



BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS
Faculty of Electrical Engineering and Informatics

Department of Telecommunications
Laboratory of Cryptography and Systems Security (CrySyS)

SECURE ROUTING IN MULTI-HOP WIRELESS NETWORKS

Collection of Ph.D. Theses
by

Gergely Ács

Research Supervisor:
Levente Buttyán, Ph.D.

SUBMITTED IN PARTIAL FULFILMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
AT
BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS
BUDAPEST, HUNGARY

February, 2009

1 Introduction

Routing is a fundamental networking function in multi-hop wireless networks. An adversary can easily paralyse the operation of a whole network by attacking the routing protocol. Moreover, attacks against the routing protocol usually do not require a lot of resources, the manipulation of a few routing messages or injecting fabricated messages are sufficient to subvert the normal operation of the network. In this dissertation, I focus on the security of the route or topology discovery phase of routing protocols proposed for multi-hop wireless networks.

Most routing protocols proposed for wireless ad hoc and sensor networks are either insecure or they are designed to be secure but their security has been analyzed by only informal reasoning. However, informal reasoning is prone to errors due to the subtle nature of attacks against multi-hop wireless routing protocols. While secure messaging and key-exchange protocols are classical and well-studied problems in traditional networks [Bellare *et al.*, 1998] [Pfitzman and Waidner, 2001a], the literature of the formal modelling of secure multihop wireless routing is surprisingly poor. Although some prior works [Yang and Baras, 2003] [Papadimitratos and Haas, 2002] have used formal techniques to model the security of multi-hop routing protocols, these ones were proposed for ad hoc network routing primarily. Moreover, these techniques are either applicable to the security analysis of specific routing protocols [Papadimitratos and Haas, 2002] exclusively, or their definition of secure routing is so strong [Yang and Baras, 2003] that it is unclear if there exists a protocol at all that would satisfy it. Even more, tools that would allow the security analysis of routing protocols are also missing.

2 Research objectives

*My objective is to identify the design principles of secure routing protocols in multi-hop wireless networks by analysing their security in a **systematic and rigorous way**.*

In particular, my goals are

1. to develop a *general* formal framework in which precise definitions of secure routing can be given, and routing protocols proposed for multi-hop wireless networks can be rigorously analyzed. *General* in this context means that the model considers the variety of routing protocols [J4] [J5] in multi-hop wireless networks.
2. to demonstrate the usefulness of the framework on real examples by showing how *existing* secure routing protocols in wireless ad-hoc and sensor networks can be analyzed in this model. I also develop new secure routing protocols and prove that they are secure in this model.

3 Methodology

In a general sense, I use an analytic technique to analyse the security of routing protocols. More particularly, I apply a technique that is similar to the simulation paradigm (see Parts V and VI of [Mao, 2004]). Note that this simulation paradigm is not related to traditional network simulations, it is rather about a computational model which has been used successfully so far to prove the security of various cryptographic protocols. My model is similar to this simulation paradigm, where security is defined in terms of indistinguishability between an

ideal-world model of the system (where certain attacks are not possible by definition) and the real-world model of the system (where the adversary is not constrained, except that it must run in polynomial time). However, there are important differences between the classical simulation approach and my work. Instead of the real-world model, the dynamic model describes the real operation of the protocol participants in my model, and I use the so-called security objective function to specify how a protocol that is under investigation should operate ideally. Particularly, at the end of each simulation run, the security objective function is applied to the routing state of all honest nodes to decide whether the protocol works according to the specified security objective. The protocol is secure if this security objective function results in a “non-acceptable” value only with a negligible probability, where the definition of what is acceptable or not is protocol dependant. This function may be different for different types of routing protocols, but the general approach of comparing the output of this function in the dynamic model to a pre-defined “acceptable” value remains the same.

I also develop an adversary model that is different from the standard Dolev-Yao model [Dolev and Yao, 1981], where the adversary can control all communications in the system. In wireless ad hoc and sensor networks, the adversary uses wireless devices to attack the systems, and it is more reasonable to assume that the adversary can interfere with communications only within its power range. In addition, in some situations it can also manipulate the communication of directly connected honest nodes, which, in turn, does not hold for wired networks. Therefore, I must also model the broadcast nature of radio communications.

4 New Results

Thesis 1 [C2] [J3] [J1] [J2] [J4] [J5] *I propose a new and general mathematical framework that consists of a model and a proof technique, which allows us to define the notion of routing security precisely, to model a given routing protocol, and to prove that a routing protocol satisfies the definition of security in this model.*

Security flaws in ad hoc and sensor network routing protocols can be very subtle. Consequently, making claims about the security of a routing protocol based on informal arguments, like in [Karlof and Wagner, 2003] or [Hu *et al.*, 2002], is dangerous. Although some attempts [Yang and Baras, 2003; Marshall, 2003; Papadimitratos and Haas, 2002] are made to use formal methods for the verification of wireless network routing protocols, they either use inappropriate assumptions to prove routing security, or they are not general enough to model the security of different routing protocols. In particular, these techniques are either applicable to the security analysis of specific routing protocols [Papadimitratos and Haas, 2002; Marshall, 2003] exclusively, or their definition of secure routing is so strong [Yang and Baras, 2003] that it is unclear if there exists a protocol at all that would satisfy it.

It is important to emphasize that the proposed framework is best suited for proving that a protocol is secure (if it really is), but it is not directly usable to discover attacks against routing protocols that are flawed. Note, however, that such attacks may be discovered indirectly by attempting to prove that the protocol is secure and examining where the proof fails. To the best of my knowledge, no similar general security framework existed before for routing protocols.

The high level overview of my framework is as follows. In my approach, a model is constructed for the protocol under investigation that is called the *dynamic model*. This model describes the operation of the protocol with all its details in a particular computational model. The dynamic model also contains an adversary that is an arbitrary process (i.e., it may not follow the protocol rules faithfully), and it is only constrained to run in polynomial time. This allows us to consider any feasible attacks which makes the model general. Instead of constructing an ideal-world model according to the standard simulation paradigm like in [Bellare *et al.*, 1998; Pfitzman and Waidner, 2001a; Shoup, 1999; Pfitzman and Waidner, 2001b; Backes and Pfitzmann, 2004], I use a security objective function to represent the security objective of the protocol under investigation (I note that a similar approach is used in [Backes and Pfitzmann, 2004], where the integrity property was defined in terms of a security objective of the Needham-Schroeder-Lowe Public-Key Protocol). In particular, this function is applied to the ensemble of the routing entries of honest nodes and decides whether the entries satisfy a certain security objective or not. The security objective also has to incorporate the tolerable imperfections of the system. Recall that the tolerable imperfections of the model are those attacks that are unavoidable, or it is too costly to defend against, and hence, we rather tolerate them. The protocol is said to be secure if the protocol executed in the dynamic model violates the security objective only with a negligible probability.

In more details, my framework consists of the following parts: adversary model, network model, security objective function, dynamic model, definition of security, and a proof technique.

Adversary model: In my model, the adversary intends to thwart the primary objectives of routing protocols [J4] [J5]. Generally, the primary goals of the adversary can be degrading the packet delivery ratio, increasing its control over traffic, increasing network delay, and shortening network lifetime depending on the routing objectives.

In the models proposed so far, the adversary has full control over the communications of the honest protocol participants. This means that it can read, modify, or delete any of the messages sent between any protocol participants, and it can also inject forged messages to any protocol participant. This may be an appropriate model in Internet-like networks, where having access to some special network elements, such as routers, allows the adversary to have this level of control. On the other hand, in wireless ad hoc and sensor networks, an adversary can have a similar level of control over the communications only if it is physically present everywhere. Although this can also hold for wireless ad hoc and even more for wireless sensor networks, this is considered to be very costly, and hence, unrealistic in many applications. Therefore, I assume that the adversary has communication capabilities comparable to those of an average node in ad hoc networks, or, in wireless sensor networks, the adversary may additionally have a more resourced laptop-class device. All of these adversarial nodes may be able to communicate via in-band (e.g., tunneling) or out-of-band channels (e.g., other frequency channel or direct wired connection to implement wormholes).

In general, when capturing honest sensor nodes, the adversary may be able to compromise their cryptographic secrets (assuming that such secrets are used in the system). This inherently allows some attacks which are either very costly to defend against or they are not directly against the routing layer. These are the side-channel attacks (including tunneling) [Burmester and de Medeiros, 2007], different DoS attacks, rushing attacks [Hu, 2003], Sybil attacks [Douceur, 2002], and node replication attacks [Bryan *et al.*, 2005]. These attacks are

excluded from my model.

The adversary is active in the sense that besides eavesdropping messages it can fabricate and insert new messages in transit, and in addition, it can modify, delete, re-order and delay existing messages that traverse it without following the routing protocol rules faithfully. Note that, in contrast to traditional wired networks, the adversary is able to manipulate the communication of two directly connected honest nodes.

In the dissertation, I do not deal with all known attacks [B1] against multi-hop wireless routing. Particularly, *I focus on the basic message manipulation attacks that aim to corrupt the routing entries of honest nodes* where a routing entry is a representation of a route towards a particular destination node. This representation can be the list of identifiers of nodes constituting the route like in DSR, or the identifier of the next-hop along which there should be a route to the destination with a certain cost. The goal of the adversary is to cause honest nodes to store such routing entries that are not consistent with the underlying network topology, where the definition of consistency is protocol dependant, and such routing entries are called as *incorrect routing entries*.

Although the set of considered attacks is only a small subset of all known attack methods, even some of the existing “secure” routing protocols are vulnerable to these attacks where the adversary uses only simple message manipulation techniques exclusively.

Network model: I assume that each honest device has exactly one transceiver in the network. If the adversary uses several transceivers I represent each of them by a distinct node. The network nodes are considered to be static, and I assume that there is a single base station in sensor networks.

The honest nodes in the network are denoted by v_0, \dots, v_k , where v_0 denotes the base station in case of sensor networks, and adversarial nodes are denoted by v_{k+1}, \dots, v_{k+m} . The set of all nodes in the network is denoted by V , and the set of adversarial nodes is denoted by V^* , where $|V| = n = m + k + 1$, and $|V^*| = m$.

In order to model the connectivity between the nodes, I introduce a matrix \mathbf{E} , called *reachability matrix*, with size $n \times n$. Here, $E_{i,j}$ ($0 \leq i, j \leq n - 1$) represents the output power level which can be used by v_i and needed for v_i to communicate with v_j (i.e., if node v_i uses power level $E_{i,j}$ to broadcast a message, then v_j also receives the message). In case v_j cannot receive any messages from v_i (e.g., there are obstacles between them, or the needed output power level is too high for v_i), $E_{i,j} = \infty$

I assume that each honest node can use a single globally unique identifier in the network, and these identifiers are authenticated in some way (e.g., by cryptographic means). I denote the set of these identifiers by L , and there is a function $\mathcal{L} : V \rightarrow 2^{L \cup \{\text{undef}\}}$ that assigns a set of identifiers to each node, where $\text{undef} \notin L$ is assigned to those nodes which have no identifiers in the network. According to my adversary model, I assume that the adversary has (authenticated) identifier(s) in the network, denoted by L^* that can be used by all adversarial nodes (i.e., $\mathcal{L}(v_{k+j}) = L^*$ for all $1 \leq j \leq m$). Moreover, for every honest node v_i ($1 \leq i \leq k$), $\mathcal{L}(v_i)$ is a singleton, and $\mathcal{L}(v_i) \notin L^*$.

Finally, a *cost function* $\mathcal{C} : V \rightarrow \mathbb{R}$ assigns a routing cost value to each node in the network (e.g., the minimal processing delay, or constant 1 to each node in order to represent hop-count, etc.) that could influence the routing decisions.

Configuration: A *configuration* of a network is a quintuple $conf = (V, V^*, \mathcal{L}, \mathbf{E}, \mathcal{C})$.

I make the assumption that the configuration is static (at least during the time interval that is considered in the analysis).

Security objective function: The state of the system is represented by the ensemble of the routing entries of *all* honest nodes in the network. The reason that I consider the result of the protocol with respect to the honest nodes exclusively is that the adversarial nodes may not follow the protocol rules faithfully. As the specification of routing entries depends on the routing protocol that is under investigation, the definition of system state is also protocol dependant. The security objective function $\mathcal{F} : \mathbb{G} \times \mathbb{S} \rightarrow \{0, 1\}$ a binary function, where \mathbb{S} denotes the set of all system states of all configurations, and \mathbb{G} denotes the set of all configurations. Let \mathcal{F} return 0 for all pairs of system states and configurations that are incorrect, otherwise it returns 1 (or vice-versa). This function intends to distinguish “attacked” (incorrect) states from “non-attacked” (correct) states.

Dynamic model: The dynamic model that corresponds to a configuration $conf = (V, V^*, \mathcal{L}, E, \mathcal{C})$ and adversary \mathcal{A} is denoted by $sys_{conf, \mathcal{A}}$. I model the operation of the protocol participants by interactive and probabilistic Turing machines. The computation of the model ends, when all machines that represent the honest nodes reach their final states, or there is a time-out.

The output of $sys_{conf, \mathcal{A}}$ is the value of the security objective function \mathcal{F} applied to the resulted system state and configuration $conf$. The system state is represented by the ensemble of the routing entries of honest machines. I denote the output by $Out_{conf, \mathcal{A}}^{\mathcal{F}}(r)$, where r is the random input of the dynamic model. In addition, $Out_{conf, \mathcal{A}}^{\mathcal{F}}$ will denote the random variable describing $Out_{conf, \mathcal{A}}^{\mathcal{F}}(r)$ when r is chosen uniformly at random.

I denote the security parameter of the model by κ (e.g., κ is the key length of the cryptographic primitive employed in the routing protocol, such as MAC, digital signature etc.). Based on the model described above, I define routing security as follows:

Definition 1 *A routing protocol is secure with respect to security objective function \mathcal{F} , if for any configuration $conf$ and any adversary \mathcal{A} , the probability that $Out_{conf, \mathcal{A}}^{\mathcal{F}}$ equals to zero is a negligible function of κ .¹*

Proof technique: In particular, by proving the security of a protocol, we must show that those system states which violate our security objective (i.e., there is a configuration $conf$ such that applying function \mathcal{F} to those system states with $conf$ results in 0) occur only with a negligible probability. However, even the number of all configurations for a given number of nodes is an exponential function of the number of all nodes. Thus, proving the security of a protocol by searching for all pairs of system states and configurations and test whether \mathcal{F} returns 0 with these pairs seems to be a hard problem at first sight. However, all such pairs can be *reduced* to a few cases for all protocols which are analysed in my work. Then, we must prove that each of these cases occurs only with a negligible probability which concludes that the protocol satisfies Definition 1. In order to do this, we show that these cases can only occur in the model, if the adversary successfully breaks at least one cryptographic primitive (like the applied MAC, digital signature, or encryption scheme) used by the routing protocol.

¹a function $\mu(x) : \mathbb{N} \rightarrow \mathbb{R}$ is negligible, if for every positive integer c and all sufficiently large x 's (i.e., there exists an $N_c > 0$ for all $x > N_c$), $\mu(x) \leq x^{-c}$

However, assuming that the applied primitives are secure, the probability of this event is a negligible function of the length of the security parameter (i.e., κ in my model).

In practice, failure of a proof usually indicates a problem with the protocol, and often, one can construct an attack by looking at where the proof failed.

Thesis 2 [C2] *I adapt the general security framework proposed in Thesis 1 to the TinyOS beaconing protocol used in wireless sensor networks, and I prove that the authenticated TinyOS beaconing protocol, proposed in [Perrig et al., 2002], is insecure with respect to a security objective function which requires that if an honest node sets another node as a next-hop towards the base station, then they must be neighbors, or each of them must have an adversarial neighbor.*

Originally, the authors of TinyOS [Hill et al., 2000] proposed a very simple routing protocol, called TinyOS beaconing. In this protocol, each node is addressed by a globally unique identifier, and the base station periodically initiates a route discovery by flooding the network with a beacon message in order to build a routing tree. A lightweight cryptographic extension is employed in [Perrig et al., 2002] in order to authenticate the beacon by the base station in TinyOS beaconing. This authenticated variant of TinyOS routing uses μ Tesla scheme to provide integrity for the beacon; each key is disclosed by the next beacon in the subsequent beaconing interval. In order to be compliant with my model, I present a variant of this protocol which provides the “same” security as the authenticated routing protocol in [Perrig et al., 2002]. Consequently, the presented attack against this new protocol also works against the protocol in [Perrig et al., 2002].

The operation and messages of this authenticated TinyOS beaconing are exemplified in Table 1. The base station B is assumed to have a public-private key pair for signature generation, and all sensor nodes have the corresponding public key. Initially, B creates a beacon, that contains a constant message identifier BEACON, a randomly generated number rnd , the identifier of the base station id_B , and a digital signature sig_B generated on the previous elements except id_B . Then, the base station floods the network by broadcasting this beacon. Each sensor node X receiving this message checks whether it has already received a beacon with the same rnd in conjunction with a correct signature before. If it is true, the node discards the message, otherwise it checks whether sig_B is correct. If so, then X sets id_B as its parent, and re-broadcasts the beacon by changing the sender identifier id_B to its own identifier id_X . Otherwise, X discards the message. Every sensor node receiving the beacon performs the same steps what X has done before.

In the data forwarding process, every sensor node receiving a data packet forwards that towards the base station by sending the packet to its parent.

$B \rightarrow *$:	$(\text{BEACON}, rnd, id_B, sig_B)$
$X \rightarrow *$:	$(\text{BEACON}, rnd, id_X, sig_B)$

Table 1: The operation and messages of authenticated TinyOS beaconing.

As the adversary can exchange messages between adversarial nodes through in-band (e.g., tunnelling) or out-of-band channels (e.g., wormholes) which is very costly to defend against, a pair of honest can be “neighbors”, if both of them have an adversarial neighbor.

Definition 2 (Pseudo neighbors) *Two honest nodes $v_i, v_j \in V \setminus V^*$ ($i \neq j$) are pseudo neighbors, if and only if there exist x, y ($k + 1 \leq x, y \leq k + m$) such that $E_{i,x} = 1$ and $E_{j,y} = 1$, and $v_x, v_y \in V^*$.*

Two nodes are pseudo neighbors, only if each of them has an adversarial neighbor. In the sequel, we distinguish pseudo neighbors from direct neighbors; two honest nodes v_i, v_j are direct neighbors, if $E_{i,j} = 1$. However, note that being direct neighbors and pseudo neighbors are not exclusive.

I recall that the state of the system is the ensemble of the routing entries of all *honest* nodes. This system state with a given configuration *conf* is represented by a matrix T^{conf} with size $(k + 1) \times (k + 2)$ in case of authenticated TinyOS beaconing, and I refer to this as a *routing topology* with configuration *conf* in the sequel. For all $0 \leq i \leq k$,

- and $0 \leq j \leq k$, $T_{i,j}^{conf} = 1$, if honest node v_i sends every message to an honest node v_j in order to deliver the message to the destination node, otherwise let $T_{i,j}^{conf}$ be 0.
- $T_{i,k+1}^{conf} = 1$, if honest node v_i sends every message to an adversarial node in order to deliver the message to the destination node, otherwise let $T_{i,k+1}^{conf}$ be 0.

Note that the rows and columns of T^{conf} are numbered from zero, and the same holds for all matrices in the rest of the dissertation.

In this way, a routing topology can also be considered as a directed graph described by matrix T^{conf} . In fact, T^{conf} is a random variable, where the randomness is caused by the sensor readings initiated randomly by the environment, processing and transmission time of the sensed data, etc.

A simple security objective of authenticated TinyOS beaconing is to guarantee the correctness of all routing entries in the network. Namely, it is desirable that a sender node v_i is always able to reach node v_j , if v_i set $\mathcal{L}(v_j)$ as its parent identifier earlier. It means that if node v_i sets node $\mathcal{L}(v_j)$ as its parent identifier, then $E_{i,j}$ should contain a finite value, or v_i as well as v_j should be pseudo neighbors.

Let the security objective function of authenticated TinyOS beaconing, denoted by \mathcal{F}^{ATB} , return 1 for all pairs of system states and configurations where for all i, j , if $T_{i,j} = 1$, then v_i and v_j are direct or pseudo neighbors. Otherwise, \mathcal{F}^{ATB} returns 0.

Theorem 1 *Authenticated TinyOS beaconing is insecure with respect to \mathcal{F}^{ATB} .*

The proof is based on the fact that the immediate sender of a beacon does not authenticate the beacon. Thus, an adversarial node can easily send a beacon in the name of any honest nodes. The whole proof is detailed in the dissertation.

Thesis 3 [J3] I designed a novel secure source routing protocol, called *endairA*, for wireless ad-hoc networks. *endairA* is the reverse of *Ariadne*, because, instead of signing the request like in *Ariadne* [Hu et al., 2002], I propose that intermediate nodes should sign the route reply. *endairA* is more efficient than *Ariadne*, because it requires less cryptographic computation overall from the nodes. I adapt the general security framework of Thesis 1 to dynamic source routing in wireless ad-hoc networks, and I prove that *endairA* is secure with respect to a security objective function which requires that each path returned by the routing protocol to a source node must contain a sequence of node identifiers such that

- the first identifier belongs to the source node,
- the last identifier belongs to the destination node,
- there are no repeating identifiers,
- and each pair of non-corrupted intermediate identifiers that are either consecutive or non-consecutive but there are only corrupted identifiers between them corresponds to a pair of nodes in the network such that they are direct or pseudo neighbors.

In *endairA*, the initiator of the route discovery process generates a route request, which contains the identifiers of the initiator and the target, and a randomly generated request identifier. Each intermediate node that receives the request for the first time appends its identifier to the route accumulated so far in the request and rebroadcasts the request. When the request arrives to the target, it generates a route reply. The route reply contains the identifiers of the initiator and the target, the accumulated route obtained from the request, and a digital signature of the target on these elements. The reply is sent back to the initiator on the reverse of the route found in the request. Each intermediate node that receives the reply verifies that its identifier is in the node list carried by the reply and that the preceding identifier (or that of the initiator, if there is no preceding identifier in the node list) and the following identifier (or that of the target, if there is no following identifier in the node list) belong to neighboring nodes. Each intermediate node also verifies that the digital signatures in the reply are valid and that they correspond to the following identifiers in the node list and to the target. If these verifications fail, then the reply is dropped. Otherwise, it is signed by the intermediate node, and passed to the next node on the route (toward the initiator). When the initiator receives the route reply, it verifies if the first identifier in the route carried by the reply belongs to a neighbor. If so, then it verifies all the signatures in the reply. If all these verifications are successful, then the initiator accepts the route.

The operation and messages of *endairA* are exemplified in Table 2.

endairA has a significant advantage over similar secure source routing protocols like *Ariadne*: it is more efficient, because, similar to SRDP [Kim and Tsudik, 2005], it requires less cryptographic computation overall from the nodes. This is because in *endairA*, only the processing of the route reply messages involves cryptographic operations, and a route reply message is processed only by those nodes that are in the node list carried in the route reply. In contrast to this, in *Ariadne*, the route request messages need to be digitally signed by all intermediate nodes; however, due to the way a route request is propagated, this means that each node in the network must sign each and every route request.

$S \rightarrow *$:	$(\text{rreq}, S, D, id, ())$
$A \rightarrow *$:	$(\text{rreq}, S, D, id, (A))$
$B \rightarrow *$:	$(\text{rreq}, S, D, id, (A, B))$
$T \rightarrow B$:	$(\text{rrep}, S, T, (A, B), (sig_T))$
$B \rightarrow A$:	$(\text{rrep}, S, T, (A, B), (sig_T, sig_B))$
$A \rightarrow S$:	$(\text{rrep}, S, T, (A, B), (sig_T, sig_B, sig_A))$

Table 2: An example of the operation and messages of endairA. The initiator of the route discovery is S , the target is T , and the intermediate nodes are A and B . id is a randomly generated request identifier. sig_A , sig_B , and sig_T are digital signatures of A , B , and T , respectively. Each signature is computed over the message fields (including the signatures) that precede the signature.

I note that SRDP [Kim and Tsudik, 2005] is similar to endairA in the sense that instead of signing the requests, each intermediate node only signs the reply messages. However, the focus of [Kim and Tsudik, 2005] is to explore different cryptographic techniques with different levels of security, efficiency, and robustness. The common characteristic of these primitives is that the signature list in the route reply can be replaced with a single aggregate signature or MAC computed iteratively by the intermediate nodes in order to reduce communication overhead. By contrast, my work is focused on the design and formal validation of secure source routing protocols. Finally, I note that endairA was first published in [Ács *et al.*, 2005], which was earlier than [Kim and Tsudik, 2005].

Intuitively, the minimum that one may require from the route discovery part of a routing protocol is that it returns only existing routes. My security objective is built on this intuition. Taking into account that the adversary can exchange messages between adversarial nodes through in-band (e.g., tunnelling, or side-channel attacks) or out-of-band channels (e.g., wormholes) which is very costly to defend against, I introduce the definition of *plausible routes*.

Definition 3 (Plausible route) *A sequence $\ell_1, \ell_2, \dots, \ell_n$ of identifiers is a plausible route with respect to configuration $conf$ and labelling function \mathcal{L} , if each of the identifiers $\ell_1, \ell_2, \dots, \ell_n$ is different and there exists a sequence v_1, v_2, \dots, v_t ($2 \leq t \leq n$) of honest nodes such that*

- $\mathcal{L}(v_1) = \ell_1$ and $\mathcal{L}(v_t) = \ell_n$;
- for all $1 \leq i \leq t - 1$,
 - there exists $1 \leq j, d \leq n - 1$ such that $\mathcal{L}(v_i) = \ell_j$ and $\mathcal{L}(v_{i+1}) = \ell_{j+d}$, where for all $j + 1 \leq z \leq j + d - 1$, $\ell_z \in L^*$;
 - v_i and v_{i+1} are direct or pseudo neighbors.

I recall that a system state represents the set of all routing entries of all *honest* nodes in the network. In the case of source routing, a routing entry includes a route (i.e., a sequence of node identifiers) that is used for data forwarding towards the destination which is the last element of this route. Let \mathcal{F} of secure dynamic source routing return 0 for all pairs of system states and configurations that contains a non-plausible route with respect to the configuration, where this non-plausible route belongs to an honest node. Otherwise, \mathcal{F} returns 1. According

to Definition 1, if any honest node in the dynamic model returns a non-plausible route with non-negligible probability for a given configuration, then the protocol is insecure.

Theorem 2 *endairA is a secure source routing protocol for wireless ad hoc networks, if the signature scheme is secure against chosen message attacks.*

The proof of this theorem is detailed in the dissertation. In contrast to this, Ariadne is insecure considering the same security objective. This demonstrates that my model is able to distinguish between source routing protocols in terms of security. A consequence of the analysis is that a secure source routing protocol should always authenticate route reply messages in order to prevent incorrect routing entries.

Thesis 4 [C1] *I adapt the general security framework of Thesis 1 to dynamic distance vector routing in wireless ad-hoc networks, and I prove that SAODV [Zapata and Asokan, 2002] is insecure with respect to the security objective function which requires that if a source node has a routing entry towards a destination node, then there exists a route in the network that*

- *starts from the source node,*
- *ends at the destination node,*
- *has a cost that is smaller than or equal to the cost recorded in the entry, and*
- *all consecutive honest nodes lying on the route have a next-hop identifier towards the destination node which is either a corrupted identifier, or the identifier of the next direct or pseudo neighboring honest node.*

SAODV [Zapata and Asokan, 2002] is a “secure” variant of the Ad hoc On-demand Distance Vector (AODV) [Perkins and Royer, 1999] routing protocol. The operation of SAODV is similar to that of AODV, but it uses cryptographic extensions to provide integrity of routing messages and to prevent the manipulation of the hop count information. Conceptually, SAODV routing messages (i.e., route requests and route replies) have a non-mutable and a mutable part. The non-mutable part includes, among other fields, the node sequence numbers, the addresses of the source and the destination, and a request identifier, while the mutable part contains the hop count information. Different mechanisms are used to protect the different parts.

The non-mutable part is protected by the digital signature of the originator of the message (i.e., the source or the destination of the route discovery). This ensures that the non-mutable fields cannot be changed by an adversary without the change being detected by the non-corrupted nodes.

In order to prevent the manipulation of the hop count information, the authors propose to use hash chains. When a node originates a routing message (i.e., a route reply or a route request), it first sets the `HopCount` field to 0, and the `MaxHopCount` field to the `TimeToLive` value. Then, it generates a random number *seed*, and puts it in the `Hash` field of the routing message. After that, it calculates the `TopHash` field by hashing *seed* iteratively `MaxHopCount` times. The `MaxHopCount` and the `TopHash` fields belong to the non-mutable part of the message, while the `HopCount` and the `Hash` fields are mutable. Every node receiving a routing message hashes the value of the `Hash` field (`MaxHopCount – HopCount`) times, and verifies

whether the result matches the value of the **TopHash** field. Then, before rebroadcasting a route reply or forwarding a route request, the node increases the value of the **HopCount** field by one, and updates the **Hash** field by hashing its value once.

The rationale behind using the above hash chaining mechanism is that given the values of the **Hash**, the **TopHash**, and the **MaxHopCount** fields, anyone can verify the value of the **HopCount** field. On the other hand, preceding hash values cannot be computed starting from the value in the **Hash** field due to the one-way property of the hash function. This ensures that an adversary cannot decrease the hop count, and thus, cannot make a route appearing shorter than it really is.

An entry of the routing table of a given node v is assumed to contain the following three fields: the identifier of the target node, the identifier of the next hop towards the target, and the cost value that represents the believed cost of the route to the given target via the given next hop. Without loss of generality, I assume that the routing metric is such that routes with lower cost values are preferred.

Consequently, the state of the system in my model will be represented by a set $Q \subset (V \setminus V^*) \times L \times L \times \mathbb{R}$ of quadruples such that for any (v, z_{tar}, z_{nxt}, c) and $(v', z'_{tar}, z'_{nxt}, c')$ in Q , $v = v'$ and $z_{tar} = z'_{tar}$ and $z_{nxt} = z'_{nxt}$ implies $c = c'$. The quadruple (v, z_{tar}, z_{nxt}, c) in Q represents an entry in v 's routing table with target identifier z_{tar} , next hop identifier z_{nxt} , and believed route cost c . The ensemble of quadruples that have v as their first element represent the entire routing table of v , and the ensemble of all quadruples in Q represent the ensemble of the routing tables of all honest nodes (i.e., the state of the system). Note that we allow that a node's routing table contains multiple entries for the same target, but the next hops should be different.

Considering that SAODV uses the hop count as a path length metric, $\mathcal{C} : V \rightarrow \mathbb{R}$ assigns a constant 1 to each node. In order to ease further formalizations, I introduce the definition of workable path.

Definition 4 (Workable path) *A sequence of honest nodes $(v_{\ell_0}, v_{\ell_1}, \dots, v_{\ell_{d-1}}, v_{\ell_d})$ is a workable path with respect to configuration $conf$ if for all $0 \leq i \leq d - 1$ v_{ℓ_i} and $v_{\ell_{i+1}}$ are direct or pseudo neighbors.*

I define correct states as follows:

Definition 5 (Correct state) *A state Q is correct if for every entry $(v_{src}, z_{dst}, z_{nxt}, c_{src}) \in Q$, there exists a sequence of entries $(v_{\ell_i}, z_{dst}, z_{\ell_i}, c_{\ell_i}) \in Q$ ($1 \leq i \leq d$) such that*

- $(v_{src}, v_{\ell_1}, \dots, v_{\ell_{d-1}}, v_{dst})$ is a workable path, where $v_{\ell_d} = v_{dst}$,
- $z_{dst} \in \mathcal{L}(v_{dst})$,
- let $v_{\ell_0} = v_{src}$ and $z_{\ell_0} = z_{nxt}$,
 - if $v_{\ell_{i-1}}$ and v_{ℓ_i} are direct but not pseudo neighbors then $z_{\ell_{i-1}} \in \mathcal{L}(v_{\ell_i})$,
 - if $v_{\ell_{i-1}}$ and v_{ℓ_i} are pseudo neighbors then either $z_{\ell_{i-1}} \in \mathcal{L}(v_{\ell_i})$, or $z_{\ell_{i-1}} \in L^*$,
- $\sum_{i=1}^{d-1} \mathcal{C}(v_{\ell_i}) \leq c_{src}$.

Intuitively, the system is in a correct state, if all the routing table entries of the *honest* nodes are correct in the sense that if v_{src} has an entry for target z_{dst} with next hop z_{nxt} and cost c_{src} , then indeed there exists a route in the network that

- starts from node v_{src} ,
- ends at a node that uses the identifier z_{dst} ,
- has a cost that is smaller than or equal to c_{src} , and
- all consecutive honest nodes lying on the route have a next-hop identifier towards v_{dst} which is either a corrupted identifier, or the identifier of the next direct or pseudo neighboring honest node.

Let the security objective function \mathcal{F} of secure dynamic distance vector routing return 0 for all pairs of system states and configurations where the system state is incorrect with respect to the configuration. Otherwise, \mathcal{F} returns 1.

Theorem 3 *SAODV is an insecure distance vector routing protocol for wireless ad hoc networks.*

The proofs are based on the fact that SAODV cannot guarantee that the next hop and the hop count information in the newly created routing table entry is correct. This is caused by the lack of previous-hop (neighbor) authentication. The proof, which describes two specific attacks against SAODV, is detailed in the dissertation.

Thesis 5 [C1] *Using the adapted security framework of Thesis 4, I prove that ARAN [Sanzgiri et al., 2002] is secure considering the same security objective function as in Thesis 4.*

Just like SAODV [Zapata and Asokan, 2002], ARAN as well uses public key cryptography to ensure the integrity of routing messages. In ARAN, nodes update their routing tables using the information obtained from the routing messages that arrive first; any later message that belongs to the same route discovery is discarded. This means that ARAN may not necessarily discover the shortest paths in the network, but rather, it discovers the quickest ones. Similarly to SAODV, the source as well as the destination authenticates the route request and route reply by digital signatures, however, in contrast to SAODV, all intermediate nodes also sign every request and reply message. Particularly, each intermediate node before re-broadcasting a request or a reply, replaces the previous-hop signature with its own signature. The operation and messages of ARAN are exemplified in Table 3.

Considering that ARAN uses the message propagation delay (i.e., physical time) as a path length metric, $\mathcal{C} : V \rightarrow \mathbb{R}$ assigns the minimal delay of routing messages to each node in the network (i.e., the minimal delay that the particular node can cause in the travel of the message).

The correct state, and thus, the security objective function is defined in the same way as in the case of SAODV.

Theorem 4 *ARAN is a secure distance vector routing protocol for wireless ad hoc networks, if the signature scheme is secure against chosen message attacks.*

The proof of this theorem is detailed in the dissertation and in [C1]. In contrast to this, SAODV (Secure AODV) [Zapata and Asokan, 2002] is insecure considering this security objective due to the lack of the next-hop authentication [C1]. This clearly demonstrates that

$S \rightarrow *$:	$(RREQ, T, cert_S, N_S, t, Sig_S)$
$A \rightarrow *$:	$(RREQ, T, cert_S, N_S, t, Sig_S, Sig_A, cert_A)$
$B \rightarrow *$:	$(RREQ, T, cert_S, N_S, t, Sig_S, Sig_B, cert_B)$
$T \rightarrow B$:	$(RREP, S, cert_T, N_S, t, Sig_T)$
$B \rightarrow A$:	$(RREP, S, cert_T, N_S, t, Sig_T, Sig_B, cert_B)$
$A \rightarrow T$:	$(RREP, S, cert_T, N_S, t, Sig_T, Sig_A, cert_A)$

Table 3: An example of the operation and messages of ARAN. The initiator of the route discovery is S , the target is T , and the intermediate nodes are A and B . N_S is a randomly generated request identifier. Sig_A , Sig_B , and Sig_T are digital signatures of A , B , and T , resp., and $cert_A$, $cert_B$, and $cert_T$ are the public-key certificates of A , B , and T , respectively. t denotes the current time-stamp. Each signature is computed over the message fields (including the signatures) that precede the signature.

my model is able to distinguish between distance vector routing protocols in terms of security. A conclusion of the analysis is that source and destination authentication in the route discovery process are not sufficient to guarantee security in my model.

Thesis 6 [C3] *I adapt the general security model of Thesis 1 to link-state routing in wireless sensor networks, and I prove that INSENS [Deng et al., 2002] is secure with respect to the objective function which requires that*

- *if an honest node v sets another node v' as its parent node for data forwarding, then the base station has indeed computed v' as the parent node for v ,*
- *if the base station is aware of the fact that two nodes are neighbors, then they are direct or pseudo neighbors.*

INSENS is a secure link-state routing protocol for wireless sensor networks. First, the base station initiates the routing topology construction by flooding the network with a route request message (Phase 1). Each node constructs its own neighborlist by overhearing the request messages sent by its neighbors. Afterwards, each sensor node sends its own neighborhood information to the base station on the route on which the request was received previously (Phase 2). Then, the base station computes the routing table for each individual sensor node, and propagates the routing tables to respective nodes in a breadth-first manner (Phase 3). INSENS employs symmetric key cryptography to ensure message authenticity and integrity (each node has a pairwise key shared with the base station). Besides using message authentication codes (MACs), all topology information carried by the routing messages is also encrypted.

The operation and messages of INSENS are described in Table 3. First, the base station v_0 initiates the routing topology construction by flooding the network with a route request message (Phase 1), where $hash$ is the next element of the hash chain in reversed direction. The hash chain mechanism is intended to provide authenticity and some defense against DoS attacks. Each node constructs its own neighborlist by overhearing the request messages sent by its neighbors. Every subsequent node v_{ℓ_i} receiving the first authentic request stores $MAC_{v_{\ell_i-1}}^{REQ}$

in conjunction with the identifier of $v_{\ell_{i-1}}$ locally. Before re-broadcasting, v_{ℓ_i} replaces $\text{MAC}_{v_{\ell_{i-1}}}^{\text{REQ}}$ in the request with $\text{MAC}_{v_{\ell_i}}^{\text{REQ}}$ which is the MAC generated on the elements that precede the MAC. If a node v_{ℓ_x} does not receive further request messages for a specified time, it sends its neighborlist to $v_{\ell_{x-1}}$ from which it received the first authentic request (Phase 2). Here, $\text{Enc}_{v_{\ell_x}}(\text{path}_{v_{\ell_x}}, \text{neighborlist}_{v_{\ell_x}})$ is the neighborhood information and the path information of v_{ℓ_x} encrypted by the symmetric key shared with the base station, $\text{neighborlist}_{v_{\ell_x}}$ contains the identifiers of each neighboring node *and* their corresponding MACs received in Phase 1, $\text{path}_{v_{\ell_x}}$ is $[v_{\ell_x}, \dots, v_{\ell_1}, v_0, \text{MAC}_{v_{\ell_x}}^{\text{REQ}}]$, which is the reverse of the path received in the corresponding request message including the MAC of node v_x , and $\text{MAC}_{v_{\ell_x}}^{\text{NLIST}}$ is the MAC computed by node v_{ℓ_x} on these elements. Upon the reception of all neighborhood information, the base station can compute all routing information for each node. Then, the base station distributes this routing information in a breadth-first manner (Phase 3). $\text{Enc}_{v_{\ell_1}}(\text{ftable}_{v_{\ell_1}})$ is the encrypted form of the forwarding table of v_{ℓ_1} , and $\text{MAC}_{v_{\ell_1}}^{\text{FTABLE}}$ is the MAC generated on the elements of the message.

Phase 1:	
$v_0 \rightarrow *$: (REQ, hash, $[v_0]$)
$v_{\ell_i} \rightarrow *$: (REQ, hash, $[v_0, \dots, v_{\ell_{i-1}}, v_{\ell_i}]$, $\text{MAC}_{v_{\ell_i}}^{\text{REQ}}$)
Phase 2:	
$v_{\ell_x} \rightarrow v_{\ell_{x-1}}$: (NLIST, hash, $\text{MAC}_{v_{\ell_{x-1}}}^{\text{REQ}}, v_{\ell_x}, \text{Enc}_{v_{\ell_x}}(\text{path}_{v_{\ell_x}}, \text{neighborlist}_{v_{\ell_x}}), \text{MAC}_{v_{\ell_x}}^{\text{NLIST}}$)
Phase 3:	
$v_0 \rightarrow v_{\ell_1}$: (FTABLE, v_{ℓ_1} , hash, $\text{Enc}_{v_{\ell_1}}(\text{ftable}_{v_{\ell_1}})$, $\text{MAC}_{v_{\ell_1}}^{\text{FTABLE}}$)

Table 4: The operation and messages of INSENS.

Similarly to Thesis 2, the system state with a given configuration conf is represented by a matrix T^{conf} with size $(k+1) \times (k+2)$, which is called the *routing topology* with configuration conf . In the following, I will omit the index conf of T when the configuration can be unambiguously determined in a given context.

I show that INSENS has the following properties:

1. If an honest sensor node v_i ($1 \leq i \leq k$) sets $v_j \in V$ ($0 \leq j \leq n-1$) as its parent node for data forwarding, then the base station has indeed computed v_j as the parent node for v_i .
2. If the base station is aware of the fact that node v_j is a neighbor of node v_i , then node v_i can reach node v_j by either a direct contact, or an adversarial relaying (one can also imagine the adversarial relaying as a wormhole between some honest nodes).

Intuitively, if INSENS has these two properties, then it is ensured that each honest node has a *neighboring* parent node that is computed by the base station. Moreover, it is also guaranteed that this computation performed by the base station is based on, perhaps incomplete (the adversary can always drop routing messages containing neighborlists, which we are unable to defend against), but correct neighborhood information.

In order to formalize the above security objective, I introduce a matrix function \mathcal{G} . \mathcal{G} models the centralized construction of the topology performed by the base station, where the

argument of \mathcal{G} with size $(k+1) \times (k+2)$, denoted by \mathbf{N} , describes the neighborhood relations among the sensor nodes that is believed by the base station to be correct (i.e., $N_{i,j} = 1$ if the base station believes that v_i is a neighbor of v_j , otherwise $N_{i,j} = 0$. For any $0 \leq i \leq k$, $N_{i,k+1} = 1$, if honest node v_i has at least one adversarial neighbor, otherwise $N_{i,k+1} = 0$). The output of \mathcal{G} is the ensemble of the routing entries (the routing topology) that should be set by each node.

Definition 6 (Correct routing topology) *A routing topology \mathbf{T} is correct with respect to configuration conf , if there exists a matrix \mathbf{E}' such that for all i, j it holds that if $T_{i,j} = 1$, then $\mathcal{G}(\mathbf{E}')_{i,j} = 1$, where \mathbf{E}' is derived from \mathbf{E} with size $(k+1) \times (k+2)$ and it is defined as follows. For all $0 \leq i, j \leq k$, $E'_{i,j} = 0$, if v_i and v_j are neither direct nor pseudo neighbors. For all $0 \leq i \leq k$, $E'_{i,k+1} = 0$, if v_i has no direct adversarial neighbor.*

Let the security objective function \mathcal{F} of secure link-state routing return 0 for all pairs of system states and configurations where the system state (i.e., the routing topology) is incorrect with respect to the configuration. Otherwise, \mathcal{F} returns 1.

Theorem 5 *INSENS is a secure link-state routing protocol for wireless sensor networks, if the MAC scheme is secure against chosen message attack, and the symmetric encryption scheme is secure against plaintext recovery attack.*

The proof, which is detailed in the dissertation, strongly relies on the fact that INSENS encrypts the local neighborhood information that needs to be transferred to remote nodes. The encryption of neighborlists used in INSENS is crucial; apart from providing confidentiality for the neighborhood relations, the encryption of neighborlists prevents the adversary to impersonate honest nodes that are not covered by the transmission range of any adversarial nodes. For instance, if the neighborlists were not encrypted, an intermediate adversarial node could easily retrieve the identities and corresponding $\text{MAC}^{\text{REQ}}_s$ from NLIST messages, and then it could re-broadcast fabricated REQ messages. Note that the adversary is not required to reach the impersonated node directly. Apparently, this would also violate our security objective detailed above, as the adversary could cause the base station to consider false neighborhood relations. Furthermore, as $\text{MAC}^{\text{REQ}}_s$ are correct, it can happen that neither the neighbors of the adversary nor the base station could detect the misdeed. My formal analysis lead me to the following observation: in case of link-state routing, all local neighborhood (routing) information that is needed by remote nodes to authenticate neighborhood relations must be transferred in an encrypted form.

Thesis 7 [J6] I propose a novel secure decentralized label-switching routing protocol called *Secure-TinyLUNAR* that is the secure variant of *TinyLUNAR* [Osipov, 2007] for wireless sensor networks. Using label-switching routing and efficient symmetric key cryptography exclusively, *Secure-TinyLUNAR* is more applicable in wireless sensor networks than existing secure ad hoc network routing protocols like *ARAN* [Sanzgiri et al., 2002]. I adapt the general security model of *Thesis 1* to label-switching routing in wireless sensor networks, and I prove that *Secure-TinyLUNAR* is secure with respect to the objective function which requires that if a source node has a routing entry towards a destination node, then there exists a route in the network that

- starts from the source node,
- ends at the destination node,
- has a delay that is not greater than the delay recorded in the entry, and
- all consecutive honest nodes v, v' lying on the route have routing entries r, r' , resp., such that the next-hop address and the outgoing label of r equals to identifier of v' and the incoming label of r' , resp., if v and v' are direct but not pseudo neighbors. If v and v' are pseudo neighbors, then either the next-hop address and the outgoing label of r equals to identifier of v' and the incoming label of r' , resp., or v' is an adversarial node. Moreover, if v' is the destination node, then the outgoing label of r' is an application identifier.

Considering the variety of sensor applications, it is also clear that it is not possible to propose a unique secure routing protocol that fits for all applications. An alternative solution could be to apply some secure ad hoc network routing protocol like [Zapata and Asokan, 2002], [Sanzgiri et al., 2002] [Hu et al., 2002]. However, these protocols are not primarily designed for low-powered sensor nodes, and the applied asymmetric cryptographic primitives can result in extensive communication, processing and memory costs. Therefore, I design a novel secure routing protocol for wireless sensor networks, called *Secure-TinyLUNAR*, which takes into consideration the resource constraints of the wireless sensor nodes and uses MAC exclusively in the route discovery phase. Moreover, using the label-switching routing paradigm, *Secure-TinyLUNAR* has only one byte addressing overhead per packet in the data forwarding phase, which, considering the high communication costs in wireless environment, makes it an efficient routing scheme in relatively static networks.

Similar to *TinyLUNAR* [Osipov, 2007], *Secure-TinyLUNAR* uses node identifiers to address sensor nodes and assumes bidirectional links in the network. It is also assumed that each pair of nodes share a symmetric pairwise key in the network. Any symmetric key pre-distribution schemes proposed for wireless sensor networks (see [Çamtepe and Yener, 2005] for a good overview) can be employed for this purpose. Additionally, each node is assumed to be aware of its local (one-hop) neighborhood.

In the following, I only describe the main operational differences with respect to the original (and insecure) *TinyLUNAR* [Osipov, 2007] protocol.

Route request: Let us denote the identifier of a neighboring node of node A by N_x^A , where x can have a value between 1 and the number of the neighboring nodes of A (e.g., if A has neighbors J, T, P , then a potential notation is $N_1^A = J, N_2^A = T, N_3^A = P$, and $1 \leq x \leq 3$).

When a node S wishes to send a message to destination D , it broadcasts the following route request message:

$$S \rightarrow * : (\text{RREQ}, \text{rnd}, S, D, \text{addr}_S, \text{label}_{S \rightarrow S}^{\text{In}}, \text{MAC}_{S,D})$$

where $\text{rnd}, S, D, \text{addr}_S, \text{label}_{S \rightarrow S}^{\text{In}}$ are the same as in the original TinyLUNAR protocol, and $\text{MAC}_{S,D}$ is the message authentication code generated by S on the elements of the message excluding addr_S and $\text{label}_{S \rightarrow S}^{\text{In}}$ using the pairwise key shared with D . Upon the reception of this broadcast message, J checks whether S is a neighboring node. If so, node J unicasts the following message to each neighbor except the node who sent the request to J earlier (here, this is S):

$$\text{for all } x \text{ such that } N_x^J \neq S, J \rightarrow N_x^J : (\text{RREQ}, \text{rnd}, S, D, \text{addr}_J, \text{label}_{J \rightarrow S}^{\text{In}}, \text{MAC}_{S,D}, \text{MAC}_{J,N_x^J}^{\text{prv}})$$

where $\text{MAC}_{J,N_x^J}^{\text{prv}}$ is the previous-hop MAC generated on all elements of the message using the pairwise key shared between J and N_x^J . Each neighbor of S and all subsequent nodes receiving a request follow the same steps that J did (except that they update the previous-hop MAC). Finally, D receives a request message, let us assume, from node Z first.

During the propagation of a request, it is assumed that each node can send the unicast request message to its immediate neighbors in an atomic manner (i.e., the sender does not release the channel until all request messages are transmitted to each neighbor), and each neighboring node does not begin to forward the request until all neighbors of the sender receive that.

Route reply: Upon the reception of the request message, destination D verifies both $\text{MAC}_{S,D}$ and $\text{MAC}_{Z,D}^{\text{prv}}$. If the verifications are successful, D creates the following reply message and sends this directly to node Z :

$$D \rightarrow Z : (\text{RREP}, \text{rnd}, \text{addr}_D, \text{label}_{Z \rightarrow S}^{\text{Out}}, \text{label}_{D \rightarrow D}^{\text{In}}, \text{MAC}_{D,S})$$

where rnd is the request id received in the corresponding route request message, and $\text{MAC}_{D,S}$ is the message authentication code generated by D on the elements of the above message excluding $\text{addr}_D, \text{label}_{Z \rightarrow S}^{\text{Out}}$, and $\text{label}_{D \rightarrow D}^{\text{In}}$ using the pairwise key shared with S . Receiving this unicast message, Z first checks if D belongs to its neighborhood. If so, Z sends the message directly to node K , from which Z received the corresponding request message identified by rnd :

$$Z \rightarrow K : (\text{RREP}, \text{rnd}, \text{addr}_Z, \text{label}_{K \rightarrow S}^{\text{Out}}, \text{label}_{Z \rightarrow D}^{\text{In}}, \text{MAC}_{D,S}, \text{MAC}_{Z,K}^{\text{prv}})$$

Here, $\text{MAC}_{Z,K}^{\text{prv}}$ is the previous-hop MAC generated by Z on the elements of the message including $\text{addr}_Z, \text{label}_{K \rightarrow S}^{\text{Out}}$, and $\text{label}_{Z \rightarrow D}^{\text{In}}$. Following the same rules, all intermediate nodes perform the same steps that Z did (except that each intermediate node updates the previous-hop MAC). Finally, the reply reaches the source S , which then, after verifying the previous-hop MAC and $\text{MAC}_{D,S}$ in the reply message, can use the established route for data forwarding.

The reason that I do not use digital signatures to authenticate request and reply messages is that public key cryptography (PKC) incur a substantial computation overhead in sensor networks. In particular, PKC still falls behind the standard symmetric cryptography approaches in terms of computational performance; the verification of a digital signature is 3 orders of magnitude slower than MAC verification, while the signature generation is 4 orders of magnitude slower.

Let the *cost function* $\mathcal{C} : V \rightarrow \mathbb{R}$ assign the minimal delay of routing messages to each node in the network (i.e., the minimal delay that the particular node can incur in the travel of the message). We assume that $\mathcal{C}(v^*) = 0$ for all $v^* \in V^*$.

Before introducing the security objective function of label-switching routing in sensor networks, I introduce some definitions in order to ease its formalization.

Definition 7 (Anchor entry) An anchor entry $(v_{src}, v_{dst}, \mathbf{addr}_{nxt}, \mathit{label}_{v_{src} \rightarrow v_{dst}}^{\text{Out}}, \mathit{delay}_{v_{src}, v_{dst}})$ is the representation of a routing entry at source v_{src} , where the destination node is identified by v_{dst} , the next-hop towards the destination has (local) address \mathbf{addr}_{nxt} , the outgoing label of the source towards the destination is $\mathit{label}_{v_{src} \rightarrow v_{dst}}^{\text{Out}}$, and the delay of the quickest path through \mathbf{addr}_{nxt} to the destination is $\mathit{delay}_{v_{src}, v_{dst}}$.

Definition 8 (Intermediate entry) An intermediate entry $(v_{im}, \mathbf{addr}_{nxt}, \mathit{label}_{v_{im} \rightarrow v_{dst}}^{\text{In}}, \mathit{label}_{v_{im} \rightarrow v_{dst}}^{\text{Out}})$ is the representation of a routing entry at an intermediate node v_{im} , where the next-hop towards the destination has (local) address \mathbf{addr}_{nxt} , the incoming label and the outgoing label of v_{im} towards the destination are $\mathit{label}_{v_{im} \rightarrow v_{dst}}^{\text{In}}$ and $\mathit{label}_{v_{im} \rightarrow v_{dst}}^{\text{Out}}$, respectively.

Definition 9 (Matching property) A routing entry r_1 of node v_i matches a routing entry r_2 of node v_j ($i \neq j$), if

- the outgoing label of r_1 equals to the incoming label of r_2 ,
- the next-hop address of r_1 is used by v_j .

I recall that the state of the system is represented by the ensemble of all anchor and intermediate entries of all *honest* nodes.

Definition 10 (Correct state) A state is correct with respect to configuration $conf$, if for every anchor entry $r_0 = (v_{src}, v_{dst}, \mathbf{addr}_{nxt}, \mathit{label}_{v_{src} \rightarrow v_{dst}}^{\text{Out}}, \mathit{delay}_{v_{src}, v_{dst}})$, where $v_{src}, v_{dst} \in V \setminus V^*$, there exists a sequence of intermediate entries $r_i = (v_{\ell_i}, \mathbf{addr}_{nxt}, \mathit{label}_{v_{\ell_i} \rightarrow v_{dst}}^{\text{In}}, \mathit{label}_{v_{\ell_i} \rightarrow v_{dst}}^{\text{Out}})$ ($1 \leq i \leq d$) of honest nodes such that

- $v_{\ell_d} = v_{dst}$ and $\mathit{label}_{v_{dst} \rightarrow v_{dst}}^{\text{Out}}$ is an application identifier of v_{dst} ,
- $(v_{src}, v_{\ell_1}, \dots, v_{\ell_{d-1}}, v_{dst})$ is a workable path,
- let $v_{\ell_0} = v_{src}$,
 - if $v_{\ell_{i-1}}$ and v_{ℓ_i} are direct but not pseudo neighbors then r_{i-1} matches r_i ,
 - if $v_{\ell_{i-1}}$ and v_{ℓ_i} are pseudo neighbors then either r_{i-1} matches r_i , or the next-hop address of r_{i-1} belongs to a neighboring adversarial node,
- $\sum_{j=1}^{d-1} \mathcal{C}(v_{\ell_j}) \leq \mathit{delay}_{v_{src}, v_{dst}}$ (i.e., the delay of the discovered route between v_{src} and v_{dst} is not greater than the delay recorded in the routing (anchor) entry of v_{src})

Let the security objective function \mathcal{F} of secure label-switching routing return 0 for all pairs of system states and configurations that are incorrect, otherwise it returns 1.

Theorem 6 *Secure-TinyLUNAR* is a secure label-switching routing protocol for wireless sensor networks, if the MAC scheme is secure against chosen message attacks.

The proof is detailed in the dissertation. Note that intermediate nodes do not need to check the authenticity of the message origin, which means that the source do not need to use expensive global broadcast authentication methods based on asymmetric cryptography. Instead, Secure-TinyLUNAR uses pairwise MACs based on the more-energy conserving symmetric key cryptography for previous-hop (neighbor) and message origin authentication. Of course, the lack of a global broadcast authentication scheme exposes Secure-TinyLUNAR to various DoS attacks. However, note that using a one-way hash chain to authenticate the source is sufficient in this case. Moreover, hash chains have much less computational and communication overhead than other broadcast authentication schemes like digital signatures, one-time signatures, or μ Tesla [B1].

5 Application of New Results

The benefit of my work is twofold:

- First, I proposed a precise and general formal framework that has been proved to be useful in validating various security objectives of secure ad-hoc and sensor network routing protocols. Although the validation process has not been automated yet, a protocol designer can easily prove the security of any multi-hop routing protocols in wireless context by using the given proof technique. If the protocol is insecure in my model, the designer can find an attack by investigating where the proof fails.
- Second, I have identified several design principles of secure routing in wireless ad-hoc and sensor networks by proving the security of various routing protocols in my model. For instance, these include the requirements of route reply authentication in dynamic source routing, the per-hop authentication in dynamic distance vector routing, or the encryption of local topology information in link-state routing. Even if a protocol designer did not use any formal methods to validate a newly proposed secure routing protocol, following these principles he could avoid many potential vulnerabilities that are anyway difficult to identify by using informal reasoning exclusively.

Some of my results were used within an European research project called UbiSec&Sens (Ubiquitous Sensing and Security in the European Homeland, <http://www.ist-ubisecsens.org/>) between 2005 and 2008. The project was an IST STReP and received research funding from the European Community's Sixth Framework Programme. The primary objective of UbiSec&Sens was to provide a security and reliability architecture for medium and large-scale wireless sensor networks acting in volatile environments. In particular, it provided a complete toolbox of security and reliability aware components for sensor network application development. UbiSec&Sens' work is focused on the intersection of security, routing and in-network processing to design and develop efficient and effective security solutions and to offer effective means for persistent and encrypted data storage for distributed (and tiny) data base approaches. The solutions are prototyped and validated in the representative wireless sensor application scenarios of agriculture, road services and homeland security.

Another European research project focusing on the security of critical infrastructures is called WSan4CIP (Wireless Sensor and Actuator Networks for Critical Infrastructure Protection, <http://www.wsan4cip.eu/>), which is started at the beginning of 2009 and lasts for 3 years. This project is also an IST STReP and receives research funding from the European Community's Seventh Framework Programme. The project goals are to enhance the

reliability of critical infrastructures by providing surveillance data for the management of the critical infrastructure using wireless sensor and actuator networks (WSANs), and to increase the dependability of critical infrastructures security by providing self-healing and dependability modules for WSANs. The project also aims at providing appropriate tool support, and demonstrating the feasibility of this approach using energy generation and distribution as a representative of critical infrastructures. As such, the project will develop models for the reliability and security analysis of routing protocols for WSANs, which will be used to analyse various routing protocols both in terms of prevention of and reaction to attacks (the latter means attack detection and efficient recovery by reorganization of the overlay network topology). Here, prevention naturally includes the formal security validation of the route discovery phase of routing protocols which is the focus of my dissertation.

Publications of New Results

Book Chapters

- [B1] G. Ács and L. Buttyán,
Secure Routing in Wireless Sensor Networks,
Wireless Sensor Network Security (Cryptography and Information Security Series)",
Eds. J. Lopez and J. Zhou, ISBN: 978-1-58603-813-7, pages 154–203, IOS Press, 2008.

Journal Papers

- [J1] G. Ács and L. Buttyán,
Ad hoc útvonalválasztó protokollok bizonyított biztonsága,
Híradástechnika, 60(3):41–45, March 2005.
- [J2] G. Ács and L. Buttyán,
Provable Security for Ad Hoc Routing Protocols,
Híradástechnika (English Edition), 60(6):34–38, June 2005.
- [J3] G. Ács and L. Buttyán and I. Vajda,
Provably Secure On-demand Source Routing in Mobile Ad Hoc Networks,
IEEE Transactions on Mobile Computing, 5(11):1533–1544, 2006.
- [J4] G. Ács and L. Buttyán,
Útvonalválasztó protokollok vezeték nélküli szenzorhálózatokban,
Híradástechnika, 61(12):3–12, December 2006.
- [J5] G. Ács and L. Buttyán,
A taxonomy of routing protocols for wireless sensor networks,
Híradástechnika (English Edition), 62(1):32–41, January 2007.
- [J6] G. Ács and L. Buttyán,
Designing a Secure Label-switching Routing Protocol for Wireless Sensor Networks,
To appear in Periodica Polytechnica (<http://www.pp.bme.hu/>), December, 2008.

International Conference/Workshop Papers

- [C1] G. Ács and L. Buttyán and I. Vajda,
Provable Security of On-Demand Distance Vector Routing in Wireless Ad Hoc Networks,
In Proceedings of the Second European Workshop on Security and Privacy in Ad Hoc and Sensor Networks (ESAS), pages 113–127, Visegrád, Hungary, July 2005.

- [C2] G. Ács and L. Buttyán and I. Vajda,
Modelling Adversaries and Security Objectives for Routing Protocols in Wireless Sensor Networks,
In Proceedings of the Fourth ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN), pages 49–58, Alexandria, VA, USA, October 2006.
- [C3] G. Ács and L. Buttyán and I. Vajda,
The Security Proof of a Link-state Routing Protocol for Wireless Sensor Networks,
In Proceedings of the 3rd IEEE Workshop on Wireless and Sensor Networks Security (WSNS), pages 1–6, Pisa, Italy, October 2007.

Citations

- [J3] G. Ács and L. Buttyán and I. Vajda,
Provably Secure On-demand Source Routing in Mobile Ad Hoc Networks,
IEEE Transactions on Mobile Computing, 5(11):1533–1544, 2006.

is cited by

1. T. R. Andel,
Formal Security Evaluation of Ad-Hoc Routing Protocols,
PhD Dissertation, Florida State University, College of Arts and Sciences, 2007.
2. W. J. Tsaur, P. Haw-Tyng,
A new security scheme for on-demand source routing in mobile ad hoc networks,
In Proceedings of the International conference on Wireless communications and mobile computing (IWCMC), pp 577–582, 2007.
3. W. Tsaur, H. Pai,
A Secure On-Demand Source Routing Scheme Using Hierarchical Clustering in Mobile Ad Hoc Networks,
In Lecture Notes in Computer Science 4743: 513, Springer, 2007.
4. K. El-Defrawy, G. Tsudik,
ALARM: Anonymous Location-Aided Routing in Suspicious MANETs,
In Proceedings of the IEEE International Conference on Network Protocols (ICNP), pages 304–313, 2007.
5. S. Chakrabarti, S. Chandrasekhar, M. Singhal, K. Calvert
Authenticating DSR Using a Novel Multisignature Scheme Based on Cubic LFSR Sequences,
In Proceedings of The Fourth European Workshop on Security and Privacy in Ad hoc and Sensor Networks (ESAS), pages 156–171, 2007.
6. M. Burmester, B. de Medeiros,
On the Security of Route Discovery in MANETs,
To appear in IEEE Transactions on Mobile Computing, 2009.
7. T. R. Andel, A. Yasinsac,
Automated Security Analysis of Ad Hoc Routing Protocols
In Proceedings of the Workshop on Foundations of Computer Security and Automated Reasoning for Security Protocol Analysis, 2007.
8. J. van der Merwe, D. Dawoud, S. McDonald,
Key Distribution in Mobile Ad Hoc Networks Based on Message Relaying,
In Proceedings of The Fourth European Workshop on Security and Privacy in Ad hoc and Sensor Networks (ESAS), pages 87–100, 2007.
9. X. Su, R. V. Boppana,
On identifying malicious nodes in ad hoc networks,
In Proceedings of the International Conference on Wireless Communications and Mobile Computing, pages 254–259, 2007.

10. D. Djenouri, O. Mahmoudi, M. Bouamama, D. Llewellyn-Jones, M. Merabti,
On Securing MANET Routing Protocol Against Control Packet Dropping,
In Proceedings of the IEEE International Conference on Pervasive Services (ICPS),
pages 100–108, 2007.
11. R.V. Boppana, X. Su,
Secure Routing Techniques to Mitigate Insider Attacks in Wireless Ad Hoc Networks,
In Proceedings of the IEEE Wireless Hive Networks Symposium, 2007.
12. T. R. Andel, A. Yasinsac,
Surveying security analysis techniques in manet routing protocols,
In IEEE Communications Surveys & Tutorials 9: (4) 70–84, 2007.
13. T. R. Andel, A. Yasinsac,
Adaptive Threat Modeling for Secure Ad Hoc Routing Protocols,
In Proceedings of the 3rd International Workshop on Security and Trust Management (STM), pages 3–14, 2008.
14. X. Su, R. V. Boppana,
Crosscheck mechanism to identify malicious nodes in ad hoc networks,
In Security and Communication Networks, John Wiley & Sons Ltd., 2: (1) 45-54,
2008.
15. A. König, M. Hollick, T. Krop, R. Steinmetz,
GeoSec: quarantine zones for mobile ad hoc networks,
In Security and Communication Networks, John Wiley & Sons Ltd., 2008.
16. X. Su, R. V. Boppana,
Mitigation of colluding route falsification attacks by insider nodes in mobile ad hoc networks,
Wireless Communications and Mobile Computing, John Wiley & Sons Ltd., 2008.
17. K. El-Defrawy, G. Tsudik,
PRISM: Privacy-friendly routing in suspicious MANETs (and VANETs),
In Proceedings of the IEEE International Conference on Network Protocols (ICNP),
pages 258–267, 2008.
18. M. Fanaei, M. Berenjkoub, A. Fanian,
Resistant TIK-Based endairA Against the Tunneling Attack,
In Proceedings of the 10th International Conference on Advanced Communication
Technology, pages 1461–1466, 2008.
19. Q. Li, Y.-C. Hu, M. Zhao, A. Perrig, J. Walker, W. Trappe,
SEAR: a secure efficient ad hoc on demand routing protocol for wireless networks,
In Proceedings of the ASIAN ACM Symposium on Information, Computer and
Communications Security (ASIA CCS), pages 201–204, 2008.
20. M. Poturalski, P. Papadimitratos, J.-P. Hubaux,
Secure neighbor discovery in wireless networks: formal investigation of possibility,
In Proceedings of the ASIAN ACM Symposium on Information, Computer and
Communications Security (ASIA CCS), pages 189–200, 2008.

21. J. Kim, G. Tsudik,
SRDP: Secure route discovery for dynamic source routing in MANETs,
In Elsevier Ad Hoc Networks, 2008.
22. S. Eidenbenz, G. Resta, P. Santi
The Commit protocol for truthful and cost-efficient routing in ad hoc networks with selfish nodes,
In IEEE Transactions on Mobile Computing 7: (1) 19–33, 2008.
23. M. Poturalski, P. Papadimitratos, J.-P. Hubaux,
Towards provable secure neighbor discovery in wireless networks,
In Proceedings of the ACM Workshop on Formal Methods in Security Engineering (FMSE), pages 31–42, 2008.
24. M. Moe, B. E. Helvik, S. J. Knapskog,
TSR: trust-based secure MANET routing using HMMs,
In Proceedings of the 4th ACM Symposium on QoS and Security for Wireless and Mobile Networks, pages 83–90, 2008.
25. Y. Ren, A. Boukerche,
ARMA: a scalable secure routing protocol with privacy protection for mobile ad hoc networks,
Wireless Communications and Mobile Computing, John Wiley & Sons Ltd., 2009.
26. F. Mohammad, M. Berenjkoub,
Prevention of Tunneling Attack in endairA,
In Proceedings of the Advances in Computer Science and Engineering, Communications in Computer and Information Science (ACSE), pages 994–999, 2009.

[J5] G. Ács and L. Buttyán,
A taxonomy of routing protocols for wireless sensor networks,
Híradástechnika (English Edition), 62(1):32–41, January 2007.

is cited by

1. E. Osipov,
tinyLUNAR: One-Byte Multihop Communications Through Hybrid Routing in Wireless Sensor Networks,
In Proceedings of the Next Generation Teletraffic and Wired/Wireless Advanced Networking, pages 379–392, 2007.

[C1] G. Ács and L. Buttyán and I. Vajda,
Provable Security of On-Demand Distance Vector Routing in Wireless Ad Hoc Networks,
In Proceedings of the Second European Workshop on Security and Privacy in Ad Hoc and Sensor Networks (ESAS), pages 113–127, Visegrád, Hungary, July 2005.

is cited by

1. J. Kong, X. Hong, M. Gerla,
Modeling Ad-hoc rushing attack in a negligibility-based security framework,

- In Proceedings of the 5th ACM Workshop on Wireless Security (WiSe), pages 55–64, 2006.
2. M. Wen, L. Dong, Y. Zheng, K. Chen,
A Framework for Proving the Security of Data Transmission Protocols in Sensor Network,
 In Lecture Notes in Computer Science, Intelligence and Security Informatics: 288–294, 2007.
 3. X. Su, R. V. Boppana,
On identifying malicious nodes in ad hoc networks,
 In Proceedings of the International Conference on Wireless Communications and Mobile Computing, pages 254–259, 2007.
 4. J. Eriksson, M. Faloutsos, S. V. Krishnamurthy,
Routing amid Colluding Attackers,
 In Proceedings of the IEEE International Conference on Network Protocols, pages 184–193, 2007.
 5. X. Su, R. V. Boppana,
Crosscheck mechanism to identify malicious nodes in ad hoc networks,
 In Security and Communication Networks, John Wiley & Sons, 2: (1) 45-54, 2008.
 6. M. Burmester, B. de Medeiros,
On the Security of Route Discovery in MANETs,
 To appear in IEEE Transactions on Mobile Computing, 2009.
- [C2] G. Ács and L. Buttyán and I. Vajda,
Modelling Adversaries and Security Objectives for Routing Protocols in Wireless Sensor Networks,
 In Proceedings of the Fourth ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN), pages 49–58, Alexandria, VA, USA, October 2006.
- is cited by*
1. W. Wei-ping, C. Liang, W. Jian-xin,
A Source-Location Privacy Protocol in WSN Based on Locational Angle,
 In Proceedings of the IEEE International Conference on Communications (ICC), pages 1630–1634, 2008.
 2. M. Burmester, B. de Medeiros,
On the Security of Route Discovery in MANETs,
 To appear in IEEE Transactions on Mobile Computing, 2009.
- [C3] G. Ács and L. Buttyán and I. Vajda,
The Security Proof of a Link-state Routing Protocol for Wireless Sensor Networks,
 In Proceedings of the 3rd IEEE Workshop on Wireless and Sensor Networks Security (WSNS), pages 1–6, Pisa, Italy, October 2007.

is cited by

1. L. Tobarra, D. Cazorla, F. Cuartero, J. Pardo,
Modelling secure wireless sensor networks routing protocols with timed automata,
In Proceedings of the 3rd ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks, pages 51–58, 2008.

The total number of known independent citations is 36.

References

- [Backes and Pfitzmann, 2004] M. Backes and B. Pfitzmann. A cryptographically sound security proof of the needham-schroeder-lowé public-key protocol. *IEEE Journal on Selected Areas of Computing (JSAC)*, 22(10):2075–2086, 2004.
- [Bellare *et al.*, 1998] M. Bellare, R. Canetti, and H. Krawczyk. A modular approach to the design and analysis of authentication and key exchange protocols. In *Proceedings of the ACM Symposium on the Theory of Computing*, 1998.
- [Bryan *et al.*, 2005] P. Bryan, A. Perrig, and V. Gligor. Distributed detection of node replication attacks in sensor networks. In *Proceedings of the IEEE Symposium on Security and Privacy*, 2005.
- [Burmester and de Medeiros, 2007] M. Burmester and B. de Medeiros. Towards provable security for route discovery protocols in mobile ad hoc networks, 2007.
- [Deng *et al.*, 2002] J. Deng, R. Han, and S. Mishra. INSENS: Intrusion-tolerant routing in wireless sensor networks. Technical Report CU-CS-939-02, Department of Computer Science, University of Colorado, 2002.
- [Dolev and Yao, 1981] D. Dolev and A. C. Yao. On the security of public key protocols. In *Proceedings of the IEEE 22nd Annual Symposium on Foundations of Computer Science*, pages 350–357, 1981.
- [Douceur, 2002] J. R. Douceur. The sybil attack. In *Proceedings of the International Workshop on Peer-to-Peer Systems (IPTPS)*, 2002.
- [Hill *et al.*, 2000] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister. System architecture directions for networked sensors. In *Proceedings of the International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, 2000.
- [Hu *et al.*, 2002] Y.-C. Hu, A. Perrig, and D. Johnson. Ariadne: A secure on-demand routing protocol for ad hoc networks. In *Proceedings of ACM Conference on Mobile Computing and Networking (Mobicom)*, 2002.
- [Hu, 2003] Yih-chun Hu. Rushing attacks and defense in wireless ad hoc network routing protocols. In *Proceedings of ACM Workshop on Wireless Security (WiSe)*, pages 30–40, 2003.
- [Karlof and Wagner, 2003] C. Karlof and D. Wagner. Secure routing in wireless sensor networks: attacks and countermeasures. *Ad Hoc Networks*, 1, 2003.
- [Kim and Tsudik, 2005] J. Kim and G. Tsudik. Securing route discovery in dsr. In *Proceedings of the IEEE Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous)*, July 2005.
- [Mao, 2004] W. Mao. *Modern Cryptography: Theory and Practice*. Prentice Hall PTR, 2004.

- [Marshall, 2003] J. Marshall. An analysis of the secure routing protocol for mobile ad hoc network route discovery: using intuitive reasoning and formal verification to identify flaws. msc thesis, department of computer science, florida state university, 2003.
- [Osipov, 2007] E. Osipov. tinylunar: One-byte multihop communications through hybrid routing in wireless sensor networks. In *Proceedings of the 7th International Conference on Next Generation Teletraffic and Wired/Wireless Advanced Networking (NEW2AN 2007)*, 2007.
- [Papadimitratos and Haas, 2002] P. Papadimitratos and Z. Haas. Secure routing for ad hoc networks. In *Proceedings of SCS Communication Networks and Distributed Systems Modelling Simulation Conf. (CNDS)*, 2002.
- [Perkins and Royer, 1999] C. Perkins and E. Royer. Ad hoc on-demand distance vector routing. In *Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications*, 1999.
- [Perrig *et al.*, 2002] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar. SPINS: Security protocols for sensor networks. *Wireless Networks Journal (WINE)*, 8, 2002.
- [Pfitzman and Waidner, 2001a] B. Pfitzman and M. Waidner. A model for asynchronous reactive systems and its application to secure message transmission. In *Proceedings of the 22nd IEEE Symposium on Security & Privacy*, 2001.
- [Pfitzman and Waidner, 2001b] B. Pfitzman and M. Waidner. A model for asynchronous reactive systems and its application to secure message transmission. In *Proceedings of the 22nd IEEE Symposium on Security & Privacy*, 2001.
- [Sanzgiri *et al.*, 2002] K. Sanzgiri, B. Dahill, B. Levine, C. Shields, and E. Belding-Royer. A secure routing protocol for ad hoc networks. In *Proceedings of the International Conference on Network Protocols (ICNP)*, 2002.
- [Shoup, 1999] V. Shoup. On formal models for secure key exchange (version 4). Technical report, revision of IBM Research Report RZ 3120, 1999.
- [Yang and Baras, 2003] S. Yang and J. Baras. Modeling vulnerabilities of ad hoc routing protocols. In *Proceedings of the ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN)*, 2003.
- [Zapata and Asokan, 2002] M. G. Zapata and N. Asokan. Securing ad hoc routing protocols. In *Proceedings of the ACM Workshop on Wireless Security (WiSe)*, 2002.
- [Ács *et al.*, 2005] G. Ács, L. Buttyán, and I. Vajda. Provably secure on-demand source routing in mobile ad hoc networks. *Technical Report, CrySyS Lab, Budapest University of Technology and Economics. Also available at <http://eprint.iacr.org/> under report number 2004/159.*, April 2005.
- [Çamtepe and Yener, 2005] S. A. Çamtepe and B. Yener. Key distribution mechanisms for wireless sensor networks: a survey. Technical Report TR-05-07, Rensselaer Polytechnic Institute, Computer Science Department, 2005.