23rd IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2012)

Performance Evaluation of PASER - an Efficient Secure Route Discovery Approach for Wireless Mesh Networks

Mohamad Sbeiti
Communication Networks Institute (CNI)
Faculty of Electrical Engineering and Information Technology
Dortmund University of Technology, Germany
Email: Mohamad.Sbeiti@tu-dortmund.de

Jakob Pojda
Vodafone D2 GmbH
Duesseldorf, Germany
Email: Jakob.Pojda@vodafone.com

Christian Wietfeld
Communication Networks Institute (CNI)
Faculty of Electrical Engineering and Information Technology
Dortmund University of Technology, Germany
Email: Chrsitian.Wietfeld@tu-dortmund.de

***Abstract*—In emergency and rescue operations, wireless mesh networks are attracting increased attention as a high-performance and low-cost solution for ubiquitous network access. In this paper, we evaluate the novel secure mesh route discovery protocol PASER, which has been designed to address the mesh network security in such critical environments. The protocol aims to set up reliable ad-hoc routes between network nodes and to combat unauthorized nodes of manipulating the route look-up process. Especially, its lightweight symmetric authentication scheme is noteworthy. The protocol is investigated together with the well-established routing protocols AODV, DYMO, BATMAN and OLSR under various scenario conditions and different attacks. In contrast to its counterparts, the results show that PASER is able to secure the network without noticeable computational overhead. On top of that, the paper reveals that PASER outperforms its counterparts in many cases in terms of packet delivery ratio and maximum end-to-end delay.**

I. Introduction

Urban and crowded areas have been recently witnessing growing vulnerabilities to natural, technological and terrorist disasters. In such environments, the fast and secure exchange of information between rescue fighters is imperative to efficiently manage the crisis. However, those cannot rely on public (cellular) networks, because, these networks are either demolished or overloaded (panic calls). Besides, dedicated emergency services/networks such as TETRA suffers from insufficient data rates, max. 7 kbit/s using one slot. Thereby, by using those networks, rescue fighters cannot share heavy (multimedia) data, thus, they cannot leverage the advantages of the advanced applications or equipment (helmet camera) they have. For that matter, rescue organizations are giving wireless mesh networks (WMN) increased attention as an appropriate low-cost and high performance solution.

Nonetheless, one of the main challenges the WMN technology is still confronting, is efficiently establishing on the fly reliable ad hoc routes over multiple mesh nodes from a sender to its destination. The latter makes WMN prone to a new range of routing attacks such as blackhole and wormhole [1]. And, without a satisfactory level of security, rescue organizations lack motivation to utilize this communication system. Then, terrorists or benefiting organizations may try to disrupt the communication between rescue fighters and their command and control systems. They might try to inject fraud packets to create a routing hole, which attracts and sniffs data packets, where any release of such sensitive data is troublesome. Thus, not only the security of data packets, for which a profusion of mechanisms already exists such as IPsec, is important. But also establishing accurate routes in presence of attackers is a fundamental need. Therefore, a core challenge of the communication in rescue and emergency operations is putting in place a secure and efficient routing protocol that makes all necessary data available at the right time.

The rest of this paper is organized as follows: Section II reports on related work. In Section III, PASER is demonstrated, where in Section V its performance and security are evaluated in comparison to well-known WMN routing protocols. Finally, in Section VI we conclude the paper and give some outlook for future work.

II. Related Work

Most WMN mesh routers nowadays are either built upon the IETF MANET routing protocols drafts, AODV [2], DYMO [3] and OLSR [4] or on the 'Freifunk-Community' protocol BATMAN [5]. Nevertheless, none of those protocols were designed with a non-trivial security in mind. They assume that conventional solutions like IPsec may be applicable. However, retrofitting pre-existing cryptosystems (e.g., IPsec) without redesign is not straightforward and clearly not efficient. Therefore, many approaches to secure routing have been recently proposed, see [6]. For instance, ARAN [7] is a protocol based on asymmetric cryptography, in which every node, originator or intermediate, must sign every message and appends this signature to the message. This provides a high level of security, but it is very expensive with respect to computational overhead and delay. In case of SAODV [8], only originators sign the message. The idea behind SAODV is to use a signature to authenticate non-changeable fields of a routing message and to use hash chains to authenticate the hop count. In contrast to MANET, in WMN, the hop count is not the only changeable field in the routing message. TTL field, metric field, per destination flag are also mutable and, thus, should be protected. The latter cannot be achieved using hash chains. SRP [9] is based on symmetric cryptography. But, it requires that for every route discovery, source and destination must have a

security association between them. Thus, the efficiency and scalability of this approach is questionable. Recently, Castor [10] has been proposed to provide secure routing without the need for routing messages. This approach is simple and yet it protects against a lot of attacks, especially, wormhole. Nonetheless, it produces huge byte- and packet overhead due to the Castor header added to each data packet and because of the acknowledgment required from each destination for each data packet. Besides, Castor assumes that each pair of end nodes either shares a symmetric key or they know the public key of each other. This is, however, not straightforward in WMN. Apart from that, from performance point of view, the aforementioned protocols along with a plenty of other MANET routing protocols does not take advantage of specific WMN characteristics. They deal with the network as a flat network. In WMN, yet, the network is hierarchical. Most data flow is destined to the gateway (e.g., from rescue fighters to officer in charge). Besides, they assume that a central infrastructure is not available, nevertheless, in WMN, this assumption is invalid.

To exploit the WMN characteristics, IEEE released the 802.11s standard [11], in which the Hybrid Wireless Mesh routing Protocol (HWMP) is specified. Security in routing or forwarding functionality is not considered in that protocol. It relies on the security of IEEE 802.11s. The latter is based on a new protocol called simultaneous authentication of equals, which allows a simultaneously authentication of two arbitrary peers, router or clients. The authentication protocol assumes a pre-shared secret, namely, a password to be known to all legitimate network peers. This approach is rudimentary and inflexible. Then, once an attacker reveals the password in use they can access the network. In that case, they can impersonate any arbitrary peer since the only additional identity attribute used is the MAC address. Revocation mechanisms to exclude a specific client or router from the network are not addressed in the protocol. On top of that, HWMP including the 802.11s security is still vulnerable against wormhole attacks.

For this end, we proposed in [12] the PASER protocol. It comprises a hybrid security system (symmetric and asymmetric) to efficiently achieve its security goals without the need for a pairwise security association between a source and a destination. The protocol has undergone several performance and security enhancements. Among others, it now endorses a key management scheme, which allows the revocation of any node at any time. The new version of the protocol is presented and evaluated in this paper.

III. PASER: *P*OSITION *A*WARE *S*ECURE AND *E*FFICIENT *R*OUTE DISCOVERY PROTOCOL

The Position Aware Secure and Efficient Route Discovery protocol (PASER) aims to efficiently establish accurate routes in terms of metric and legitimate mesh nodes in WMN networks in presence of external attackers. For that matter, it achieves the following goals: *Node authentication*, *message freshness and integrity* and *neighbor transmissions authentication* (necessary to protect against wormhole). Hereby, it assumes the following. First, only legitimate mesh nodes hold a valid certificate. These certificates are typically issued by the fire brigade organization (network operator). Second, GPS signals are available at the application scene. Nodes incorporate a secure GPS device (secure in terms of integrity and authenticity e.g., Galileo public regulated services).

A. PASER's Routing Part

From a routing point of view, PASER is a hierarchical reactive route discovery protocol. That is, it differs between gateways and mesh routers/access points, and mesh routers are always responsible, on demand, to register themselves once at a gateway. Figure 1(a) illustrates this process. It gives an overview of how a new node S, wanting to register itself at a gateway, performs the route discovery to that gateway. As this figure shows, PASER adopts the path accumulation approach (forwarding nodes append their own address to each route discovery message) and destination nodes replies to all received requests. The figure also depicts that in PASER, new neighbors establish a trust relationship between each other, after-which they communicate via unicast messages.

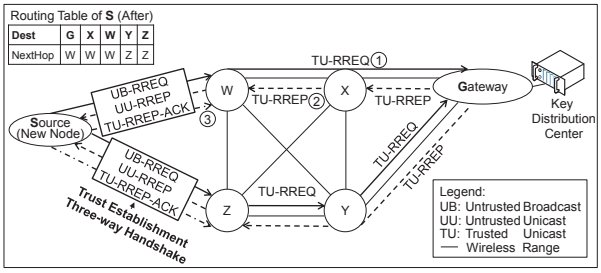
B. Security Mechanisms of PASER

PASER combines digital signature with lightweight authentication tree, keyed hash function and a key management scheme to secure the routing messages. Figure 1(c) provides an example on the application of these mechanisms.

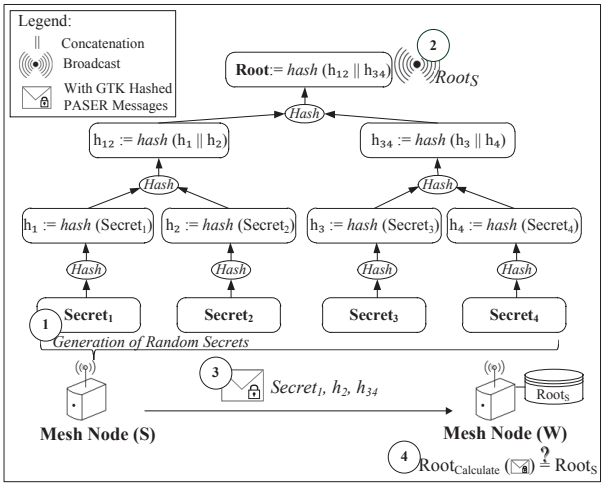
a) Digital Signature Scheme: It is used for the authentication of broadcast-messages; to establish trust between new one-hop neighbors. Hereby, a node use the key pair bounded to its certificate $\rightarrow_{achieved\ security\ goals}$ *node authentication and non-repudiation, message integrity and freshness (broadcast messages always comprise a sequence number), and neighbor transmission authentication (broadcast messages always comprise the GPS information of sending nodes. Only those who are located in the signal range of the receiver are considered as neighbors).*

b) Symmetric Authentication Scheme: It is based on Merkle trees [13] to authenticate unicast-messages between trusted one-hop neighbors. Figure 1(b) illustrates this process: First, mesh node S generates random secrets by concatenating an initialization vector/counter with random values. These secrets are the leaf pre-images of the authentication tree. Second, S computes the tree root and broadcasts it to its neighbors. Third, S, wanting to send data to neighbor W, discloses a secret (e.g., secret_1) and sends it along with the corresponding tree path (e.g., h_2 and h_{34}) with the data packet to W. Fourth, W, already knowing Root_S, computes the root of the secret it has received and verifies if it matches the root of S. If true, W can trust that the message has been sent by S $\rightarrow_{achieved\ security\ goals}$ *node authentication, message freshness (a secret is only used once), and neighbor transmission authentication (only neighbors exchange the root element).*

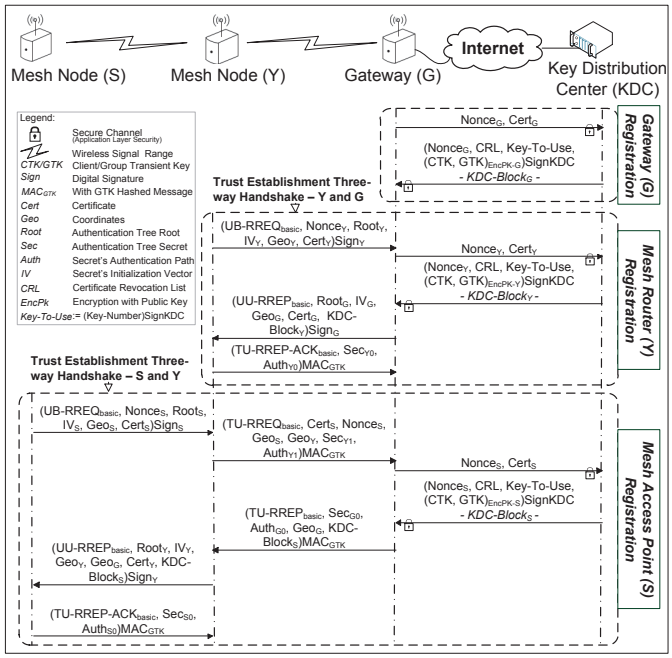
c) Keyed hash function: It is applied to guarantee the integrity of unicast-messages based on a transient group key. This function is always used in combination with the lightweight



(a) Overview of the route discovery in PASER during the registration of a new node



(b) PASER's symmetric authentication scheme



(c) Example of PASER's trust establishment three-way handshake during the registration of new nodes

Fig. 1. Overview of the PASER protocol

symmetric scheme to secure PASER messages between trusted neighbors (see Y and G after the handshake in Figure 1(c)) $\rightarrow_{achieved\ security\ goals}$ *message integrity*.

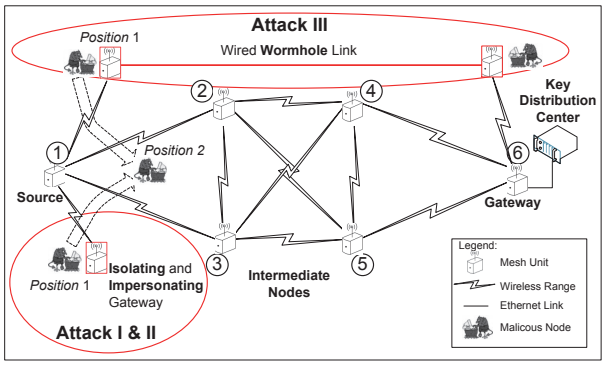


Fig. 2. Main network setup

d) Key Management Scheme: The dynamic distribution of the transient group key and the mesh access client key is illustrated in Figure 1(c). The latter occurs at network setup, when a node registers itself for the first time at a gateway. The gateway forwards the node's request to a KDC over a secure channel. The KDC responds to that request by sending the network keys encrypted with the node's public key. Hereby, a nonce is used to guarantee the freshness of the messages. Besides, a signed Key-To-Use mark is also sent to that node.

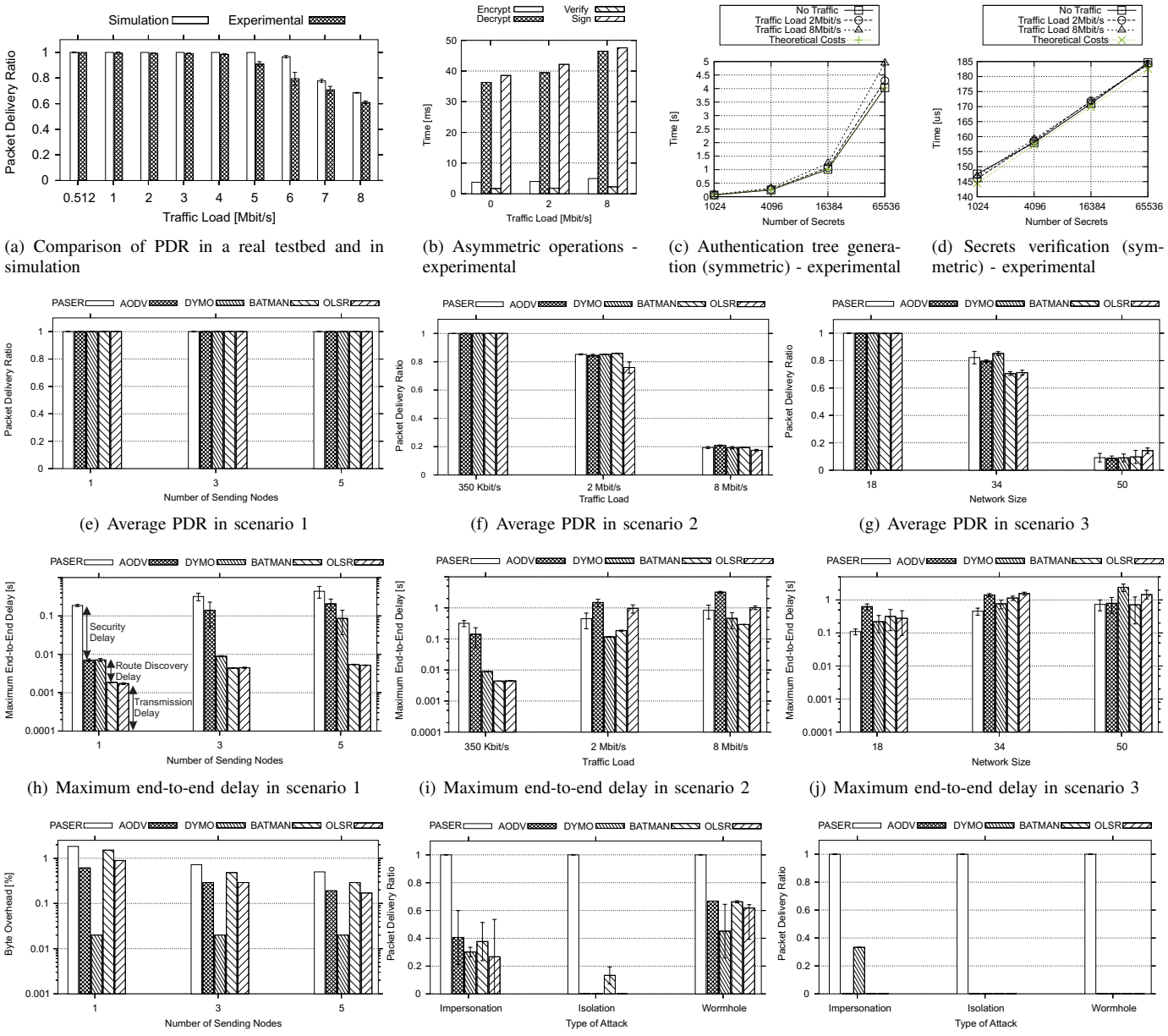
Nodes always include the number of the key in use in each PASER unicast message. This number is increased by one by each new generated key. The key is regenerated in case a node gets compromised. In that case, a new Key-To-Use mark, initialized by the KDC, is flooded in the network and the certificate of the compromised node is blacklisted. Upon receiving the new mark, each node resets its routing table and re-registers itself by the gateway. If a legitimate node was meanwhile unreachable, the node detects from the higher key number in use, that a key refreshment has occurred. Its neighbors even proof the latter using the new Key-To-Use mark. As a result, that node goes in a reset state. Due to the Key-To-Use mark, an attacker, who comprised a node, cannot denial the service of neighbor nodes by just increasing the key number of its messages $\rightarrow_{achieved\ security\ goals}$ *dynamic key distribution and key refresh/revocation.*

IV. Performance Evaluation

For a precise simulation-based evaluation, the performance of a mesh network has been first investigated in a real testbed using the BATMAN protocol. Afterwards, the simulation environment is tweaked to map the real performance of that network. Besides, the timing costs of the PASER security operations were first measured on an embedded hardware. The obtained results have been then fed into simulation to map the real delay of those operations.

A. Experimental Measurements

The testbed consists of six mesh nodes as depicted in Figure 2. These nodes operate with a reduced transmit power to allow for multi-hop communication on one floor of our institute. The embedded hardware board used is a Roboard RB-110 with a VortexX86 32-bit CPU running at 1000 MHz and



(a) Comparison of PDR in a real testbed and in simulation

(b) Asymmetric operations - experimental

(c) Authentication tree generation (symmetric) - experimental

(d) Secrets verification (symmetric) - experimental

(e) Average PDR in scenario 1

(f) Average PDR in scenario 2

(g) Average PDR in scenario 3

(h) Maximum end-to-end delay in scenario 1

(i) Maximum end-to-end delay in scenario 2

(j) Maximum end-to-end delay in scenario 3

(k) Average byte overhead scenario 1

(l) Average PDR in scenario 4 with attacker is at position 1

(m) Average PDR in scenario 4 with attacker is at position 2

Fig. 3. Experimental and simulation results

a 256MB DRAM. The board is equipped with a Wistron DMNA92 miniPCI WLAN device. The latter is operating according to the 802.11a technology. Furthermore, the unit provides an Ethernet port, which is dedicated for monitoring and configuration purposes. As for the software configuration, Debian Squeeze with the 2.6.37 Linux kernel is installed on the board. The goal of these measurements is on one hand, as aforementioned, to validate the simulation model. On the other hand, they aim to estimate the maximum achievable goodput as well as the deviation of these costs when advanced emergency and rescue applications are running. 2 Mbit/s is currently a satisfactory threshold for most emergency applications and 8 Mbit/s is a future need. Five repetitions for each measurement/simulation are performed. The results are depicted in Figure 3(a). They show that a traffic load up to 4 Mbit/s is completely delivered to the destination in both environments, real testbed and simulation. For higher traffic loads, the packet delivery ratio (PDR) of the testbed decreases nearly linearly to about 60% in case of 8 Mbit/s offered load. As expected, the experimental results comprise a bit more fluctuations, since the experimental environment is less controlled in comparison to simulation. Apart from that, the PDR in simulation is strongly comparable with the testbed results.

Using the boards and Iperf, the timing costs of the PASER security operations were measured in case of 2 and 8 Mbit/s constant UPD traffic loads, respectively. Hereby, two mesh

routers are used, one as sender and the second as destination. The aim of these measurements is to estimate the pure cost of those operations as well as the deviation of these costs by various data rates. The PASER operations are implemented using a non-optimized C language and the OpenSSL library. The secret size is set to 256 bits with an initialization vector of 32 bits. Five repetitions for each measurement are performed. The timing costs of the PASER asymmetric part are illustrated in Figure 3(b). Decryption and signature have by far a higher cost than encryption and verification, since the private exponent/key used needs to have the same length as the RSA modulus (1024 bits). In contrast, the public key is just 17 bits long. Apart from that, the figures illustrate the increase of those costs by higher traffic/processor load. These results show that the use of asymmetric cryptography for each message by each node, e.g., ARAN, is impracticable (almost $40ms$ by 0 traffic load) and, thus, raises the need for a symmetric scheme. The average timing costs of the PASER symmetric scheme are provided in Figure 3(c) and 3(d). They are defined by the following equations:

$$t_{root} = (n-1) \cdot (t_{sec}) + 2 \cdot (n-1) \cdot t_{hash} + t_{sec_i} + t_{hash_i} \quad (1)$$

$$t_{verification} = (log(n)) \cdot t_{hash} + t_{hash_i} \quad (2)$$

where n denotes the number of secrets, and t_{root} and $t_{verification}$ are the timing costs of the root generation and a secret verification, respectively. Hereby, t_{sec_i} and t_{hash_i} are the initial costs of a secret generation and a hash operation, respectively. These costs occur when the operations instructions haven't been loaded in the processor cache yet. The costs of those terms are $t_{hash_i} = 83.3\ \mu s$ and t_{sec_i} = $2534.3\ \mu s$. While the main costs of the computational operations of the functions are $t_{hash} = 6.3\ \mu s$ and $t_{sec} = 48.1\ \mu s$.

Figure 3(c) shows that the generation cost of an authentication tree to secure 65536 control messages is almost $4s$ and it increases with high processor load. This generation, however, occurs only at network setup or after all the secrets have been used up. In the former case, the $4s$ are definitely less than the time required to drop the mesh units at the incident scene (after turning them on). In the latter case, the regeneration of the tree occurs transparently to the user, when the processor is idle or in case of multi-core processors, it can be run in parallel. Thus, these $4s$ are in the practice mainly negligible. A more relevant impact is imposed by the results provided in Figure 3(d). These depict the time needed to verify the security of each received PASER message. As this figure shows, the time is less than 200 μs by a tree consisting of 65536 secrets. This value is only 10 % of the WLAN round-trip time. It even doesn't depend on the processor load, since it is very lightweight. Even more, it is about 200 times faster than the asymmetric scheme, and this is a gain achieved at each hop by each message.

B. Simulation-based Analysis

The discrete event-based simulation environment OMNeT++ [14] and its INETMANET framework are used for the investigation of PASER. The latter comprises the simulation of the standard network protocols as well as the here considered mobile ad hoc network protocols. PASER is evaluated in comparison with the two reactive protocols AODV and DYMO as well as the two proactive protocols BATMAN and OLSR. Figure 2 depicts the initial network setup used for that matter. All nodes are static, since only the route discovery is of interest. For a fair comparison, the protocols are configured with their optimal parameters with respect to PDR (based on several tests). Table I illustrates the network and traffic models used, the different scenarios addressed, and the most relevant protocol configurations. The RTS/CTS option at the MAC layer is turned off by all measurements, because, it has been verified that this technique yields rather performance penalties instead of benefits in the defined network, regardless of the RTS threshold. The latter is mainly caused by the network buffer overflow, which occurs while waiting for the CTS signal.

TABLE I
SIMULATION PARAMETERS

Common Network and Traffic Models

Parameter	Value	Parameter	Value
Nodes Transmission Range [m]	250	Mobility Pattern	Static
Antenna Type	Omni-directional	Mac Layer	802.11a
Channel Model	Free-space	Simulation Time [s]	125
Number of Simulations	10	Traffic Model	CBR-UDP
Packet Size	512 Bytes	Network Buffer Size [Packets]	100

Scenario Specific Network and Traffic Models

Parameter	Value			
Scenario Number	**1**	**2**	**3**	**4**
Number of Nodes	6	6	18/34/50	6
Data Rate [Mbit/s]	0.35	0.35/2/8	0.35	0.35
Number of Flows	1/3/5	3	9/17/25	3
Attacker's Position	-	-	-	1/2

Protocol Configuration Parameters

Protocol	Parameter	Value	Protocol	Parameter	Value
AODV,	routeTimeout [s]	6	PASER	neighborTimeout [s]	1
DYMO,	routeDeleteTimeout [s]	30		neighborDeleteTimeout [s]	20
PASER	RREQWaitTime [s]	1		number-of-Secrets	2^{14}
OLSR	Hello-interval [s]	2	AODV	Hello-interval [s]	1
	TC-interval [s]	6	DYMO	intermediateRREP	true
	Willingness	7	BATMAN	OGM-interval [s]	3

1) **Protocols Performance in General:** In this section, the measurements of scenario 1, 2 and 3 are discussed. The behavior of the protocols is first analyzed by a constant bit rate (CBR) of 350 kbit/s and an increasing number of senders (1; 1,2&3; 1,2,3,4&5) - scenario 1. Afterwards, the 3 sender model is selected as a reference and all measurements are repeated at higher data rates - scenario 2. In this context, the PASER experimental results of the different traffic loads are used as initialization parameters. Third, the number of intermediate nodes is increased (first by doubling the length and width of the network and then by successive increasing of the length) reaching a total network size of 50 nodes in scenario 3 - No attacks are performed in all 3 scenarios. All measurements are performed for 125s per run, which is the maximal time of a session by most emergency applications e.g., sending of building maps from the command and control system to the officer in charge via web services. The mean values and a confidence interval of 95% (by packet delivery

ratio and max. delay) are presented in the figures.
The results of scenario 1 are illustrated in Figure 3(e). These reveal that all protocols suit current rescue applications like VoIP or incident-scene-oriented videos with respect to PDR under the aforementioned conditions. In contrast, Figure 3(f) depicts a similar deterioration of the PDR by all protocols at higher data rates. Nonetheless, this is essentially caused by the overflow of the node's network buffer while waiting to access the medium in a WLAN mesh network. This mainly does not depend on the routing protocol. Figure 3(g) shows that all protocols perform almost the same with respect to PDR in larger networks with a tiny advantage for PASER and DYMO by the network size of 34. This is mainly caused by the low number of protocol packets and by the path accumulation concept, on which both protocols are based.
While the average end-to-end delay of the data packets is similar by all protocols, Figure 3(h) depicts their differences with respect to the maximum end-to-end delay value. This parameter typically reflects, in case of low network collusion, the packet transmission time from sender to destination by proactive protocols. In case of reactive protocols, it also comprises the time needed to discover the route to that destination. These observations are depicted in flow 1, which also reflects the typical delay imposed by the security mechanisms of PASER, mainly the costs of the asymmetric scheme. The negative impact of these costs, however, strongly decreases by increasing traffic as shown in flow 2 and 3. The latter gets even more apparent in Figure 3(i), which shows that the maximum delay caused by PASER is nearly even to that of the other protocols by high data rates. Because, in that case, the delay caused by the retransmission of collided UDP packets is much higher than the security delay itself. Figure 3(j) shows that PASER performs in average the best with respect to the maximum delay. This sheds again the light on the efficiency of the PASER symmetric scheme since, for instance, despite the verification of the security of a message by 32 nodes, PASER performs the best with respect to delay by the network size of 34 nodes.
Still, for a fair comparison, it is important to mention that the size of the PASER messages is higher than those of the other protocols, see Figure 3(k), which depicts the ratio of protocol bytes, counted at each hop, to data bytes. The integration of the node certificates or the tree authentication paths into the messages causes this impact by PASER. The latter generally leads, in contrast to the aforementioned, to a higher collision probability of a PASER message.

2) **Protocols Behavior in Presence of Attacks:** In this section, the behavior of the protocols is investigated under *impersonation, isolation and wormhole* attacks, respectively. For that matter, the PDR of three sending nodes (1,2 & 3) is measured in a network of 6 nodes and the attacker has been placed in 2 different positions - scenario 4. Figure 2 depicts the network setup. The following observations are made:
a) Impersonation: the attacker has the same IP-address as the gateway. They just reply to route requests addressed to that gateway in order to redirect the data traffic of the requesting node to themselves. In case of PASER, we assume that the attacker has compromised a node and that node credentials have been blacklisted by the key distribution center.
AODV, DYMO, BATMAN & OLSR: At position 1 in Figure 2, node 1 always selects the route to the attacker, since it is the optimal one. In contrast, by the other 2 nodes, namely 2 & 3, the distance to the attacker is as far as to the gateway. Thus, whether the nodes decide to send their packets to the attacker or to the gateway mainly depends on both, random events like WLAN medium access and on the protocol design. Thereby, the PDR varies from 20 to 60 % as illustrated in Figure 3(l). At position 2, all protocols but DYMO choose the route to that attacker as the best route, since they are only one hop away to all sending nodes. In DYMO, when node 1 starts a route request to the gateway, both, the attacker and the gateway, reply to that request. The gateway's reply reaches S either over node 2 or 3. As a result, that node detects the correct route to the gateway and does not start a route request. Thereby, DYMO achieves 30 % at position 2, see 3(m). This effect is not provided by AODV, since all nodes register the attacker as a neighbor because of their Hello messages.
PASER: Since the attacker has no valid credentials any more, they are neither able to join the network nor able to falsify or to reply to any messages regardless of their position. Thus, the PDR remains 1 in both cases.
b) Isolation: The attacker goes beyond impersonation and tries, in addition, to isolate the gateway so that all networks nodes use the attacker as a gateway. For that purpose, they broadcast repeatedly corrupted packets with very high sequence numbers. As a result, the gateway packets having lower sequence numbers are not taken into consideration.
AODV & DYMO: Since both protocols are reactive, the gateway just replies to route requests. Meanwhile, the attacker repeatedly sends in the name of the gateway route requests looking for an unknown node. These requests are flooded in the whole network. Thus, all nodes register the route to the attacker as a valid gateway-route and all packets are then sent to that attacker. Thereby, the PDR of both protocols in both positions decreases to 0% .
BATMAN: This protocol maintains a trust parameter of one byte for each route. At the beginning, this parameter is set to zero, i.e., no body trusts anybody. Upon receiving valid Hello/OGM messages, the trust parameter of the route increases. In case an OGM arrives via several routes, only the one arriving over the most trusted route will be taken into consideration. If all routes taken by that OGM are equally trusted, the OGM with the highest sequence number will be considered. At position 1, node 1 is closer to the attacker than to the other nodes. As a result, it will trust the attacker more than the gateway, since the OGMs of the attacker arrive faster. On the other hand, nodes 2 and 3 have almost the same distance to the attacker as to the gateway, thereby, the route choice is random. Nonetheless, after the trust value of both routes gets saturated, only the attacker's OGMs are to be taken into consideration, since these have higher sequence numbers.

The latter justifies the 15% packet delivery ratio in Figure 3(l). At position 2, the attacker is closer to all sending nodes than the gateway, thereby, only their packets are considered which is also reflected by 0 % PDR in Figure 3(m).

OLSR: Both, the attacker and the gateway are 2 hops far from Nodes 2 and 3. Thereby, according to OLSR with the number of hops as metric, in case there are several routes to a destination with the same number of hops, the last inserted route is chosen. Since the attack is implemented in a way that the last inserted route is the route to the attacker, i.e., the attacker starts broadcasting packets after the gateway, nodes 2 and 3 choose this route as the best route, the same applies to node 1. Thereby, the PDR in Figure3(l) is 0 %. At position 2, the attacker is one hop away from all sending nodes. The latter justifies the PDR of 0% in Figure3(m).

PASER: Since impersonation attack is not possible, isolation attack is also not possible. Thereby, the PDR remains 1 in both positions.

c) Wormhole: a pair of attackers linked via a fast transmission path (e.g., Ethernet) forward route requests more quickly than legitimate nodes. While conventional cryptosystems e.g., IEE802.11i-PSK might protect to some extent from the previous attacks at the expense of performance and management and as long as the password hasn't been cracked, they cannot combat against this attack. The main reason for this is that control packets are simply forwarded, without any changes, from one end to the other end of the tunnel. The tunneled packets can propagate faster than those through a normal multi-hop route. This causes victim nodes to always use the tunneled route to transmit their packets, which are then dropped by the attacker.

AODV, DYMO, BATMAN & OLSR: At position 1, node 1 keeps trying to establish a route to the gateway over the tunnel, thinking the gateway is its neighbor. However, it keeps receiving the WLAN-ACK of the gateway after its timeout is exceeded because of the forwarding delay over the tunnel. The latter keeps this node busy the whole time, since after restarting a route request it will again choose the tunnel as the best route. As a result, node 1 is not able to forward the route requests of node 2 and node 3 over the tunnel. Thereby, both nodes always choose the legitimate route to the gateway. Note, however, that this is the simplest implementation of a wormhole. By a more complex implementation, an attacker could replay a WLAN-ACK. At position 2 all three nodes use the wormhole as the best route achieving a PDR of 0%.

PASER: Since all sending nodes are not in the signal range of the gateway, it detects the wormhole and drops all packets arriving over the tunnel.

V. Conclusion

In this paper, we analyze the performance of the secure mesh route discovery approach, PASER. The protocol is investigated together with the two reactive protocols AODV and DYMO and the two proactive protocols BATMAN and OLSR in different scenarios and under various attacks. The results show that PASER achieves a good tradeoff between security and performance. In contrast to the other protocols, it is able to protect the network against routing attacks, while achieving a comparable performance with those protocols.

In ongoing work, we are designing the route maintenance part of PASER based on a combination of both, WLAN-ACK and Hello messages. Besides, we are in an advanced phase of the experimental implementation of PASER and its evaluation on embedded hardware. Apart from that, we are extending the wormhole protection of PASER for indoor scenarios using a novel virtual localization technique.

In future work, we intend to formally prove the security of PASER using practice-oriented provable security. Besides, we aim to analyze the energy consumption imposed by PASER especially by the GPS component it incorporates.

Acknowledgment

The authors would like to thank Eugen Paul for his technical assistance. Our work has been conducted within the SPIDER project, which is part of the nationwide security research program funded by the German Federal Ministry of Education and Research (BMBF) (13N10238). We also acknowledge the support of the AVIGLE project, which is co-funded by the German federal state North Rhine Westphalia (NRW) and the European Union (European Regional Development Fund: Investing In Your Future).

References

[1] B. Kannhavong, H. Nakayama, Y. Nemoto, N. Kato, and A. Jamalipour, "A survey of routing attacks in mobile ad hoc networks," *IEEE Wireless Communications*, vol. 14, no. 5, pp. 85–91, 2007.

[2] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) routing," RFC 3561, Jul. 2003.

[3] I. Chakeres and C. Perkins, "Dynamic MANET On-Demand (DYMO) Routing," draft-ietf-manet-dymo-21, Jul. 2010.

[4] T. Clausen and P. Jacquet, "Optimized Link State Routing (OLSR) Protocol," RFC 3626, Oct. 2003.

[5] (2012, Jun.) Better Approach To Mobile Ad hoc Networking (B.A.T.M.A.N.). Freifunk Community. [Online]. Available: http://www.open-mesh.org/

[6] L. Abusalah, A. Khokhar, and M. Guizani, "A Survey of Secure Mobile Ad hoc Routing Protocols," *IEEE Communications Surveys and Tutorials*, vol. 10, no. 4, pp. 78–93, 2008.

[7] K. Sanzgiri, D. Laflamme, B. Dahill, B. N. Levine, C. Shields, and E. Belding-Royer, "Authenticated Routing for Ad hoc Networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, pp. 598–610, Mar. 2005.

[8] M. Zapata and N. Asokan, "Securing ad hoc routing protocols," in *ACM WiSe*, Sep. 2002, pp. 1–10.

[9] A. Papadimitratos and Z. Haas, "Secure routing for mobile ad hoc networks," in *SCS CNDS*, Jan. 2002, pp. 314–318.

[10] W. Galuba, P. Papadimitratos, M. Poturalski, K. Aberer, Z. Despotovic, and W. Kellerer, "Castor: Scalable Secure Routing for Ad Hoc Networks," in *IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[11] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Std 802.11 - 2012, Mar. 2012.

[12] (2012, Jun.) Position Aware Secure and Efficient Mesh Routing (PASER). [Online]. Available: http://www.paser.info

[13] A. Menezes, P. van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 1996, ch. 13, p. 556.

[14] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment," in *SIMUTools*, Feb. 2008, pp. 60:1–60:10.