, IEEE, 2002, pp. 78-87.

[12] K. Altisen et al., "SR3: secure resilient reputation-based routing," IEEE Int. Conf. on Distributed Computing in Sensor Systems (DCOSS), 2013, pp. 258-265.

[13] G. Hatzivasilis et al., "ULCL: an Ultra-Lightweight Cryptographic Library for embedded systems," Measurable security for Embedded Computing and Communication Systems (MeSeCCS), 2014, Lisbon, Portugal, pp. 11-18.

[14] G. Hatzivasilis, I. Papaefstathiou, and C. Manifavas "ModConTR: a Modular and Configurable Trust and Reputation-based system for secure routing," 11th ACS/IEEE International Conference on Computer Systems and Applications, Qatar, 10-13 November, 2014, pp. 56-63.

[15] A. K. Trivedi et al., "A Semi-distributed Reputation Based Intrusion Detection System for Mobile Adhoc Networks," Journal of Information Assurance and Security, vol. 1, 2006, pp. 265-274.

[16] C. Shepherd et al., "Secure and Trusted Execution: Past, Present, and Future - A Critical Review in the Context of the Internet of Things and Cyber-Physical Systems," 2016 IEEE Trustcom/BigDataSE/ISPA. pp. 168–177, 2016.