NORTHWESTERN
UNIVERSITY

Electrical Engineering and Computer Science Department

Technical Report
NWU-EECS-11-05
May 2, 2011
Security of Electrostatic Field Persistent Routing: Taxonomy of Attacks and Defense Mechanisms

Oliviu C. Ghica[1] Cristina Nita-Rotaru[2] Goce Trajcevski[1] Peter Scheuermann[1]

(1) Department of Electrical Engineering and Computer Science
Northwestern University, Evanston, IL
{ocg474,goce,peters}@eecs.northwestern.edu

(2) Department of Computer Science,
Purdue University, West Lafayette, IN
crisn@cs.purdue.edu

Abstract

Electrostatic field-based routing (EFR) is a form of geographical multi-path routing where packets are routed along a collection of electrostatic field lines, defined by electrostatic charges associated with source and sink nodes. EFR provides an efficient and scalable solution to the workload-balancing problem. However, it assumes that the nodes behave in a cooperative manner. Since wireless sensor nodes may be deployed in adversarial environments, EFR-based routing protocols can be subject to various attacks.
In this article, we investigate the security aspects of EFR-based routing protocols. More specifically, we focus on an instance of EFR, called Multi-Pole Field Persistent Routing (MP-FPR), for which we identify the categories of attacks that can target different components of the protocol, and propose a set of corresponding lightweight defense mechanisms. We are motivated by the observation that, while certain categories of attacks can be mounted with little resource-effort, they can be highly destructive to system performance and its workload balanced operation. We present extensive experimental evaluations of the impact of the different attacks and the effectiveness of the proposed defense mechanisms for various components of the MP-FPR protocol.

Keywords: Wireless Sensor Networks, Multipath Routing, Security.

Security of Electrostatic Field Persistent Routing: Taxonomy of Attacks and Defense Mechanisms

Oliviu C. Ghica Cristina Nita-Rotaru Goce Trajcevski Peter Scheuermann

***Abstract*—Electrostatic field-based routing (EFR) is a form of geographical multi-path routing where packets are routed along a collection of electrostatic field lines, defined by electrostatic charges associated with source and sink nodes. EFR provides an efficient and scalable solution to the workload balancing problem. However, it assumes that the nodes behave in a cooperative manner. Since wireless sensor nodes may be deployed in adversarial environments, EFR-based routing protocols can be subject to various attacks.**

In this article, we investigate the security aspects of EFR-based routing protocols. More specifically, we focus on an instance of EFR, called Multi-Pole Field Persistent Routing (MP-FPR), for which we identify the categories of attacks that can target different components of the protocol, and propose a set of corresponding lightweight defense mechanisms. We are motivated by the observation that, while certain categories of attacks can be mounted with little resource-effort, they can be highly destructive to system performance and its workload balanced operation. We present extensive experimental evaluations of the impact of the different attacks and the effectiveness of the proposed defense mechanisms for various components of the MP-FPR protocol.

I. Introduction

Wireless Sensor Networks (WSN) [11] have emerged as a promising paradigm for many application domains that require combined capabilities of sensing, processing, and communication in different physical environments. Given the resource-constraints of the individual nodes (energy, bandwidth, etc), one of the problems that has generated a large number of research results in the recent years is the problem of efficient *routing* in WSN settings [10].

In a typical WSN application, a user-initiated query is disseminated to the appropriate *source* nodes where the data of interest is locally collected. The resulting point-to-point data-stream is relayed back to a remote *sink* node which, in turn, interfaces with the user. Many routing protocols for WSN are designed under the location-aware assumption and rely on the *geography*-based (greedy) routing principle, according to which packets are forwarded to nodes that are physically closer to a given destination [46]. A specific type of geographic routing is trajectory based forwarding (TBF) [57], in which packets are routed towards the intended destinations along pre-defined ”virtual” trajectories. Such trajectories resemble the behavior of various physical fields [72].

Electrostatic Field-based Routing (EFR) [56] is a multi-path routing protocol that reduces the complexity of determining and managing the collection of underlying trajectories by representing them as electrostatic field lines, rather than relying on geometric models. The field lines originate at source nodes, where the data is produced, and lead towards a designated sink node, where the data is being consumed. The main advantage of EFR as a multi-path routing protocol is that it creates *implicitly* spatially *disjoint trajectories* – a consequence of the disjointness property of electrostatic field lines. EFR has small computational and communication overheads which are associated with performing local forwarding decisions. EFR is also a form of gradient-based routing, inspired by several field-based approaches [47], [43] in the context of sensor networks [48] and mesh networks [17]. EFR achieves workload balancing in dense and uniformly distributed networks. In networks where this assumption does not hold, path-merging can occur reducing the workload balancing capabilities. Multi-Pole Field Persistent Routing (MP-FPR) protocol [73] extends EFR’s applicability to less-dense and often non-uniform network distributions by actively seeking to separate any merged paths, whenever network conditions allow.

MP-FPR is based on the assumption that nodes in the network always operate correctly. Such assumption is no longer valid when MP-FPR is deployed in an adversarial environment. As many applications for WSNs require deployment in adversarial environments, it is critical to provide mechanisms to ensure that routing protocols operate correctly and securely.

In this article we analyze the resilience of the MP-FPR protocol in adversarial environments and identify the main *data*- and *control*-level components that can be exploited by an attacker. We study not only disruptions to the users’ data streams, but also disruptions to the system-wide performance and resource-utilization as a result of a network attack. For example, we are interested in the disruption of the *load-balancing* performance that MP-FPR is designed to provide if certain protocol components are compromised. We quantify the severity and likelihood of different attacks by taking into consideration the relative easiness of their staging, and we identify solutions to prevent or mitigate their effects. In summary, our main contributions are:

- We identify a set of potential security risks factors in MP-FPR and assess their impact on the entire system. Specifically, we first identify a set of *control-level* attacks: *path deflection*, *path diversity deflation*, *family path intersection wild-path* and *field-line hopping* attacks, all of which are specific to electrostatic-field based routing. These attacks are carried through the control messages in MP-FPR, and can lead to quality of service degradation by disrupting the workload-balancing operation. We next identify a set of *data-level* attacks: *data denial of service* (DoS), *data pollution*, and *data stream invalidation*

attacks, which directly target users' payload-data.

- We evaluate analytically and empirically the resilience of MP-FPR to adversarial scenarios and observe the epidemic character of several attacks as a primary focus for the defense mechanisms. Epidemic attacks can yield significant performance degradation with minimal staging efforts. For example, a *single* attack consisting of inserting eight forged charges in the system via a sink node can nearly double the standard deviation of the residual energy levels – a representative metric for describing the workload balancing performance.
- We propose two classes of defense mechanisms, one addressing the integrity and authentication of the MP-FPR messages, and the second one providing resilience against selective forwarding of various protocol messages. Specifically, we analyze and compare the cost-effectiveness of three types of cryptographic solutions: PIKE, DS/ECC and TESLA, and justify our selection for the MP-FPR protocol. Subsequently, we propose two multi-path solutions, k-EF and k-RPEF, in the electrostatic context, to address the selective forwarding problem, and a complementary *path diversity monitoring scheme* (PDMS) to provide closed-loop control over path diversity. We report the quantitative observations regarding the effectiveness of the proposed approaches based on an extensive set of experimental evaluations.

Outline. The rest of the article is organized as follows. In Section II we overview the main aspects of the EFR and MP-FPR multipath routing protocols. The details of the adversarial model are presented in Section III and an outline of the proposed countermeasures is discussed in Section IV. Section V overviews several cryptographic approaches that can provide integrity verification support to MP-FPR, and a corresponding overhead and feasibility analysis is provided in Section VI. Resilience mechanisms against attacks carried through selective message forwarding is presented in Section VII. The results of our experimental investigation are presented in Section VIII. We overview the related work in Section IX and conclude the article in Section X.

II. Multi-Pole Field Persistent Routing

In this section we first describe the network and application models we assume in this work. We then present an overview of the EFR routing protocol and provide a detailed description of the improved MP-FPR protocol in the context of WSNs.

A. Network and Application Model

We assume that a given network consists of a set **SN** = $\{sn_1, sn_2, \ldots, sn_n\}$ of n wireless sensor nodes, each capable of acting both as a *relay* and a *source* of sensed data. Users formulate queries specifying properties of the data stream that is to be collected from a particular geographic location, and submit them via *sink* nodes, which act as gateways between the user and the sensor network. Queries are relayed to specific nodes in charge of their processing, i.e. the source nodes, and the resulting, possibly long-term, data stream is collected and relayed back to the sink. In order to promote workload balancing, multiple paths are established between the source and sink end-points and the transmission of individual packets alternates among the different paths.

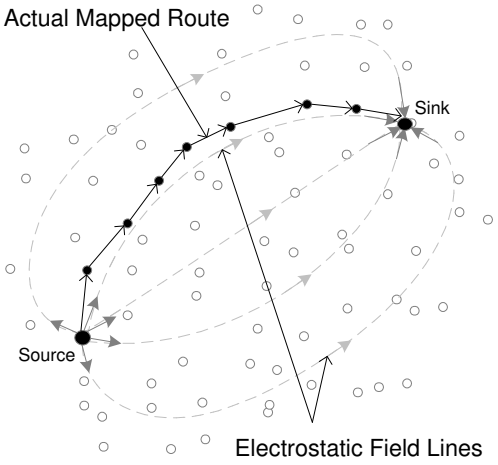


Fig. 1. Mapping of routes to electrostatic field lines with EFR routing. Due to finite distributions, the actual route cannot be precisely mapped to a field line and, in reality, it can deviate

B. Electrostatic Field-based Routing

Electrostatic field-based routing is a form of trajectory-based routing where the spatial trajectories are represented via *electrostatic field lines*. The field lines originate at source nodes, which are assigned a "positive charge", and terminate at designated sink nodes which are assigned a "negative charge". In order for a particular relay-node to know how to route a packet towards the sink, all it needs to know is the location and the electrostatic charge information of the source and sink nodes, as well as its own location.

In essence, EFR works as follows. Given the position and the assigned charge of the sink, a source node probes several paths, each of which is constructed on-the-fly along different electrostatic field lines between that source and the sink. The sink will acknowledge certain paths that meet a particular criteria, i.e. length and/or measured delay incurred along a path. Each acknowledgement identifies a different path (along a different electrostatic field line) and only acknowledged paths will be subsequently used by the source node to transmit data-packets towards the sink. A given current relay node in the multi-hop sequence from the source towards the sink needs to select a subsequent relay node from among its 1-hop neighbors. The selection criteria amounts to finding a neighbor which has the smallest deviation, if any, from the field line the current relay node belongs to, as well as providing the maximum advancement of the packets towards the sink. Figure 1 depicts an instance of a route built along a specific electrostatic field line.

One characteristic present in EFR is that permanent path deviations may occur when a given relay node cannot find subsequent relay node(s) that are along or in the immediate

vicinity of a particular electrostatic field line. As a consequence, two or more adjacent paths may intersect and/or merge, resulting in overloading a subset of the downstream relay nodes. While this phenomenon cannot be avoided, especially in sparser networks, a particular drawback of EFR is that it cannot recover from this condition once it occurred, i.e. it does not attempt redistributing previously-merged paths when the network conditions allow.

C. Multi-Pole Field Persistent Routing

MP-FPR is an extension of EFR, which overcomes the limitation of re-creating spatially disjoint routes via splitting previously merged routes. Unlike EFR where packets travel only along field lines that the current relay node resides on, in MP-FPR packets will travel along the original field line from which a packet may have been diverted. MP-FPR piggybacks the identity of a given field line on data-packets. This, in turn, is subsequently used by the relay nodes to determine the *original* field line which will be given priority for that particular packet. Figure 2(b) and 2(c) illustrate the path merging and recovery process. Figures 3(a) and 3(b) illustrate the benefits in terms of diversity of routes obtained via MP-FPR in comparison to EFR.

All messages used by MP-FPR are sent using two basic forwarding mechanisms: Electrostatic Field (EF) forwarding which relies on electrostatic fields and Shortest Geographical Path (SGP) which is a greedy based geographical routing.

EF forwarding: MP-FPR uses for routing a discrete subset of field lines out of the infinite number of field lines that can be established between a given *(source, sink)* pair. We refer to this set S_f as a family of paths. Figure 2(a) illustrates a family of field lines established between a source and a sink node. Each field line in S_f is uniquely identified by the value of the angle φ_j, determined by the *tangent* to a given/chosen field line at the source, and the line segment between the source and the sink[1] . For example, assuming a uniform selection of the tangential-angle from the interval $[0, \pi]$, a particular field line φ_j can be chosen from a field line set $S_f = \{k\frac{2\pi}{N_r} \mid k = \overline{1, N_r}\}$, where N_r represents the desired cardinality of the family of routes S_f.

Every node sn_i in the network can determine the tangent angle $\varphi_j \in S_f$ of the field line that it *actually* belongs to based on the (1) location and charge information of the source(s), (2) location and charge information of the sink, and (3) its own location. Once sn_i receives a packet, the information about the field line that the packet is *supposed* to be forwarded along, i.e. φ_j, is piggy-backed to the packet as part of the field line persistency mechanism. From a routing perspective, each route built along a particular field line φ_j is uniquely identified by a route index parameter, denoted r_j. For simplicity, we assume $r_j = \varphi_j$. Given this information, a particular relay node will select, as its subsequent relay node, one of its 1-hop neighbors which exhibits the smallest field line deviation $|\varphi_j - \varphi_i|$, where φ_i represents the actual field line a downstream relay sn_i actually resides on, and it is furthest away towards the sink (cf. [73]).

[1]Note that the cardinality of the S_f, as well as the criteria for selecting a particular φ_j can be user-specified.

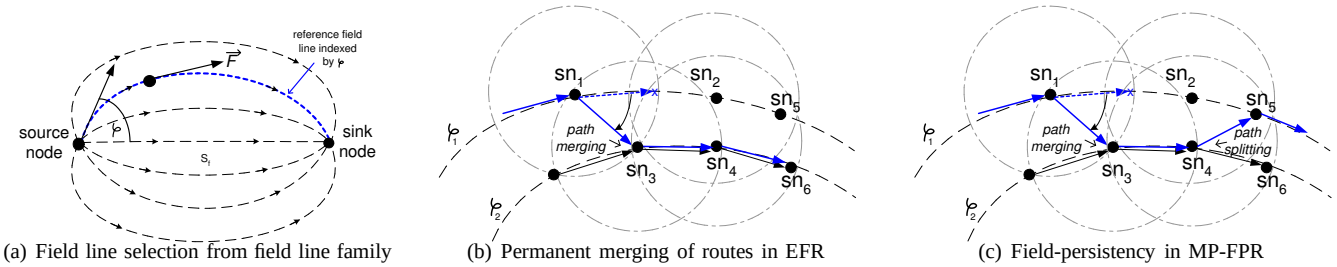
SGP forwarding: MP-FPR partly relies on a greedy geographic routing mechanism similar to BVR [27], where packets are sent via a geographically shortest path towards a known physical destination. In MP-FPR nodes determine their own position via a lightweight localization service external to the routing protocol (see [34] for a survey), as well as the position of their 1-hop neighbors through a periodic location information exchange.

MP-FPR consists of the following protocol components: *query dissemination and charge allocation*, *route establishment*, and *data forwarding*. Below we provide an overview of each component and summarize the type and content of the messages used by the protocol in Table I.

Query dissemination and charge allocation: This protocol component consists of messages generated by the sink and has several goals. First is to forward the user query towards the source and is achieved through a QUERY message sent by the sink with SGP forwarding towards L_{src} – the location within the area where data relevant to the query should be collected from. A sensor node which is geographically closest to L_{src} will assume the role of the source for the given QUERY message and initiate its processing. Second goal is to disseminate electrostatic charges information, which consists of a set of (location, magnitude and expiration) information associated with each routing end-point, i.e. source or sink node, in the network. For example, if there are m source nodes relaying data-streams to a common sink, the QUERY message contains a set $C_e = \{e_{snk}\} \cup \{e_i | i \in \overline{1, m}\}$ of electrostatic charges. Third goal is to limit the number of alternative paths to be built in order to correspondingly bound the duration of the route establishment protocol component. We refer to this limit as the *path diversity quota*, and it can be either user specified or system predefined. Path diversity quota is controlled via a numerical parameter $N_r = |S_f|$ embedded in the body of the QUERY message.

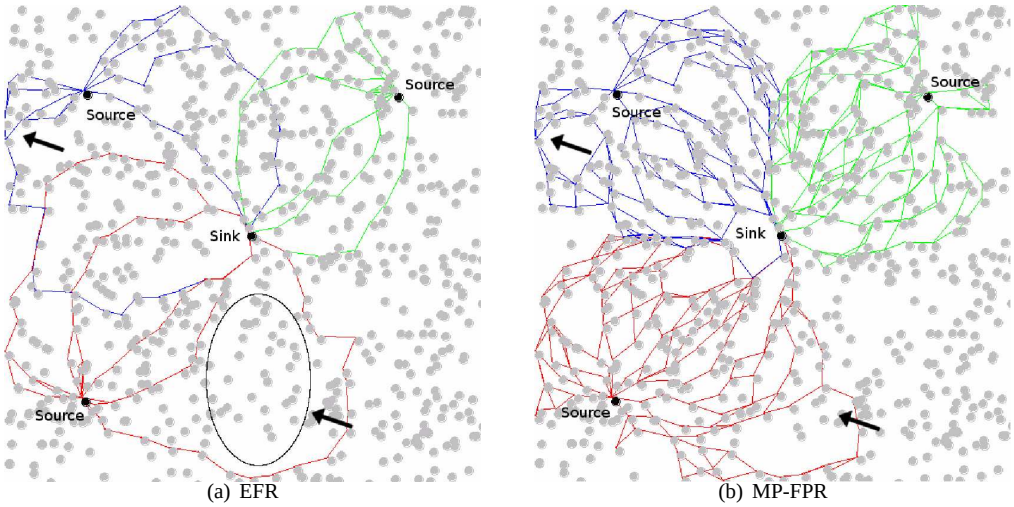
Whenever a new data source is added to the existing set of source-nodes, a new corresponding charge is added to the virtual electrostatic field. The charge information is being updated at each of the source nodes via an UPDATE message. For example, if there were m different sources in the network, excluding the newest activated one by the last QUERY message, then m UPDATE messages are sent via the SGP forwarding mechanism to each of the m existing source nodes. Upon receiving an UPDATE, the route establishing process is re-initiated by the source nodes in order to establish new families of routes that are consistent with the new charge distribution.

By convention, positive charges are associated with source nodes and negative charges with sink nodes. Thus, the



(a) Field line selection from field line family (b) Permanent merging of routes in EFR (c) Field-persistency in MP-FPR

Fig. 2. MP-FPR mechanism. (a) Sample family of multiple field lines between a source and a sink node, used for alternate path routing; selection of an arbitrary angle φ and associated incidental reference field line that is followed by the corresponding indexed route; (b) Path merging in sparser areas: node sn_1 is unable to reach node sn_2 and redirects the route to node sn_3, which is already servicing another route r_2 associated to field line φ_2 (c) Un-merging previously merged paths in MP-FPR: node sn_4 redirects the route r_1 that went through sn_1 to sn_5 to resume routing along φ_1



(a) EFR (b) MP-FPR

Fig. 3. Path merging and boundary effects in EFR vs. MP-FPR in low density networks. MP-FPR consistently achieves richer and more evenly distributed families of routes. As it may be observed at the arrowed pointers, path merging effects are not permanent in MP-FPR, as path splitting does occur when possible. Path merging effects are also visible in the circled area, where a coverage hole at the bottom of the network leads to a larger un-utilized relay area in EFR

direction of the data flow is consistent with the direction of the electrostatic field vectors, i.e. originating at a source and converging towards a sink node. Conform [73], the sink node's charge magnitude is equated to the sum of magnitudes of the all charges associated with the source nodes, i.e. $|e_{snk}| = |\sum_{i=1}^{m} e_i|$.

Route establishment: Initiated upon receiving a QUERY or UPDATE message at a source node, the *route establishment* is a two-phase, request-acknowledgment process. During the *requesting phase*, the source transmits a set of RREQ messages along *distinct* electrostatic field lines towards the sink. A RREQ message carries a list of network's current charges C_e as well as the field line index (equivalently route index) $r_i \in S_f$ identifying the field line a specific RREQ message is to be sent along. To amortize the associated transmission cost of the charges, this information is sent only once along RREQ messages, and cached locally by the relay nodes along a route; subsequent DATA messages will not carry them. The source node will also incorporate its actual location information L_{src} in the RREQ message such that sink's maintains a more accurate representation of the actual sources. Note that the actual source's location may not coincide with the user-specified location within the QUERY message due to finite coverage of the deployment area. A timestamp t_{sent} is also included in the RREQ message to assist in determining the quality (e.g. latency) of a specific route. We assume that nodes have loosely synchronized clocks [70].

If, upon receiving a RREQ message, it is determined that RREQ's route exhibited an admissible latency, the route is acknowledged, during the *acknowledgment phase*, by sending back a corresponding ACK message to the specific source. The route index r_i corresponding to the route that is being acknowledged is included in the ACK message. Note that ACK messages are sent back via the SGP mechanism towards the actual location of the source L_{src}, and not via EF mechanism the corresponding RREQ message was sent. The reason for which ACK messages are using the SGP mechanism comes from a simplicity and energy-efficiency perspective: SGP provides the smallest energy overhead and the fastest packet delivery; ACK messages are infrequently used, thus the energy imbalance caused by ACK messages is

TABLE I
MP-FPR MESSAGES

Type	Originator	Recipient	Functionality	Protocol Phase	Forwarding Mechanism	Fields of Interest
QUERY	Sink	Sources	Query Specification Wrapper	Query Dissemination and Charge Allocation	SGP	L_{src}, C_e, N_r,
UPDATE	Sink	Sources	Charge Information Update	Query Dissemination and Charge Allocation	SGP	L_{src}, C_e
RREQ	Sources	Sink	Route Request (Probe)	Route Establishment	EF	L_{src}, C_e, r_i, t_{sent}
ACK	Sink	Sources	Route Acknowledgment	Route Establishment	SGP	L_{src}, r_i
DATA	Sources	Sink	User Data-Payload Wrapper	Data Forwarding	EF	r_i, $Data$

negligible and does not justify building multiple paths under the original MP-FPR's assumptions. Every acknowledged route is added to a source-maintained set of acknowledged routes $S_f^{ack} \subseteq S_f$, i.e., a pool of routes that are available for data forwarding.

Data forwarding: The DATA messages pertaining to a data-stream as a result of query processing are forwarded back to the sink node. DATA messages, which contain user specified information as payload, are forwarded in an alternating manner among the individual routes r_i from the set of acknowledged routes S_f^{ack}, via the EF mechanism.

III. TAXONOMY OF ATTACKS

In this section we identify a representative set of attacks that can be carried against the MP-FPR protocol. In particular, we focus on attacks that exploit vulnerabilities introduced by the use of electrostatic field lines and by the field persistency mechanism. Several attacks require minimal effort from the attacker, but can severely impact the performance, user experience, and energy efficiency/consumption patterns.

MP-FPR is a network-layer protocol, consequently we consider only attacks carried at this layer. We proceed with presenting MP-FPR's goals and the network-level adversarial model, followed by the details of each identified attacks.

A. MP-EFR System Goals

MP-FPR has two main system goals that can be compromised by attacks:

- Increase network lifetime by promoting delivery of the data stream in a workload balanced manner.
- Ensure certain soft QoS guarantees, such as bounded end-to-end data stream delivery latencies, with respect to user's data stream.

MP-FPR promotes workload balancing by route alternation, as well as maintaining rich path diversities between two endpoints of a data-stream. Balancing the load correlates to balancing in-network energy consumption, equivalently reducing both the likelihood and severity of hot-spots, with a net result observed in network overall operational lifetime. When it comes to performance of the data-stream deliverability, the MP-FPR protocol does not impose a policy for handling parts of the data stream that violate QoS contracts. However, in this work we assume that such data is treated by the user as outdated and subsequently discarded, i.e. considering that the data-stream may feed into a user-level real-time application outside of the network. Consequently, from an user perspective, any compromise to the timely-deliverability of the data-stream is considered in this work as a compromise of the data-stream.

B. Adversarial Model

We assume that the only trusted nodes in the network are the sink and the source nodes. We also assume that honest nodes participate correctly in the routing protocol, whereas malicious nodes may act alone or in collusion with other malicious nodes. We refer to any arbitrary action of authenticated nodes resulting in the disruption of the routing service as Byzantine behavior, and to such an adversary as a Byzantine adversary. Examples of Byzantine behavior include: dropping, delaying, modifying or replaying packets.

We assume the forwarding mechanisms employed by MP-FPR, i.e. EF and SGP are not secure. However, since there already exists a body of work addressing the security aspects of the SGP mechanism [23], we focus mainly on the EF mechanism, and only touch-base with the vulnerabilities of SGP when necessary.

Both EF and SGP rely on a localization service. We assume security mechanisms [76], [68] are in place to protect the localization service. Similarly, we assume that the time synchronization mechanism is also secure [28], [16]. Any node in the network can be subject to an attack - such as DoS during hop-by-hop forwarding. We assume that an attacker can alter only the transient information (i.e. contents of the data and message buffers), but it cannot alter the binary representation of any program containing algorithmic implementation. Nodes are not required to be tamper resistant and an attacker that compromises a node can extract data and/or code stored on that node.

C. Attack Classification

In the sequel, we detail the suite of attacks that can be mounted against individual components of the MP-FPR protocol. We classify the attacks as *data-level* and *control-level* attacks based on their target, the user-data or the network operation, respectively. For example, some attacks against *query dissemination, charge allocation* and *data-forwarding* qualify as data-level since they primarily focus on preventing the execution of a user's query or the delivery of the associated data-stream to the user. Attacks against *route-establishment* qualify as control-level attacks since their primarily focus is disrupting the effectiveness of the MP-FPR's energy management and workload balancing. Note that there are certain attacks against *query dissemination and charge allocation* that

can also result in energy disruption and thus we classify those as control-level.

An attacker can drop, delay, or modify any of the five type of messages the MP-FPR protocol relies on: QUERY, UPDATE, RREQ, ACK, and DATA. In this work we do not consider replay-attacks because they can be easily detected and defended against, for example, by means of using packet sequencing or timestamps. In addition, MP-FPR already implements a set-oriented logic, thus replay attacks do not have a functional impact and only contribute to resource wastage, such as energy, computational and bandwidth.

Table II summarizes the main control- and data-level attacks that MP-FPR protocol is susceptible to. Table III provides a more detailed classification, outlining the representative metrics that can be used to assess the impact-level of an attack. We also note that different adversarial mechanisms can lead to the same net outcome(s).

D. Query Dissemination and Charge Allocation Attacks

Attacks during the query dissemination and charge allocation protocol phase can be mounted by targeting the QUERY and UPDATE messages (see Table I). We systematically analyze the modification of the fields of interest in these messages, namely L_{src}, C_e, and N_r, as well as the alternative of dropping or delaying these messages. We identify the following attacks: *data DoS*, *data stream invalidation*, *path diversity deflation*, *path deflection*, and *family path intersection*.

Data DoS. This attack aims at disrupting the delivery of users' data-flow. Although this attack can be easily mounted by maliciously dropping QUERY messages, the absence of the entire data-stream can be easily detected and thus the underlying attack unveiled.

Data Stream Invalidation. An attacker can alter the parameters of a user-submitted query, such as sample rate, filtering criteria, or the source of the data-stream itself. Specifically, an attacker can alter the L_{src} parameter in the body of the QUERY message. As a result, the user will receive an invalid data-stream. It is important to note the stealth property of this attack: as opposed to the *data DoS* attack, the user does receive an uninterrupted data-stream; however, the user may not be aware that it is not the data that he requested.

Path Diversity Deflation. This attack targets the path diversity property which MP-FPR tries to promote. A reduction in the number of alternate paths that the protocol can utilize will affect the load-balancing performance. In MP-FPR , the number of paths that the protocol will try to establish is bounded by a parameter N_r, included in the QUERY message. This parameter is used to control the time-length of the path establishment phase, thus reducing energy wastage by preventing from building too many[2] routes. This duration-control can be defeated if an attacker increases N_r substantially. However, decreasing N_r has the most damaging potential as it reduces path diversity. For example, if N_r is maliciously set to 1, MP-FPR will effectively degrade to *single*-path routing, although not necessarily a *shortest*-geo-path routing such as SGP. *Path diversity deflation* can also be considered a stealth attack, as it may not have an immediate, noticeable impact to the user, however, its damaging effect can be observed over a longer-term period, through a significant reduction of network's lifetime.

Path Deflection. The outcome of this attack consists of a geographical shift of the existing families of routes, or a constraining of the field-region in which routes can be built. This attack can be conducted by modifying charge related information in either the QUERY or UPDATE messages. For example, an attacker can modify the magnitude of a particular charge, or introduce new "fake" charges in the system. In normal usage, the ratio of charge magnitudes correlates with the area of the field-regions within each of which a family of routes can be built. Therefore, maliciously altering the magnitude of a charge will affect the load-balancing among distinct families of routes. In extreme cases, it is possible to narrow the admissible relay field so much that most of the paths within merge, leading to a single-path routing behavior, which is the equivalent of a *path diversity deflation* attack. Adding one fake charge may result in a geographical shift of the existing families of routes, possible leading to increased routes' lengths, with a consequent increase of the end-to-end delivery latencies. Figure 4(b) presents an example of a family-path geographical shift as a result of one fake charge. Adding multiple fake charges is, in fact, the most dominant risk as it can have wide impact over all the families of routes in the network. Adding a significant number of fake charges can ultimately lead to complete isolation of the source nodes from the sink, since the deflection of the connecting field lines may lead to unacceptably long routes.

Inserting fake charges in one of the possible multiple sources can result to charge information inconsistencies, with overlapping field-regions.

Family Path Intersection. This attack targets the fundamental property of electrostatic field based routing formalism: the disjointness of the electrostatic field lines. This property concerns paths pertaining to the same family, as well as paths pertaining to distinct families. Inserting fake charges in only one of the possible multiple sources can result to charge information inconsistencies, with overlapping field-regions as a result and thus overlapping field lines. Maintaining charge consistency is a MP-FPR requirement in order for the electrostatic field line disjointness property to hold among field lines originating at different sources. Such an attack can be mounted by either dropping UPDATE messages or by modifying the L_{src} parameter in the UPDATE message. Some of the conditions that lead to a path deflection may also create intersection between routes pertaining to different families if charge information becomes inconsistent among families. It is important to note, however, that paths pertaining to the same family of routes will continue to maintain the non-intersection property among themselves, however, distinct families of routes will cross each-others geographical bounds. Such path

[2]In [73], it has been observed that there exists a saturation point in terms of the number of multi-paths, beyond which no lifetime gains can be achieved.

TABLE II
ATTACK TAXONOMY

Category	Attack	Drop	Delay	Modify
Control Level	Path Deflection	-	-	QUERY(C_e), UPDATE(C_e)
	Path Diversity Deflation	RREQ, ACK	RREQ	QUERY (N_r), ACK(r_i, L_{src}), RREQ(r_i, L_{src}, t_{sent})
	Family Path Intersection	UPDATE	UPDATE	QUERY(C_e), UPDATE(L_{src}, C_e)
	Wild Path	-	-	RREQ(C_e)
	Field-Line Hopping	-	-	RREQ(r_i), DATA(r_i)
Data Level	Data DoS	QUERY, DATA, ACK	DATA	DATA(r_i), ACK(r_i)
	Data Pollution	-	-	DATA(payload)
	Data Stream Invalidation	-	-	QUERY(L_{src})

TABLE III
OVERVIEW OF ADVERSARIAL SCENARIOS' MAIN TARGETS

Attack	Method	Primary Target	Impact Metric
Path Deflection	Alter charge information in QUERY or UPDATE messages	Workload balancing	Standard deviation of energy reserves
Path Diversity Deflation	Selective forwarding of RREQ or ACK messages	– " –	– " –
Path Diversity Deflation	Delay RREQ messages	– " –	– " –
Path Diversity Deflation	Alter path diversity quota in QUERY messages	– " –	– " –
Path Diversity Deflation	Alter route index in RREQ or ACK messages	– " –	– " –
Path Diversity Deflation	Alter source location information in RREQ or ACK messages	– " –	– " –
Family Path Intersection	Selective forwarding or delaying of UPDATE messages	– " –	– " –
Family Path Intersection	Alter source location information in the UPDATE messages	– " –	– " –
Wild Path	Alter charge information within RREQ messages	– " –	– " –
Field-Line Hopping	Alter route index information in RREQ or DATA messages	– " –	– " –
Data DoS	Selective forwarding of QUERY, DATA	Data delivery reliability	Fraction of non-delivered data-packets
Data DoS	Dropping all ACK messages	– " –	– " –
Data DoS	Selectively delaying DATA messages	– " –	– " –
Data DoS	Alter route index information in DATA messages	– " –	– " –
Data DoS	Alter route index information in ACK messages	– " –	– " –
Data Pollution	Alter data-payload information	Data integrity	Fraction of compromised data-packets
Data Stream Invalidation	Alter source location information in QUERY messages	Data validity	Fraction of compromised data-streams

intersections create resource utilization hot-spots with direct consequences on the overall network's lifetime. Figure 4(c) illustrates an instance of the family path intersection attack.

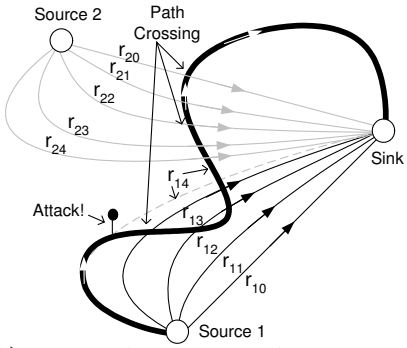
E. Attacks during Route Establishment

Route establishment consists of two phases: *route request* and *route acknowledgment*. The attacks that can be carried during either phase are qualified as *control-level*, as they target RREQ and ACK control messages respectively. An attacker can drop, delay, or modify any of the fields of interest of these messages. Replaying RREQ and ACK messages will mostly result in resource wastage, with no functional impact. We systematically analyze these strategies and identify, in addition to the previously described *data DoS* and *path diversity deflation* attacks, several other attacks.

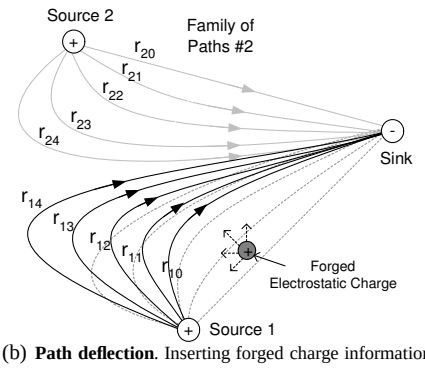
Path Diversity Deflation. In this instance, dropping either RREQ or ACK messages may result in an overall reduction of the route content within a family of routes. Since paths are designed to spread through a larger network-area for workload balancing purposes, an attacker can target an arbitrary node, without a priori insider information. Additionally, delaying RREQ messages or altering the embedded source-transmission timestamp t_{sent} may adversely disqualify a route, from a qualitative perspective, since the route delivery latency as measured at the sink may be increased beyond a user-defined tolerance. Changing the source location information L_{src} in the RREQ or ACK messages will cause ACK messages to be delivered to an arbitrary node, different than the source node. Lastly, altering route index information r_i in the RREQ or ACK messages can also lead to the same outcome. For example, in either case, the (corresponding) acknowledgment will acknowledge an arbitrary route, which may have been already acknowledged, while the intended route will be dropped from usage. Figure 4(e) presents an example of a *path diversity deflation* attack where route r_0's acknowledgment is never received by the originating source node. All of these conditions can ultimately lead to diminished energy consumption balancing performance.

Data DoS. This attack can be mounted by targeting ACK messages. During the *route acknowledgment* phase, compromising ACK messages vs. RREQ messages can lead to different effects, because distinct forwarding mechanisms handle the two types of messages: ACKs are sent via a single-path (SGP), whereas RREQ via EF. Therefore, if a single node along the SGP path is compromised, *all* ACK messages can be compromised or dropped. Since path diversity can be effectively reduced to zero, the user's data-stream will be completely blocked. Alternatively, a malicious node may alter the route index r_i information of the route in the ACK message. In this case, an arbitrary route will receive an acknowledgment, possibly one that was not probed or one that may not satisfy user deliverability requirements, such as end-to-end delivery latency. When it comes to the latter case, the sink checks for latency performance and discards those messages that fail to be delivered within timing requirements. Such data-dropping behavior effectively creates a user *data DoS* attack condition.

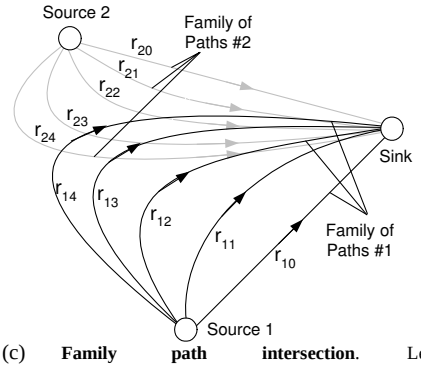
Wild Path. The wild path attack leads to a condition in which one particular route from within a family of routes breaks the disjointness property of electrostatic field based routing and starts intersecting other routes. There are two im-



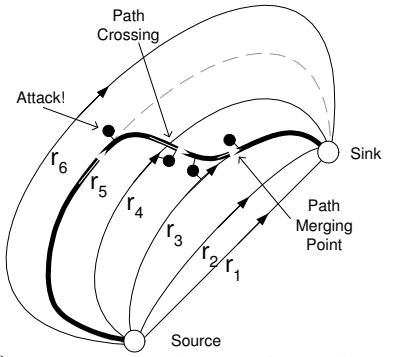
(a) **Wild path**. If electric charges information is altered within a RREQ message at one relay node along a route, the affected route can deviate severely from its prescribed field line and begin intersecting other routes, both within the same family of routes as well as ones pertaining to other families; for example, path r_{14} connecting source 1 to the sink, deviates from its original route as a result of an attack and intersects with other paths.



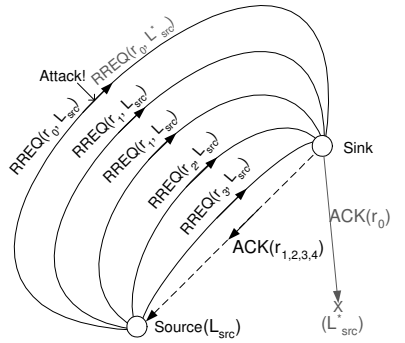
(b) **Path deflection**. Inserting forged charge information in the network, e.g. via compromising UPDATE messages, can lead to geographical shifts of existing families of routes, increasing the overall route-length of all routes within; for example, consider adding one fake charge to the charge-set information of source node 1: routes r_{10} through r_{14} will be detoured around the area where the added charge resides, due to the repulsive effect of the charge



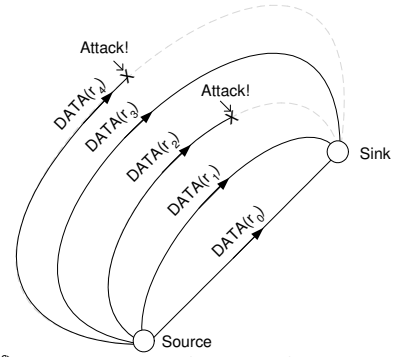
(c) **Family path intersection**. Let $S_{f_1} = \{r_{10}, \ldots, r_{14}\}$ and $S_{f_2} = \{r_{20}, \ldots, r_{24}\}$ be two families of routes, corresponding to two distinct source nodes. If network-wide charge information inconsistencies occur among the two distinct source nodes, such as dropping one UPDATE message, the disjointness property of the routes can no longer be guaranteed, and families of routes will start intersecting with each-other, even though paths within the same family of routes continue to be disjoint; for example, if source node 1 is unaware of the electrostatic charge associated with source 2, S_{f_1} routes will intersect S_{f_2}.



(d) **Field line hopping**. If path-identification information r_i within RREQ or DATA messages is maliciously altered, some of these paths may begin violating the disjointness property and merge with other paths within the same family; for example, if the identification information r_5 is maliciously changed to r_3, the original path r_5 deviates, intersecting r_4 before it merges with r_3, doubling the load downstream from the merging point on r_3.



(e) **Path diversity deflation**. ACK messages are forwarded via SGP towards the corresponding source node using its actual location. Altering this information L_{src} to L^*_{src} through an RREQ message will prevent the ACK message from being routed back to the correct source at L_{src}. For example, path r_0 will never be used for data-forwarding since its never acknowledged: its corresponding ACK message never reaches the source located at L_{src}.



(f) **Data DoS**. Selective forwarding of DATA messages sent along routes r_2 and r_4 nullifies those routes; user will receive an incomplete data-stream at the sink node.

Fig. 4. Examples of attacks against the MP-FPR protocol

portant differences from the *family path intersection* attack: (1) a wild path attack targets a single route, rather than an entire family of routes, and (2) the compromised route intersects not only other routes within the same family, but also routes pertaining to other families. The wild path attack is carried via altering charge information within a relay node along a particular route. Recall that charge information transmitted via RREQ messages are cached by the relay nodes for subsequent use. Consequently, the attack can be carried by altering the RREQ messages before their contents are cached. The entire path downstream of the compromised node will exhibit an abrupt deviation from the designated field line. Figure 4(a) illustrates an instance of a *wild path* attack.

Field Line Hopping. Consider a route indexed by r_j, which is built along a reference field line φ_j. If the route index information embedded in the RREQ message is altered, the original route will suddenly change its reference field line, i.e. will "hop" to a different one within the same family. The immediate consequence is path intersection or merging. This situation is different from a *wild path* situation, because field lines do not change; rather the actual route changes field lines. Figure 4(d) shows an example of *field line hopping* attack. Field line hopping creates relay node overload, resulting in degraded energy consumption balancing and reduction of lifetime expectancy.

F. Data Forwarding

DATA messages carry the information-load resulting from processing a user-submitted query. Since DATA messages follow probed and acknowledged paths, they are virtually susceptible to the same likelihood and means of attack as carried against RREQ messages.

Data DoS. This attack blocks (parts of) a user data-stream. It can be mounted by selectively dropping DATA messages along a particular path, i.e. if one of the relay nodes along the path is compromised. Figure 4(f) illustrates this scenario, in which two different compromised nodes along different routes drop all incoming DATA messages, effectively nullifying those paths. In some instances, altering the route index information r_i in the DATA messages, which can redirect the message along non-probed and possibly long paths, or simply delaying these messages, may similarly lead to a *data DoS* attack. In both cases, it is likely that the message will be discarded at the sink node if not received within certain admissible delay tolerances.

Field Line Hopping. Analogous to the attacks carried through RREQ messages, DATA messages can be maliciously "re-routed" along different routes than the originally prescribed ones, resulting in path merging and overloading of some of the downstream relay nodes. The net effect consists of energy consumption balancing disruption and a reduction of network's lifetime. This attack can be achieved by modifying the route index r_i embedded in the DATA message.

Data Pollution. Lastly, the attacker may directly alter the user-payload within the DATA message itself. This attack can be severe, since the user may not be able to distinguish valid data from faux, and it may require advanced data analysis to detect anomalies in the data-stream.

IV. Defense Overview

In this section we present the basic assumptions and expectations with respect to the cost-effectiveness of the proposed defense mechanism and the feasibility domain in terms of sensor network platforms considered. Subsequently, we identify the areas in which secure solutions are readily available, as well as the areas in which complementary solutions need to be devised. In this sense, we outline the main authentication and integrity mechanisms considered for analytical comparison, as well as the set of resilience mechanisms proposed against selective forwarding of MP-FPR protocol messages.

A. Assumptions

All the proposed solutions need to have a reasonable cost that will: (1) not outweigh the benefits provided and (2) not limit their applicability with respect to realistic platform limitations.

We assume that the only trusted entities in the sensor network are the source and sink nodes. We refer to the source and sink nodes as the *trusted end-points* with respect to a given route. The relay nodes, which represent the vast majority of sensor nodes in the network, have a high-risk of being compromised and are consequently not trusted.

We design defense mechanisms that will not reduce the scope and applicability of MP-FPR protocol, i.e. it needs to fully comply with MP-FPR's system settings: (1) very large sensor networks typically consisting of thousands of nodes, and (2) possibly non-uniform network distributions of various densities. Also, the solutions need to account for the resource limitations of real motes, such as memory and processing capabilities. We evaluate the candidate solutions against several popular mote platforms: Mica2Dot [5], MicaZ [6], TelosB [7], Tmote Sky [8] and Imote2 [4]. A summary of the relevant specifications of these sensors is outlined in Table IV. We note that, with the exception of the small-sized *Mica2Dot*, which is representative for large-scale distributions, the selection is consistent with the one made in [50], where an actual implementation of a cryptographic solution on various platforms is tested.

The SGP and EF message forwarding mechanisms in MP-FPR require a separate, lightweight, and trusted localization service. In this work, we assume that a localization service which meets this criteria is readily available, as existing works have thoroughly addressed this problem [34], [78], [12]. A secure time synchronization service is required to loosely maintain time consistency across the entire network. MP-FPR relies on temporal dimension in order to estimate the quality of paths by time-stamping certain protocol messages, for example, RREQ messages. For this, we rely on solutions such as [16], [28] to provide security guarantees over the time synchronization services. We assume that the localization and time synchronization services are robust to abuses towards resource depletion via link and physical layer jamming [33].

B. Overview the Defense Mechanisms in MP-FPR

As seen in Table II there are two main fundamental causes of the identified attacks: (1) the lack of message *authentication* and *integrity* mechanisms, and (2) the lack of a robust delivery mechanism resilient to malicious message dropping. Specifically, attacks that rely on modification of control or data messages can be prevented by enabling detection of such modifications with the help of message authentication and integrity cryptographic mechanisms. Attacks that manifest through selective forwarding or delaying of messages can be prevented by providing redundancy in the forwarding mechanism, which reduces the likelihood of dropping all copies of a given message.

In Section V we present three message authentication and integrity mechanisms, namely PIKE, DS/ECC, and TESLA, and assess the trade-off between security properties and costs in Section VI. They are primarily considered to address the attacks carried out via message-forging as outlined in Table II, by enabling nodes to detect and filter out modified messages. Additionally, they enable detection of adversarial activity for which isolation mechanisms can be employed. Specifically, path deflection, diversity deflation, family path intersection and wild-path carried through forging electrostatic charges

TABLE IV
OVERVIEW OF RELEVANT MOTE'S SPECIFICATIONS

Platform	Voltage	Current Drawn CPU Active	Current Drawn TX	Current Drawn RX	Data Rate	Program Memory FLASH	RAM SRAM	ROM EEPROM
	[V]	[mA]	[mA]	[mA]	[kbps]	[KB]	[KB]	[KB]
Mica2Dot	3.0	8	27.0	10.0	38.4	128	512	4
MicaZ	3.0	8	17.4	19.7	250.0	128	512	4
TelosB	3.0	1.8	27.0	23.0	250.0	48	10	1024
Tmote Sky	3.0	1.8	19.5	21.8	250.0	48	10	1024
Imote2	4.5	31	44.0	44.0	250.0	256	32,000	32,000

in QUERY, UPDATE or RREQ messages can be prevented. Moreover, field-line hopping, data DoS, data pollution, and data stream invalidation can be prevented as well by authentication and integrity mechanisms.

In Section VII we present three resilience mechanisms to improve robustness of MP-FPR to attacks carried through selective forwarding of MP-FPR protocol messages, namely k-EF, k-RPEF and PDMS. The k-EF represents a multi-path resilient variant of the EF forwarding mechanism, designed to provide defensive against data DoS attacks. The k-RPEF (reverse-path k-EF) aims at replacing the SGP mechanisms with the EF for handling QUERY, UPDATE and ACK messages, in order to provide adequate protection against path diversity deflation, family path intersection and certain data DoS attacks. The path diversity monitoring scheme (PDMS) is a reactive defense mechanism against path diversity deflation attacks. PDMS is designed to complement the k-RPEF defense solution by providing a close-loop control mechanism for ensuring adequate path diversity in an adversarial context. We note that the k-EF and k-RPEF robustness mechanisms can also provide the same benefits to attacks carried through delaying of MP-FPR messages, as they increase the likelihood that at least one instance of a given message reaches the destination on time.

Table V presents a consolidated view of the representative attacks along with the corresponding defense mechanisms and Table VI summarizes the feasibility of all of the candidate solutions considering realistic resource limitations, which will be detailed shortly.

V. AUTHENTICATION AND INTEGRITY MECHANISMS

In this section we overview *three* cryptographical approaches that can provide authentication and integrity verification to MP-FPR protocol's messages. We considere instances of both *symmetric* key cryptography, namely *HMAC*[42] and *public* key cryptography, namely *digital signatures* [40], [35], as well as a hybrid cryptographic solution: *TESLA* [60].

A. Background of Cryptographic Solutions

Symmetric Key Cryptography. Historically, symmetric-based authentication and integrity mechanisms were the preferred method for WSNs application due to the prohibitive cost of the public-key cryptography alternative. Even more, due to the resource constraints of WSNs, key pre-distribution schemes dominate the solution space with respect to symmetric key cryptography, where pair-wise keys required for secure communication link establishment are loaded in each sensor before deployment. There are several theoretical approaches to key pre-distribution schemes: (1) single-mission key pre-distribution, (2) fully pair-wise key pre-distribution, (3) random/probabilistic key pre-distribution schemes, (4) centralized key distribution center schemes (KDC), and (5) decentralized key distribution center schemes (dKDC).

Single-mission and fully pair-wise pre-distribution schemes both have limitations that make them inadequate solutions [26]. Namely, the single-mission keys scheme incurs the least overhead but provides very poor resilience to attacks, as the security is solely based on the secrecy of a single key distributed network-wide. Alternatively, full pair-wise key pre-distribution promises the best achievable security, but introduces a scalability concern, as without prior knowledge of network's topology, the memory overhead becomes $O(n)$, where n is the number of nodes in the network.

Probabilistic key pre-distribution schemes address the full pair-wise scalability concerns while achieving comparable security benefits. The scheme [26] relies on probabilistic key sharing among nodes to establish an initial (connected) topology upon which localized-key sharing would be achieved, at run-time, when needed. The memory overhead is effectively reduced to $O(k)$, $k << n$, where k represents the size of a probabilistically set of keys that would be pre-loaded on each node. The scheme has been improved, most notably in [21], providing increased resilience to attacks. Fundamentally, these approaches rely on a random-graph model, which is connected with very high probability if and only if the average degree of nodes is large [18]. Based on the analytical results of Erdos and Renyi [69], for a typical range of acceptable low-connectivity risk probabilities of 10^{-2} through 10^{-5}, the absolute lower bound on the node degree requirement varies between $13 - 20$ neighbors per node for smaller networks of 1000 nodes, and increases to $15 - 22$ for larger networks of $10,000$ nodes. Consequently, such a scheme will severely limit the applicability of MP-FPR to high-density applications only, offsetting its core benefits in practical lower density networks (i.e. as low as 8 neighbors per node, on average).

KDC-based schemes rely on the presence of a centralized resource-rich key distribution center (KDC) to act as a trusted arbiter for key establishment. Example of such schemes include SPINS [61] and Kerberos [53]. As with all centralized approaches, the distribution center becomes a single point of failure for the security of the entire network. Moreover, centralized approaches do not scale. Therefore, this

TABLE V
SUMMARY OF ATTACKS AND CORRESPONDING DEFENSE MECHANISMS - EFFECTIVENESS PERSPECTIVE

Attack		Defense					
Classification	Mechanism	PIKE	DS/ECC	TESLA	k-EF	k-RPEF	PDMS
Path Deflection	Alter charge information in QUERY or UPDATE messages	√	√	√	-	-	-
Path Diversity Deflation	Selective forwarding of RREQ messages	-	-	-	-	-	√
Path Diversity Deflation	Selective forwarding of ACK messages	-	-	-	-	√	-
Path Diversity Deflation	Delay RREQ messages	-	-	-	-	-	√
Path Diversity Deflation	Alter path diversity Quota in QUERY messages	√	√	√	-	-	-
Path Diversity Deflation	Alter route index in RREQ or ACK messages	√	√	√	-	-	-
Path Diversity Deflation	Alter source location information in RREQ or ACK messages	√	√	√	-	-	-
Family Path Intersection	Selective forwarding or delaying of UPDATE messages	-	-	-	-	√	-
Family Path Intersection	Alter source location information in the UPDATE messages	√	√	√	-	-	-
Wild Path	Alter charge information within RREQ messages	√	√	√	-	-	-
Field-Line Hopping	Alter route index information in RREQ or DATA messages	√	√	√	-	-	-
Data DoS	Selective forwarding of QUERY messages	-	-	-	-	√	-
Data DoS	Selective forwarding of DATA messages	-	-	-	√	-	-
Data DoS	Dropping all ACK messages	-	-	-	-	√	-
Data DoS	Selectively delaying DATA messages	-	-	-	√	-	-
Data DoS	Alter route index information in DATA messages	√	√	√	-	-	-
Data DoS	Alter route index information in ACK messages	√	√	√	-	-	-
Data Pollution	Alter data-payload information	√	√	√	-	-	-
Data Stream Invalidation	Alter source location information in QUERY messages	√	√	√	-	-	-

TABLE VI
SUMMARY OF ATTACKS AND CORRESPONDING DEFENSE MECHANISMS - FEASIBILITY PERSPECTIVE

	Defense					
Property	PIKE	DS/ECC	TESLA	k-EF	k-RPEF	PDMS
No Platform Memory Limitations	-	√	√	√	√	√
Low Communication Overhead	-	√	√	√	√	√
Negligible Processing Overhead	√	-	√	√	√	√
Low Overall Latency Overhead	-	-	√	√	√	√
Low Energy Overhead	-	-	√	√	√	√

class of solutions does not meet the scalability and security requirements on MP-FPR.

Decentralized key distribution schemes (dKDC) like PIKE (Peer Intermediaries for Key Establishment) [18] have a reduced overhead and are fundamentally compliant to MP-FPR's requirements. PIKE relies on a trusted subset of nodes, the "intermediaries", to perform key-management.

Public Key Cryptography. Public-key cryptography does not require secure initial exchange of one or more secret keys between sender and receiver. Energy savings can be achieved by trading off computational overhead for communication overhead reduction. Recent technological advances in sensor networks led to an increase of computational and memory resources, which have reduced the overhead gap associated to public key cryptography. Specifically, the comprehensive experimental analysis performed during the development of TinyECC by Liu and Ning in [50] gave compelling arguments to consider public key cryptography in our analysis as a potential alternative.

One of the most computationally efficient types of public-key cryptography is the Elliptic Curve Cryptography (ECC), which represents the motivational grounds for TinyECC implementation. In addition, ECC features small key sizes and compact signatures, i.e. to provide equivalent security to 1024-bit RSA, an ECC scheme only needs 160-bit key size. ECC is based on the algebraic structure of elliptic curves over finite fields [32].

Hybrid Public/Symmetric Key Cryptography. *Hybrid* solutions to the authentication and integrity problem aim at combining the benefits of symmetric and public-key based schemes: the smaller computational overhead of using symmetric keys and the smaller communication overhead corresponding to public key cryptography. Traditionally, public key cryptography can be used to establish a secure path that in turn exclusively relies on symmetric key cryptography. Hence, the potentially large computational overhead incurred with bootstrapping a path can be amortized if the path is being used for a prolonged period of time.

A well known hybrid scheme is *TESLA* (Time Efficient Stream Loss-tolerant Authentication), which has been proposed by Perrig et. al. in [60]. The main idea in TESLA is to continuously sign streams of data using keyed message authentication codes (MACs). TESLA relies on public key cryptography to securely disseminate the initial signature. More specifically, the scheme is based on the following idea: the sender commits to a random key k, obtained via a pseudo-random function with collision resistance, and transmits it to the receivers. The sender then attaches a keyed message authentication code (MAC) to the next packet P_i and uses the key k as the MAC key. In a later packet P_{i+1}, the sender decommits to k, which allows the receivers to verify the commitment and the keyed MAC of packet P_i. If both verifications are correct, then a receiver knows that the packet P_i is authentic. A chaining of symmetric-keys is thus established, and its security is provided via preceding messages. To bootstrap this scheme, the sender uses a regular public signature scheme to sign the initial commitment, whereas all the other packets are authenticated through chaining.

B. End-to-End Path Security Requirements

In MP-FPR, it is required to secure the end-points' communication routes, in order to enable sensor nodes to discern bogus information from valid data. Additionally, from an energy stand-point, it would also be more efficient to discard bogus information as soon as possible, ideally before it reaches destination nodes, in order to prevent wasteful energy spending of relaying such information over long routes. Thus, we explore the feasibility of performing early detection of unauthenticated information on-route. Under this scheme, *each* relay node will perform authenticity and integrity checks and discard those messages for which the verification does not succeed. We refer to this scheme as the *Hop-by-Hop Authentication* (HHA).

C. HMAC via PIKE

HMAC is a hash-based message authentication code which relies on symmetric keys. Security of a communication link relies on the secrecy of the symmetric key. PIKE implements the key pre-distribution and establishment mechanism that enables the use of HMAC. PIKE is compatible with both low and high density networks as well as non-uniform distributions, which complies with the context under which MP-FPR operates.

The basic idea in PIKE is to devise and pre-distribute a set of $\sqrt{n}$ keys to guarantee connectivity initially to a subset of nodes. These nodes form a basis for further key establishment via *intermediaries*. A key intermediary is a node that shares keys with two other nodes in the network, through which a secure communication path can be established. In fact, each node in the network will act as an intermediary for two other nodes. The key shared between two arbitrary pairs of nodes is unique hence the security is maximized. The (possibly) large body of intermediaries limits the scope of an attack to the few links managed by the compromised intermediaries.

Secure Path Establishment. In order to establish a secure path to another node, the initiating node generates a new path-key and sends it encrypted to one (of possible two) intermediary node with whom both end-nodes share independent keys. The intermediary decrypts and re-encrypts the path-key using the other end-node's shared key, before sending it through. A nonce message is sent back to the initiating node to confirm the establishment of the path. Duplex secure paths can be achieved by the same procedure but in reverse order.

To provide HHA authentication, symmetric path-keys need to be established between an initiating node and each of the relay nodes along a particular route towards the other end of the route. The keys should be distinct, since the security level provided by sharing an otherwise common path-key along the entire route is very weak: capturing one node along a path will compromise the security of the entire path. For sink-to-source secure path establishment, an additional 'SCOUT' message will be sent before MP-FPR's QUERY message. The purpose of the SCOUT message is to trigger individual symmetric key establishment between on-route relay nodes and the initiating sink node. A SCOUT-BACK message will be returned to the sink confirming the completion of the path establishment. The process is similar for the source-to-sink multi-path: it is triggered via S-RREQ messages, which will precede MP-FPR's standard RREQ messages, thus providing authentication of sensitive charge information within RREQ. To allow undistorted path-length estimation, RREQ' packet size can be increased artificially to supersede the size of the DATA packets. Corresponding ACK messages will follow the secure path established via SCOUT.

Bootstrapping. While for each pair of nodes there exists at least one intermediary node to leverage the path-key establishment, its location is not known a priori. To enable quick discovery of such intermediaries without the need of controlled flooding, PIKE relies on a distributed data structure for storing identity and locations of peer intermediaries. Specifically, PIKE employs an address lookup service such as GHT [63] to implement a distributed geographical hash table, where the (*id*, *location*) information of the peer intermediaries are stored. GHT is supported by a subset of nodes to provide storage and lookup. The nodes that support the GHT structure are called *replication* points. GHT establishment takes place only once, during bootstrapping phase, when information about the geo-location of the intermediaries is disseminated to the replication nodes. According to PIKE, each node will send its identity and localization information to its nearest replication node, from where it is forwarded to the "correct" replication node, which in turn is determined by hashing the identity information of the intermediary.

D. Digital Signatures/ECC

The main difference between asymmetric and symmetric key approaches is that keys are generated at run-time, rather than being pre-loaded off-line. Public keys can be generated by each individual node *post*-deployment, during the operational phase, in order to enable digital signature based authentication of protocol messages exchanged in the network. In the followings we iterate the MP-FPR modifications based on public key authentication.

Secure Path Establishment. When two end-nodes intend to establish a secure path, the originating node needs to acquire the public key of the terminus node in order to digitally sign all subsequent outgoing messages. Conceptually, this is a two step process: (1) the originating node announces its intention to establish a secure channel to the terminus node; (2) the terminus replies to the originating node with the public key to be used to perform the encryption. HHA can be easily supported by public key cryptography, requiring the same modifications to the MP-FPR protocol as for HMAC/PIKE. However, instead of triggering path-key establishment between end points and intermediary relay nodes, the SCOUT and S-RREQ messages will contain the public key of the node where the route originates. The public key is stored at the destination and cached by every relay node in between.

Bootstrapping. There is no intrinsic bootstrapping overhead when using ECC-based public key cryptography scheme, with the exception of the initialization and generation times of

individual public-keys for each node. DS/TinyECC does not rely on any other services to operate.

E. TESLA

Lastly, we present the implementation details of the TESLA mechanism for the MP-FPR protocol.

Secure Path Establishment. Without loss of generality, we explain this mechanism from the DATA forwarding perspective. The path establishment process is identical with the one described in PIKE considering the HHA, with one difference: the path's originating node, i.e. the source node, will include in S-RREQ messages an initial key commitment. This step is critical in order to authenticate the entire stream of packets that will be carried and the subsequent keys and commitments within.

Bootstrapping. TESLA relies on TinyECC public-key mechanism for sending the initial commitments, thus the bootstrapping overhead, just as in DS/TinyECC variant, is given by the one-time generation of the public keys during TinyECC initialization step, along with the corresponding memory requirement for TinyECC implementation. TESLA does require that sensors are loosely time-synchronized.

VI. Analytical Comparison

In this section we analyze the overhead of PIKE/HMAC, digital signature/ECC and TESLA by examining four system metrics, (memory, processing, communication and energy) and one user specific metric (latency). For each metric, we distinguish between two phases of a typical sensor network deployment: *bootstrapping* phase – which concerns the immediate post-deployment setup, including node discovery and initial secure topology establishment, and *operational* phase – the remaining period of effective usage. We identify implementation peculiarities and devise analytical expressions of the overhead for each of the candidate solutions. We summarize the analysis in each dimension by providing real-world performance results and comparatively discuss the benefits and drawbacks of each of the candidate solutions.

A. Evaluation Metrics

We quantify the cost of the identified candidate solutions using the following metrics:

- *Memory Overhead* – analyzes the amount of RAM/ROM memory, in kilobytes (KB), that is additionally required, per mote, for storing program code and run-time data structures to provide authentication and integrity to MP-FPR's message system.
- *Communication Overhead* – quantifies the amount of supplemental information, in kilobytes, that is transmitted through wireless medium on behalf of a specific cryptographic solution for a particular task (i.e. route establishment, data forwarding, etc).
- *Processing Overhead* – each cryptographic solution increases a node's processor load and consequently processing times; because these processing times are often non-negligible, in the order of seconds, they are accounted for as well.
- *Latency Overhead* – summarizes the overall equivalent latency introduced, expressed in seconds, due to communication and processing overheads.
- *Energy Overhead* – each task a sensor node performs consumes energy. We express the unit of energy in milli-Joules (mJ). Accordingly, we outline the associated energy overhead of each cryptographic solution as a result of communication and processing tasks.

It is expected that public key cryptography solutions, i.e. Digital Signatures via ECC to yield lower memory and communication overhead at the expense of processing times, as opposed to symmetric key cryptography where lower processing times could be achieved at a cost of higher memory and communication overhead. Energy-wise, communication-costs are generally one order of magnitude greater than processing costs, as Table IV clearly demonstrates to be the case across the platform, reason for which symmetric-key cryptography is also expected to put more demand over the energy resources than public key alternative. The hybrid approach is primarily designed to combine the benefits of both public and symmetric key cryptography without correspondingly combining their drawbacks. In subsequent sections, we present the in-depth performance and overhead analysis to practically understand the extent of these tradeoffs for each solution aside.

B. Memory Overhead

PIKE/HMAC uses $\lceil\sqrt{n}\rceil + 1$ pre-distributed keys. Each *relaying* node needs to store one additional secret key known by itself and the sensor node where the route begins (the *initiator*). Given that MP-FPR aims at achieving disjoint paths, under ideal conditions, a relay node is expected to carry messages from only *one* initiator. Thus, the total expected storage overhead is $\lceil\sqrt{n}\rceil + 2$ keys.

Considering HHA requirements, the source needs to store the shared key with the sink to secure the sink-to-source communication, while the source nodes need to store $N_r R_L$ keys, to secure each of the (N_r) paths, where R_L is the average hop-count of a path. Assuming no restrictions over the location of the source and sink nodes in MP-FPR, the expected shortest-hop distance between any two nodes is guaranteed by PIKE to be $\alpha\sqrt{n}$, where α is a constant dependent on the range of nodes and shape of the deployment area. Considering the hop-count ratio β between the longest admissible alternate path and the shortest path, which models the maximal path-length query-specified restrictions in MP-FPR, the expression of the path-length in MP-FPR is given as $R_L = \alpha\frac{\beta+1}{2}\sqrt{n}$. However, because keys are pre-distributed, some of the nodes will already share keys with the source node and no additional keys need to be shared. The probability that a relay node already shares a key with the source node is $\frac{\sqrt{n}}{n} \cdot \frac{N_r R_L}{n} = \frac{N_r R_L}{\sqrt{n^3}}$ (cf. [18]), where $\frac{\sqrt{n}}{n}$ represents the probability that two arbitrary nodes share a key and $\frac{N_r R_L}{n}$ is the probability that the respective node serves one of the N_r multipaths. Consequently, the effective additional memory overhead is $N_r R_L(1 - \frac{N_r R_L}{\sqrt{n^3}})$.

PIKE has additional storage overhead due to the bootstrapping procedure that requires storage of localization information of intermediary nodes at GHT's replication points. Throughout our analysis, in order to maintain the targeted scalability of $O(\sqrt{n})$ from the perspective of GHT's overhead, and without loss of generality, we have considered the total number of replication points in the network to be $m = \lceil \sqrt{n} \rceil$, where n is the total number of sensors in the network. For this, each GHT's replication node will store an equal share of the network-wide id-location mapping. For example, assuming that κ bits are required for identification and location information, the memory overhead of a replication node is $\kappa \frac{n}{m} = \kappa\sqrt{n}$ bits.

In consequence, assuming K is the bit-size of a symmetric key, the upper bound of *per-node* memory overhead in PIKE/HMAC scheme is dictated by the source nodes and it has the following expressions:

$$M_{PIKE}^{key} \simeq K(\sqrt{n}+1) + KN_rR_L(1 - \frac{N_rR_L}{\sqrt{n^3}}) + \kappa\sqrt{n}$$

Digital Signatures/ECC – The ECC induced per-node memory overhead with MP-FPR protocol is constant (i.e. order $O(1)$), and independent of the number of links that need to be secured. The source's K-bit size public key needs to be cached at each relay node. Sink node incurs, in this case, the largest overhead: given Q_{max} – the maximum number of concomitant queries the network can support, correspondingly the number of source nodes that can exist at any time in the network, the sink needs to store all Q_{max} public keys of all the source nodes. Therefore, from the sink's perspective, the total per-node memory overhead under ECC scheme is given as:

$$M_{ECC}^{key} \simeq KQ_{max}$$

TESLA – TESLA does not incur any bootstrapping overhead. In the operational phase, the security of a path is triggered by sending an initial signed commitment, using public key cryptography, along the prospective path. Without loss of generality, we focus on DATA forwarding mechanism. The security of the path is maintained during data forwarding by piggy-backing signed commitments on DATA message, using symmetric keys, to enable authentication for future messages. To support fast data-rates, TESLA requires a buffer of dR entries to be allocated, where R represents the packet-transmission rate and d represents the disclosure lag d. Each buffer entry consists of (1) signed commitment for a future message, (2) the symmetric key used for authentication of the previous message, (3) keyed MAC codes of the current message and (4) the current messages itself. Assuming that the signed commitment, the symmetric key and the keyed MAC codes are equally sized to K bits and the payload size of the data messages is p, then the memory overhead can be specified as:

$$M_{TESLA}^{key} \simeq dR(3K + p)$$

Practical Comparative Analysis. Table VII presents the RAM/ROM memory overhead, based on real implementations of ECC in TinyECC and respectively HMAC in TinyHash, for various network sizes and densities. We remark that various optimization levels can be configured in TinyECC to trade-off processing vs. memory overhead, and we have considered the cases in which all optimizations are either enabled or disabled. Table VIII cumulates the memory overhead, based on specific memory resources of various sensor motes, and highlights the platforms which cannot accommodate the specific memory demand.

Based on these results, PIKE's memory demand is significantly higher, outweighing both ECC and TESLA by up to two orders of magnitude. Moreover, the memory demand for PIKE makes this solution impractical for the TelosB and Tmote Sky platforms, even when considering smaller networks. Alternatively, both ECC and TESLA provide reasonable memory requirements of below 2KB RAM and 3KB ROM which makes them applicable across all platforms, considering the specifications in Table IV. These results demonstrate ECC's and TESLA's excellent scalability properties, memory-wise. We finally remark that ECC is the most memory-efficient, with an approximatively 50% lower memory footprint when compared to TESLA.

C. Communication Overhead

PIKE/HMAC – It is intractable to compute precisely the communication overhead during the bootstrapping phase for each node that acts as a relay for GHT's localization information, as it may depend on the relative proximity of the replication nodes, i.e. closer nodes will relay more information to the replication nodes than distant ones. Instead, we evaluate an upper bound as it is dictated by the replication points themselves: all GHT establishment traffic flows through them. Namely, a total of $n/m + (n - n/m) = n$ messages will be carried through (receiving and transmitting) in the worst case, where n/m accounts for the receival and dissemination of information from the *local* n/m nodes, i.e. nodes that are closer to a particular replication point than any other node, and $n - n/m$ denotes the amount of location information concerning the remaining nodes, which is re-routed to the proper replication point. Recall that n represents the number of nodes in the network, whereas $m = \lceil \sqrt{n} \rceil$ is the total number of replication points in the network. A hash function serves as an index to determine which replication point contains identity/location information about a particular intermediary. The message overhead is given by the bit size κ of the identity/location information along with any packet-header overhead ϱ. The GHT-overhead is:

$$C_{PIKE}^{GHT} \simeq (\kappa + \varrho)n$$

Path-key establishment consists of a lookup of the intermediary's location followed by a key-exchange between the two peer nodes for which the key is established and their common intermediary. According to PIKE, the communication overhead for a path-key establishment is $\frac{4}{3}\alpha\sqrt{n}$ messages, where α is defined by PIKE as a constant dependent on the range of nodes and shape of deployment area. To provide HHA

TABLE VII
PER-NODE MEMORY OVERHEAD SUMMARY; $\alpha = .5$, $\beta = 2$, $K = 160$ BITS, $\kappa = 48$ BITS, $Q_{max} = 10$

, $p = 36B$, $r = 1pps$, $d = 5s$

Protocol	Net. Size n	No. Replication Points (PIKE-GHT only) $m = \lceil\sqrt{n}\rceil$	No. Routes N_r	Expected Route Length R_L	Bootstrapping Key Predistribution (ROM)	Bootstrapping Initialization Overhead (RAM)	Operational HHA (RAM)	Program (RAM)	Program (ROM)
	[nodes]	[nodes]	[routes]	[hops]	[KB]	[KB]	[KB]	[KB]	[KB]
PIKE	1,000	32	30	24	0.64	0.19	13.74	0.02	0.49
	10,000	100	30	75	1.97	0.59	43.85	0.02	0.49
TinyECC	1,000	-	30	24	0	0	0.20	0.03	1.22
(w/o opt)	10,000	-	30	75	0	0	0.20	0.03	1.22
TinyECC	1,000	-	30	24	0	0	0.20	0.18	1.83
(w/ opt)	10,000	-	30	75	0	0	0.20	0.18	1.83
TESLA	1,000	-	30	24	0	0	0.47	0.20	2.32
	10,000	-	30	75	0	0	0.47	0.20	2.32

TABLE VIII
FEASIBILITY ANALYSIS; $\alpha = .5$, $\beta = 2$, $K = 160$ BITS, $\kappa = 48$ BITS, $Q_{max} = 10$, $p = 36Bytes$, $r = 1pps$, $d = 5s$

Protocol	Net. Size n	No. Replication Points (PIKE-GHT only) $m = \lceil\sqrt{n}\rceil$	No. Routes N_r	Expected Route Length R_L	TOTAL Memory Overhead (RAM)	TOTAL Memory Overhead (ROM)	Feasibility ('-':yes, 'x':no) Mica2Dot	MicaZ	TelosB	Tmote Sky	Imote2
	[nodes]	[nodes]	[routes]	[hops]	[KB]	[KB]					
PIKE	1,000	32	30	24	13.97	1.13	-	-	x	x	-
	10,000	100	30	75	44.47	2.46	-	-	x	x	-
TinyECC	1,000	-	30	24	0.42	1.22	-	-	-	-	-
(w/o opt)	10,000	-	30	75	0.42	1.22		-	-	-	-
TinyECC	1,000	-	30	24	0.57	1.83	-	-	-	-	-
(w/ opt)	10,000	-	30	75	0.57	1.83	-	-	-	-	-
TESLA	1,000	-	30	24	1.14	2.32	-	-	-	-	-
	10,000	-	30	75	1.14	2.32	-	-	-	-	-

we consider securing all (N_r) paths by establishing path keys with each of the $N_r R_L$ relaying nodes within. Hence, the communication overhead of $\frac{4}{3} N_r R_L \alpha \sqrt{n}$ can be significant. In summary, the communication overhead of securing a multi-path is given by the expressions:

$$C_{PIKE}^{multipath} \simeq \frac{4}{3}(K + \varrho) N_r R_L \alpha \sqrt{n}$$

Given a certain route and the symmetric keys shared with all downstream nodes along the route, the HMAC code that is being generated must be embedded in the data-stream messages. In addition to the keys shared with the destination nodes, a number of R_L HMAC codes need to be generated and embedded in the same message to pass authentication checks of *all* relay nodes. Assuming the size of an HMAC code is h, the DATA message size overhead incurred due to forwarding can be expressed as:

$$C_{PIKE}^{data} \simeq h R_L$$

Digital Signatures/ECC – Communication overhead is only incurred during public-key sharing along the sink-to-source and source-to-sink paths. For HHA the public key of the source needs to be sent along each of the N_r paths and cached by each of the $N_r R_L$ nodes within. The communication overhead to establish a family of multi-paths can be expressed as:

$$C_{ECC}^{multipath} \simeq (K + \varrho) N_r R_L$$

The overhead incurred by the DATA messages, given an s-bits digital signature, is:

$$C_{ECC}^{data} \simeq s$$

TESLA – The communication overhead required for initially securing a path is comparable with ECC, since it requires the same public-key mechanism. Additionally for TESLA, however, is the inclusion of a signed commitment of size K in the message that triggers path-establishment (i.e. S-RREQ for data-forwarding). Accordingly, the communication overhead can be expressed as:

$$C_{TESLA}^{multipath} \simeq (2K + \varrho) N_r R_L$$

Data forwarding, on the other hand, relies on symmetric key cryptography. During data-forwarding, each data message, in addition to the user-payload, will incorporate the keyed MAC code of the payload, the symmetric key of the previous message and the commitment for the next message, each of which assumed to have size of K bits. We note that this overhead is independent of number of paths, data rates or disclosure lag. In consequence, the DATA-message overhead is:

$$C_{TESLA}^{data} \simeq 3K$$

Practical Comparative Analysis. To evaluate the relative performance among the cryptographic solutions considered, we have summarized the communication overhead based on the analytical results in Table IX. Since DATA forwarding is expected to dominate the bandwidth usage, it is important to observe that the overhead for providing HHA security level

with ECC or TESLA is small, namely 20 and 60 additional bytes per data-message respectively, when compared to PIKE. From a feasibility standpoint, the range of 480 through 1,500 bytes overhead incurred by using PIKE is prohibitive and impractical, even if packet fragmentation is considered, since the MAC802.15.4's packet size is limited to 120 bytes. Since MP-FPR are designed for large scale sensor networks, long paths are typically the norm, therefore, symmetric key cryptography via PIKE will not scale.

When comparing ECC and TESLA, the latter has a larger message overhead, which is due to the additional inclusion of the commitment of the future message and the actual key for the previous message. As we will shortly see in the following sections, the benefit of the additional 40 bytes per packet overhead far outweighs its cost, however, from a purely communication perspective, ECC seems to be the best solution. Scalability-wise, both ECC and TESLA demonstrate logarithmic performance, as increasing the network size by a factor of 10 increases the associated communication overhead by a factor of 3x for path-establishment.

D. Processing Overhead

We leveraged results of existing works, such as TinyECC and TinyHash implementations, to obtain estimated processing costs associated with all the cryptographic techniques analyzed. Specifically, the analysis concerns the processing times associated to the generation and verification of the digital signatures or HMAC/codes. This analysis does not include the processing times required to perform lower-network stack operations such as routing and medium access control. We also assume the bootstrapping *processing* times negligible when compared to the security-related overhead. The processing timings that we report for DATA-message forwarding are per-route basis, which is necessary to determine correctly the additional delivery-latency incurred along each route. We denote with P_g the key and digital signatures/HMAC code generation time, assumed comparable, and with P_v the validation time of incoming signatures/codes.

PIKE/HMAC – The end-points of a route, i.e. either the sink of the source nodes, are the only nodes in charge with generating keys and HMAC codes in MP-FPR. Accordingly, in order to provide HHA, $N_r R_L$ distinct keys need to be generated to be individually shared with relay nodes across the entire family of routes. The multi-path establishment processing overhead is expressed as:

$$P_{PIKE}^{multipath} \simeq N_r R_L P_g$$

For data forwarding, the processing overhead required to successfully transmit a data-packet across an entire path takes into the consideration the generation *and* validation of the HMAC codes, hence:

$$P_{PIKE}^{data} \simeq (P_g + P_v) R_L$$

where all on-route validation times are factored in, including the destination validation. The source node will need to generate R_L distinct HMAC codes for each packet sent.

Digital Signatures/ECC – ECC distinguishes from PIKE in the sense that generation of a single digital signature is sufficient to provide HHA-based data forwarding along N_r routes, hence the processing overhead incurred by TinyECC is reduced to:

$$P_{ECC}^{multipath} \simeq P_g N_r$$

and, for each of the routes carrying DATA messages,

$$P_{ECC}^{data} \simeq P_g + R_L P_v$$

which accounts for verification overhead at each of the R_L nodes along a path and the key generation at the source.

TESLA – In TESLA, the processing overhead associated with initial path establishment and path maintenance could differ substantially. This is because each of the two phases rely on different cryptographic systems. As we have mentioned, initial path establishment relies on public key cryptography, hence the performance is similar with ECC, accounting for the inclusion of the signed commitment, while path maintenance relies on HMACs. According to the experimental results presented in [60], the computational overhead associated with generation and verification of the commitments is insignificant when compared with the cost of generating an HMAC, a digital signature or performing authentication. Therefore, the processing overhead for securing a family of paths can be approximated as:

$$P_{TESLA}^{multipath} \simeq N_r P_{g_{(ECC)}}$$

where the subscript indicates that the generation times are dictated by ECC execution. When it comes to data-forwarding along a path, the processing overhead is dominated by HMAC generation timing at the source and one verification of the code at the sink and at each $R_L - 1$ relay nodes. Therefore, the data-forwarding processing overhead along an entire path is:

$$P_{TESLA}^{data} \simeq P_{g_{(HMAC)}} + R_L P_{v_{(HMAC)}}$$

Practical Comparative Analysis. We report the processing times for TelosB platform, which is commonly[3] analyzed in both TinyECC and TinyHASH. Based on the results in [44], for example, the execution time for HMAC+SHA1 algorithms, on TelosB motes, is approximatively $P_g \simeq P_v = 105ms$ for both HMAC generation and verification. Table X completes the processing timings for TelosB motes and serves as a comparative reference for the expected overhead differential. These results reaffirm, however, the main drawback of using

[3]We have used the results corresponding to Tmote Sky from TinyECC as representative for TelosB, since both platforms share the same MSP430 processor clocked at the same 8Mhz frequency

TABLE IX
COMMUNICATION OVERHEAD SUMMARY; $\alpha = .5$, $\beta = 2$, $K = 160$ BITS, $\kappa = 48$ BITS, $\varrho = 32$ BITS, $h = 160$ BITS (WITH SHA-1), $s = 160$ BITS

Protocol	Net. Size n	No. Replication Points (PIKE-GHT only) $m = \lceil \sqrt{n} \rceil$	No. Routes N_r	Expected Route Length R_L	Bootstrapping Overhead	Operational	
						Key-Path HHA	Data Forwarding HHA
	[nodes]	[nodes]	[routes]	[hops]	[KB]	[KB/multi-path]	[B/packet]
PIKE	1,000	32	30	24	9.77	360.00	480
	10,000	100	30	75	97.66	3515.63	1500
TinyECC	1,000	-	30	24	0	16.88	20
	10,000	-	30	75	0	52.73	20
TESLA	1,000	-	30	24	0	30.94	60
	10,000	-	30	75	0	96.68	60

exclusively public key cryptography, as in TinyECC: prohibitive processing timings, which can induce very long delays especially in data-forwarding. Clearly, DS/ECC cannot be a feasible solution for HHA-level security since traversing a path can take minutes (on the slower TelosB platform considered, at least), even with all optimizations enabled. For example, traversing a 24-hop route will take approximatively 1 minute. The reason is that, even though large delivery latencies "could" be accepted, it severely limits the delivery rate of the data stream, as the message queues on these motes are relatively small.

By comparison, PIKE/HMAC and TESLA induce far lower data delivery latencies, albeit the path establishment time in PIKE/HMAC in the orders of minutes is prohibitive. TESLA has low forwarding latencies via HMAC mechanism, and low setup latencies via optimized TinyECC. For example, securing a path takes only 2 additional seconds, on par with public key cryptography performance (ECC optimized) and two orders of magnitude faster than PIKE/HMAC, while the induced data delivery latencies are nearly half of the best values of PIKE, conversely, it can support data streams of double data rates. We remark however that we have solely considered the best performances achievable via optimized TinyECC, since the memory overhead required to implement these optimizations are well within the admissible memory bounds of real platforms and likely to be implemented as such.

E. Latency Overhead

All the processing and communication overhead introduce non-negligible latencies during the bootstrapping, multi-path establishment and data forwarding which negatively impact users' experiences.

PIKE/HMAC – The typical duration of the bootstrapping phase is increased due to the GHT service underneath PIKE. The exact latency increase is difficult to compute analytically due to queuing and other MAC-layer protocol specific overheads (i.e.beacons, sleep schedules, etc). Assuming quasi-parallel GHT setup, we can devise a lower bound on GHT's setup time, which is dictated by the communication overhead induced through one replication point, that is, $2C^{GHT}_{PIKE}/R$, accounting for both transmissions and receptions, where R denotes the data-rate of a particular mote platform. In the case of multi-path establishment, the latency overhead is $T^{multipath}_{PIKE} \simeq C^{multipath}_{PIKE}/R$. We note that multi-path establishment latency overhead represents an upper bound and assumes that paths are sequentially built; this is a reasonable assumption if MAC contention is to be avoided, since the communication overhead for path-establishment is significant in PIKE/HMAC. Same analysis extends to the latency incurred for data forwarding along a single route, that is $2C^{data}_{PIKE}/R + P^{data}_{PIKE}$, which include transmission and receiving timings in addition to authentication processing overhead.

Digital Signatures/ECC – ECC/Digital Signatures have no bootstrapping overhead. In the case of secure multi-path establishment and data-forwarding, since there is no significant on-path communication overhead when using public keys, paths may be built in parallel, hence the overhead is reduced to $2C^{multipath}_{ECC}/(N_r R) + P^{multipath}_{ECC}/N_r$, while for data forwarding along a single route is $2C^{data}_{ECC}/R + P^{data}_{ECC}$.

TESLA – Analytically, the performance expressions are similar to DS/ECC, i.e. for securing a family of routes the latency incurred is given by $2C^{multipath}_{TESLA}/(N_r R) + P^{multipath}_{TESLA}/N_r$, while for data forwarding along a single route is $2C^{data}_{TESLA}/R + P^{data}_{TESLA}$.

Practical Comparative Analysis. Table XI illustrates the calculated latency values expected to be exhibited on TelosB platforms, as an comparative example. Correspondingly, the latency due to PIKE's initial GHT establishment ranges between $640ms$ for small network sizes and high-data rate radios of $250kbps$, up to $6.4s$ for large network sizes.

In the case of securing paths, there is a significant trade-off that can be achieved between path-establishment latency and data-delivery latency. For example, using the fully optimized version of TinyECC allows for a quick 2-seconds multi-path establishment, however, the data-delivery latency becomes very large, i.e. up to 4 minutes for nodes comprised of 10,000 nodes, severely limiting the data rate of the user data stream. Alternatively, PIKE's setup time is the order of minutes, however it achieves better data rate margins. If we denote with x the number of multi-paths that can be used simultaneously for data delivery, the maximum data rate achievable is $.2x$ packets per second for smaller-sized networks and $.0625x$ for large networks, with PIKE.

Table XI illustrates that, if using TESLA, one can expect very good performance during both path establishment and data forwarding phases. For example, TESLA doubles the maximum supported throughput when compared to PIKE, while, at the same time, achieves the best path-establishment

TABLE X
PROCESSING TIME OVERHEAD SUMMARY; $\alpha = .5$, $\beta = 2$, $K = 160$ BITS, $\kappa = 48$ BITS, $\varrho = 32$ BITS, TELOSB AND TMOTE SKY MOTES

Protocol	Net. Size n	No. Replication Points (PIKE-GHT only) $m = \lceil\sqrt{n}\rceil$	No. Routes N_r	Expected Route Length R_L	TinyECC Generation Time P_g	TinyECC Validation Time P_v	TinyHASH Generation Time P_g	TinyHASH Validation Time P_v	Operational Secure Path HHA	Operational Data Forwarding HHA
	[nodes]	[nodes]	[routes]	[hops]	[s]	[s]	[s]	[s]	[s]	[s]
PIKE	1,000	32	30	24	-	-	0.11	0.11	75.60	5.04
	10,000	100	30	75	-	-	0.11	0.11	236.25	15.75
TinyECC	1,000	-	30	24	21.00	43.00	-	-	43.00	1,053.00
(w/o opt)	10,000	-	30	75	21.00	43.00	-	-	43.00	3,246.00
TinyECC	1,000	-	30	24	1.58	2.02	-	-	2.02	50.06
(w/ opt)	10,000	-	30	75	1.58	2.02	-	-	2.02	153.08
TESLA	1,000	-	30	24	1.58	2.02	0.11	0.11	2.02	2.63
	10,000	-	30	75	0.58	2.02	0.11	0.11	2.02	7.98

TABLE XI
ASSOCIATED LATENCY OVERHEAD FOR TELOSB PLATFORMS $\alpha = 0.5$, $\beta = 2$,$R = 250kbps$

Protocol	Net. Size n	No. Replication Points (PIKE-GHT only) $m = \lceil\sqrt{n}\rceil$	No. Routes N_r	Expected Route Length R_L	Bootstrapping Overhead	Operational Secure Path HHA	Operational Data Forwarding HHA
	[nodes]	[nodes]	[routes]	[hops]	[s]	[s/multipath]	[s/packet/path]
PIKE	1,000	32	30	24	0.64	76.39	5.78
	10,000	100	30	75	6.40	243.93	22.95
TinyECC	1,000	-	30	24	0	43.04	1,053.03
(w/o opt)	10,000	-	30	75	0	43.12	3,246.10
TinyECC	1,000	-	30	24	0	2.06	50.09
(w/ opt)	10,000	-	30	75	0	2.14	153.18
TESLA	1,000	-	30	24	0	2.09	2.72
	10,000	-	30	75	0	2.23	8.27

timings.

F. Energy Overhead

Similarly to the analysis performed in TinyECC, we assume that the evaluation of the energy consumption can be approximated through an expression $E = U \cdot I \cdot T$, which is based on the duration T of the associated task, the battery voltage (U) and the current drawn (I) specific to the sensor platforms considered, with respect to the motes' specifications outlined in Table IV.

We denote with I_P, I_{Tx} and I_{Rx} the current drawn, in milli-amperes, due to internal processing as well as transmission and receiving of data packets. By duality, let T_P, T_{Tx} and T_{Rx} represent the time-length, in milli-seconds, during which a specific task is being performed. Conversely, we are interested in the energy consumption overhead on per-node basis, in each of the following categories: (1) bootstrapping and key pre-distribution, (2) secure multi-path establishment and (3) data forwarding. The generic expression for energy consumption becomes $E = U \cdot (T_P I_P + T_{Tx} I_{Tx} + T_{Rx} I_{Rx})$.

For analytical evaluation of the energy overhead, we have taken into consideration both the processing and the communication timings results, and grouped the expressions accordingly in Table XII, for PIKE, Table XIII, for ECC and Table XIV for TESLA.

Table XV exemplifies the energy consumption overhead incurred by a sensor node during bootstrapping, secure multi-path establishment and data delivery phases. While the relative overhead is directly correlated with the one outlined in the Section VI-E where latency overhead was analyzed, the energy analysis provides an additional insight: the cost. For example, considering a small $75mAh$ battery that can provide energy for a couple of days under moderate operation, the cost of securing a family of routes under PIKE with HHA security level for networks of 10,000 nodes is approximatively 0.15% of the total battery capacity of the source node. Assuming no limits on the data rates, the cost overhead of relaying a stream of data with a sampling interval of $5s$ for 10 hours is 11% using optimized TinyECC for the source node and 0.33% for a relay node.

By comparison, TESLA achieves 97% energy savings when compared with PIKE for path establishment under 1,000 nodes networks, and 99% under larger networks. However, when compared to DS/ECC, the energy overhead in TESLA is 10% and 25% respectively for path establishment, sensibly more costly. Fortunately, path-establishment is a relatively infrequent operation and its cost can be rapidly amortized during the data-forwarding phase, where TESLA actually achieves energy savings of 92% regardless of the network size when compared to DS/ECC, and between 74% and 90% when compared to PIKE. Accordingly, TESLA proves to be the most cost effective solution by a significant margin.

G. Summary of Authentication and Integrity Defenses

Based on the performance and overhead analysis presented, we conclude that the TESLA approach is feasible across broader real sensor platforms, efficient and computationally tractable. By comparison, PIKE has certain platform limitations due to memory concerns and exhibits higher computational and energy consumption overheads. TinyECC, which is representative for the public key alternative, although a great improvement over PIKE in terms of route establishment

TABLE XII
ANALYTICAL ANALYSIS OF ENERGY CONSUMPTION OVERHEAD FOR PIKE/HMAC (TINYHASH) WITH DATA-RATES R

Phase	Security Level HHA	Units	PIKE/HMAC (TinyHASH)	Dominant Node/Role
Bootstrapping	HHA	[mJ/node]	$U((I_{Tx}+I_{Rx})\cdot(\kappa+\varrho)n/R)$	N/A
Path-Establishment	HHA	[mJ/node/path]	$U\left((I_{Tx}+I_{Rx})\cdot\frac{4}{3}(K+\varrho)N_r\alpha\sqrt{n})/R+I_P P_t\right)$	Source Node
Data-Forwarding	HHA	[mJ/node/packet]	$U((I_{Tx}+I_{Rx})\cdot hR_L/R+I_P P_t)$	Relay Node

TABLE XIII
ANALYTICAL ANALYSIS OF ENERGY CONSUMPTION OVERHEAD FOR ECC/ECCDSA (TINYECC) WITH DATA-RATES R

Phase	Security Level HHA	Units	ECC/ECCDSA	Dominant Node/Role
Bootstrapping	HHA	[mJ/node]	-	N/A
Path-Establishment	HHA	[mJ/node/path]	$U((I_{Tx}+I_{Rx})\cdot(K+\varrho)N_r R_L/R+I_P P_g N_r R_L)$	Source Node
Data-Forwarding	HHA	[mJ/node/packet]	$U((I_{Tx}+I_{Rx})\cdot s/R+I_P P_v))$	Relay Node

TABLE XIV
ANALYTICAL ANALYSIS OF ENERGY CONSUMPTION OVERHEAD FOR TESLA WITH DATA-RATES R

Phase	Security Level HHA	Units	TESLA (TinyECC + TinyHASH)	Dominant Node/Role
Bootstrapping	HHA	[mJ/node]	-	Replication Point
Path-Establishment	HHA	[mJ/node/path]	$U((I_{Tx}+I_{Rx})\cdot(K+\varrho)N_r/R+I_P P_g N_r)$	Source Node
Data-Forwarding	HHA	[mJ/node/packet]	$U((I_{Tx}+I_{Rx})\cdot 3K/R+I_P P_v))$	Relay Node

TABLE XV
PRACTICAL ENERGY OVERHEAD WITH TELOSB NODE ($R=250kbps$)

Protocol	Net. Size n	No. Replication Points (PIKE-GHT only) $m=\lceil\sqrt{n}\rceil$	No. Routes N_r	Expected Route Length R_L	Bootstrapping Overhead	Operational: Secure Path HHA	Operational: Data Forwarding HHA
	[nodes]	[nodes]	[routes]	[hops]	[mJ/node]	[mJ/source/multipath]	[mJ/relay/packet]
PIKE	1,000	32	30	24	48.00	440.09	3.44
	10,000	100	30	75	480.00	1586.79	8.33
TinyECC (w/o opt)	1,000	-	30	24	0.00	233.69	237.02
	10,000	-	30	75	0.00	236.87	233.81
TinyECC (w/ opt)	1,000	-	30	24	0.00	12.40	11.36
	10,000	-	30	75	0.00	15.57	11.12
TESLA	1,000	-	30	24	0.00	13.65	0.88
	10,000	-	30	75	0.00	19.46	0.86

overhead and on the same level with TESLA, it is an order of magnitude less energy efficient than TESLA when it comes to data forwarding. In addition, TinyECC limits the rate of the data-stream since the end-to-end route processing overhead for data-forwarding is non-negligible.

In addition, TESLA is robust against message dropping, in the sense that the security of a path is not compromised and subsequent messages can be successfully decrypted even when certain keys embedded in such messages are lost. For this, TESLA relies on a key-chain mechanism, where a correlated set of keys are generated in a manner in which the initial commitment commits to an entire key-chain. Thus, if a key from the chain is missing, it can be recovered from other keys from the same chain. Also, it is computationally infeasible for the attacker to invert or to find collisions in the pseudo-random functions that are used. Thus, arbitrarily dropping or capturing data packets does not cause any problems in the authentication of subsequent packets. This makes for another strong argument for adopting a TESLA-based solution to provide security to MP-FPR protocol. Our overhead analysis has accounted for the key-chain mechanism as well.

VII. DEFENSE AGAINST ATTACKS VIA SELECTIVE FORWARDING

In this section we provide a background on previously proposed defenses against data dropping, overview our approach, and provide details about our three proposed defenses: k-EF, k-RPEF, and Path Diversity Monitoring Scheme (PDMS). We make the observation that, although the section primarily discusses the defenses against selective forwarding, the solutions identified are also applicable for selective *delaying* of MP-FPR's protocol messages. Specifically, k-EF can provide necessary resilience against delaying DATA messages, k-RPEF addresses the delays of QUERY, ACK and UPDATE messages, while the PDMS provides a resolution mechanism to address the delaying of RREQ messages.

A. Background on Defenses Against Selective Forwarding

There is a variety of works concerning attacks carried through selective forwarding of packets in the research com-

munity, and generically these approaches employ one of the following distinct mechanisms: proactive and reactive.

Proactive mechanisms are employed to provide transparency to the user: during the interval of time between the occurrence of an adversarial behavior and the detection/isolation of it, the user is normally exposed to the effects of DoS attacks. Such mechanisms aim at improving network resilience to attacks carried through message dropping, typically by relaying replicas of the message-streams along multiple paths. For example, *k-redundant depender graphs* [74] relies on graph-topology to provide every node in a graph with k disjoint paths towards the root of the graph. This guarantees delivery even when $k-1$ paths in between have failed, either due to poor link quality of malicious activity. The *k-RIP* [77] represents an improvement by providing probabilistic redundant forwarding to k randomly picked neighboring nodes; the advantage of probabilistic forwarding consists of decreasing the vulnerability to route discovery, such as Sybil attacks. Other methods rely on a deterministic finite path-diversity model to increase robustness by a priori discovering of a family of multi-path routes [66], [36], [41], [15], then using these routes to provide redundancy in the data transmission between two end-points [58]. The MP-FPR approach natively adheres to a deterministic path-diversity model since its core soft-guarantees on packet-delivery performance cannot be maintained under an on-demand path model.

Reactive mechanisms typically employ detection and isolation techniques of misbehaving nodes. One approach consists of abstracting the adversarial activity as a link-quality deteriorations factor and addressing the problem from a robustness perspective. For example, in ODSSBR [14], B. Awerbuch et. al. proposes avoiding the under-performing links by using a modified version of a secure route discovery protocol that incorporates a link-quality metric. Similarly, [71] uses a weight-management scheme to quantify link-quality, but relies on a source-based routing algorithm to generate paths. The net effect of these schemes is avoidance of the compromised areas, allowing for a graceful degradation of service. In contrast, other schemes adopt a radical detection and isolation model: nodes exhibiting unexpected behavior are removed immediately and permanently from the network's topology. Typical approaches consist of: (1) performing end-to-end monitoring and statistical analysis of traffic patterns – the *pathrater* technique [52], and (2) exploiting topological properties in sensor networks, i.e. multiple nodes are within collision domain, which enables overhearing of node's communication in a wireless channel for the purpose of detecting unexpected communication patterns [54], [62], [13], [39].

B. Our Approach

MP-FPR uses five type of messages sent via two forwarding mechanisms, the EF mechanism and the SGP mechanism. Consequences of attacks carried through selective forwarding of the MP-FPR protocol messages are presented in Table II. The most intuitive way to protect against these attacks is to provide a proactive approach for all these messages. However, RREQ messages cannot benefit from such redundancy mechanisms since RREQ messages are bound to the route they probe and implicitly construct, copies of RREQ messages cannot be sent on different routes.

The proactive defense mechanism that we propose uses replication of outgoing messages in order to improve resilience to adversarial activities. The solution aims at providing redundancy in the forwarding mechanism. Instead of one message, a number of k-copies of a certain message may be sent along k-distinct routes, significantly reducing the probability that an attacker will successfully manage to drop all k such copies. We refer to the parameter k as the *degree of replication*. This approach is appealing because the required underlying support, i.e. multi-path routing, is readily available in MP-FPR and thus requires minimal changes.

Both source-to-sink and sink-to-source traffic must be augmented with resilient forwarding mechanisms. The source-to-sink traffic consists of DATA messages, for which resilient forwarding can be easily provided: these messages can be sent along subsets of already constructed routes. We refer to this mechanism as *k-EF*. Note that these subsets of routes are still used in alternation for workload balancing purposes.

Sink-to-source, reverse-traffic, comprises QUERY, UPDATE and ACK messages. The challenge here is that these messages rely on SGP forwarding mechanism and no routes are readily available as in the EF mechanism. There are two possible solutions that can be considered to provide k-resilience to reverse-path selective forwarding in MP-FPR: (1) replacement of the standard SGP mechanism with a k-shortest path routing [29] (which we refer to as k-SGP), and (2) adapt MP-FPR protocol to rely directly on the field-based forwarding provided by EF to forward copies along multiple *on-the-fly* built routes, which we will refer to as k-RPEF (Reverse Path Electrostatic Forwarding). In this work we adopt the secondary approach, i.e. k-RPEF, for the following three reasons: (1) it is relatively easy to implement since it relies on the same forwarding mechanism as in EF, (2) it simplifies the network-protocol stack by removing the SGP component altogether, and (3) its redundant paths inherit the non-braiding property of field-based routing, which cannot be guaranteed with k-SGP.

In the case of RREQ messages, we propose a reactive mechanism, namely the Path Diversity Monitoring Scheme (PDMS). This monitoring scheme reactively attempts to compensate for any deficiencies in path diversity by persisting in building more routes until the user defined path diversity quota is met.

C. k-EF Defense Mechanism

The k-EF mechanism provides replication of DATA messages using the set of active routes resulting from the route establishment phase. The degree of replication is given by the value of $k < N_r$, where N_r represents the maximum number of routes that can be established. We use a random selection scheme to select k paths from the total of N_r possible, we adopt a random selection scheme. We remind

that the N_r routes are uniquely identified via a route index $r_i \in \varphi_{N_r} = \{1\frac{2\pi}{N_r}, 2\frac{2\pi}{N_r} \dots N_r\frac{2\pi}{N_r}\}$, i.e. equally distributed across the $\varphi \in (0 \dots 2\pi]$ domain, hence in a k-redundant scheme, the indexes of the k routes should be randomly picked from the φ_{N_r} set without replacement.

D. k-RPEF Defense Mechanism

k-RPEF provides redundant forwarding of QUERY, UPDATE and ACK messages towards the source nodes. Forwarding will continue to be based on electrostatic field lines, but traversed in opposite direction of the field vectors, towards the source. In order to enable reverse electrostatic field lines traversal, a simple modification is due: reverse the algebraic sign of the charge's magnitudes corresponding to the sink and specific source charges for reverse path forwarding only. For example, if a sink and a source have charges of $Q_{src} = -1 \cdot 10^{-19}$ coulombs and $Q_{snk} = +1 \cdot 10^{-19}$ coulombs respectively, k-RPEF's field lines will be built on the set of charges $Q_{src} = +1 \cdot 10^{-19}$ coulombs and $Q_{snk} = -1 \cdot 10^{-19}$ coulombs instead. We note here that only the source's charge towards which we intend of forwarding the message gets the magnitude reversed, whereas other source nodes remain unchanged – this is required in order to prevent messages from reaching other source nodes by hopping on their field lines. Also, the algebraic magnitude's sign reversal is performed in isolation from other sources, i.e. such information is not broadcasted and it is only used locally. Charge magnitude reversal forces the field line vectors to point towards the source node rather than the sink, guiding the associated routes accordingly, without further modification of the forwarding algorithm.

E. Path Diversity Monitoring Scheme (PDMS)

Dropping of RREQ messages critically affects path diversity and, consequently, the energy balancing. Although the k-RPEF mechanism addresses the path diversity deflation problem from the perspective of attacks against ACK messages, it cannot be used for attacks against RREQ messages, because RREQ messages are uniquely associated to the routes they are forwarded through, hence replicas of a RREQ message cannot follow a different route. The idea in PDMS is to enable the source node to persist in probing for new routes until the user-specified *path diversity quota*, i.e. number of distinct routes N_r the user demands, is being met. PDMS relies on the observation that distinct routes will map to distinct sets of nodes, hence bypassing of compromised nodes can be achieved in subsequent attempts.

Note that PDMS cannot be used as a standalone solution for path diversity deflation attacks carried out via ACK messages, for the following reason. Recall that, in the absence of k-RPEF mechanism, ACK messages are sent via SGP forwarding, therefore compromising the single reverse path will block the acknowledgment phase completely. In this case, regardless of the number of attempted routes to be built, routes will never get acknowledged. PDMS, however, can provide *compensatory* benefits if the k-RPEF resilient mechanism is already employed for ACK messages, and our experimental results will demonstrate this benefit.

One of MP-FPR protocol goals is to evenly distribute the workload by building evenly distributed routes in the physical field. It is therefore desirable that this property is either maintained or gracefully degraded under adversarial conditions. Accordingly, the *sequence* of routes that will be probed must take into consideration the existing distribution of routes and attempt to fill any existing "gaps". Recall that MP-FPR adopts an *angular model* for route-indexing cf. Section II-C. Consequently, we rely on the assumption that the distribution of the routes indexes (i.e. distribution of radii over a disk) is representative for the distribution of the actual routes.

We propose the PDMS mechanism as a multi-phase process. The first construction phase performs the same functions as in the original MP-FPR protocol, namely a *sequence* S_1 of N_r evenly distributed route indexes are generated and iteratively probed, $S_1 = \langle r_i | r_i = \frac{2\pi}{N_r} i, i \in \overline{1, N_r} \rangle$. If the path diversity quota is not met during the first phase, subsequent construction phases are invoked. The followings apply to every phase $j \geq 1$. We refer to S_j as the *base routing sequence of phase* j. Let A_j be the *set* of *active* routes that have been successfully acknowledged up to phase j. If and only if the path diversity quota is not being met at a certain phase j, i.e. $|A_j| < |N_r|$, a subsequent phase $j+1$ is initiated. In each subsequent phase $j > 1$, a new distinct sequence S_j is being generated such that $|S_j| = N_r$ (the generation method will be addressed shortly). As opposed to the very first phase however, not all routes in S_j need to be probed, and the probing process can be interrupted at any time if the path diversity quota is being met. To prevent wasteful energy resources under severe adversarial conditions, we limit the number of phases that can be executed to a predefined value $K \geq 2$.

The base routing sequence at phase $j > 1$ is generated as a counter-clockwise rotation of the base sequence of angular indexed routes from previous phase, i.e. all route indexes from current phase are obtained by incrementing the route indexes of the previous phase by a fixed amount δ. Considering the maximum number of admissible probing phases K, in the worst case scenario, the union of all base routing sequences is $\bigcup_{j=1}^{j=K} S_j = \langle r_i | r_i = \frac{2\pi}{K \cdot N_r} i, i \in \overline{1, N_r} \rangle$, hence a total of $N_p = K \cdot N_r$ distinct and evenly distributed routes may be probed by PDMS. Figure 5 illustrates the base routing sequences for $K = 3$ construction phases and $N_r = 8$ routes per phase for which the calculated rotation is $\delta = 15°$.

In order for the PDMS to ensure even distribution of the resulting routes, the base routing sequence generation mechanism is necessary, but not sufficient. Namely, since subsequent route construction phases can be terminated immediately when path diversity quota is being met, priority must be given to routes situated in the vicinity of a failed route, whose omissions has created a "gap". The intuition is as follows: if originally the base routes led to evenly distributed routes with the exception of one route, it is desirable to build a replacement route as close as possible to the original failing

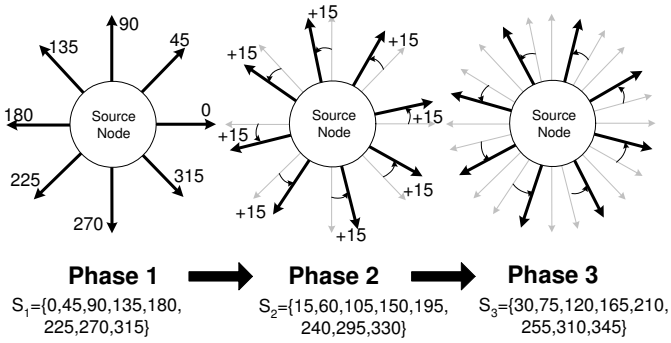


Fig. 5. Base routing sequences for $K = 3$ route construction phases and $N_r = 8$ routes per phase. Each phase's routes' indexes are obtained by applying a rotational shift of routes' indexes in previous phase by $\delta = K \cdot 2\pi/N_r = 15^\circ$

one, such that the deviation from the targeted distribution is minimized. This motivates the phased generation of the base sequence of routes, where δ represents the deviation added to the routes from original location.

The advantage of the proposed PDMS scheme versus a purely random one, in which route indexes are randomly, with uniform distribution, generated, is twofold: (1) PDMS maintains full control of the probed routes by primarily targeting areas with lower densities of routes (i.e. in *immediate* vicinity of failed routes) to improve route distribution, and (2) it avoids route merging effects caused by new routes that may be randomly chosen "too close" to existing ones by guaranteeing a minimum path-spacing through δ. Also, from a users' experience perspective, the current PDMS scheme does not increase the interval of time until the first data-stream path is established. This is also advantageous over another possible path-generating mechanism in which a super set of KN_r routes are generated as base routes in one phase, and subsequently retain a subset of N_r routes that satisfies certain distribution requirements – the latter mechanism is also wasteful, in terms of energy and bandwidth resources, as it requires a large number of routes to be built, even under non-adversarial conditions, the majority of which not being unused.

We present now the prioritization mechanism that is applied to the base routing sequence of phase j, $\overline{S}_j$. The key idea is to determine the angular-gap size between any two adjacent route indexes from the ordered set of active routes A_j, and store these gaps' information in an ordered set G_j in descending order of the gap-size. Given a base routing sequence $\overline{S}_j$, we reorder the sequence such that the i^{th} element in $\overline{S}_j$ is situated within the bounds of the i^{th} gap in G_j. Algorithm 1 details this mechanism.

An example of the priority base route generation in PDMS is presented in Figure 6, containing direct references to Algorithm 1. The key of this algorithm is found in lines $\#10-\#12$ where an ordering of the route indexes is established based on the gap-size a particular route fills. The role of line $\#5$ is to align the routes in S_j from line $\#3$ with the set of active routes

Algorithm 1 Priority Base Route Generation in PDMS

Input:
j: current PDMS phase number ($j = 1$ for 1^{st} construction phase generation)
K: maximum number of phases
A_j: set of active routes at phase j ($A_j = \emptyset$ if $j = 1$)
N_r: targeted number of routes
Output:
S_j - **sequence** of base routes for phase $j+1$

1: $A_j \leftarrow A_{j-1}$
2: $\delta = K \cdot 2\pi/N_r$
3: $S_j = \langle r_i | r_i = \frac{2\pi}{N_r} i + \delta(j-1), i \in \overline{1, N_r} \rangle$
4: $S_j = S_j \cup A_{j-1}$
5: $S_j = Sort(S_j)$ // ascending order sequence
6: $first = 0$
7: $last = Max(0, |S_j| - 1)$;
8: $B = \langle S_j[last], S_j, S_j[first] \rangle$ // Wrap around sequence
9: $T = \langle \emptyset \rangle$ // sequence of $\langle key, value \rangle$ tuples
10: **for** $i = first; i \leq last; i = i + 1$ **do**
11: $gapSize = (|B[i+1] - B[i]| + |B[i+2] - B[i+1]|)\%\pi$
// Insert new key-value entry: $\langle$ gap size, route index $\rangle$
// and order descendingly by key: gap size
12: $T \leftarrow InsertionSort(T, \langle gapSize, S_j[i] \rangle)$
13: **end for**
14: $S_j = \langle \emptyset \rangle$ // Clear for prioritized order
15: **for** $i = first; i \leq last; i = i + 1$ **do**
16: // Build, in order, the prioritized sequence of routes
17: $S_j \leftarrow \langle S_j, T[i].value \rangle$ // append route index at the end
18: **end for**
//Remove active routes from base routes
19: $S_j \leftarrow S_j \setminus A_j$

A_{j-1}, from previous phase, such that proper gap evaluation is achieved.

F. Conclusion of Resilience Mechanisms in MP-FPR

Resilience mechanisms improve robustness of MP-FPR to attacks carried through selective forwarding (or delaying) of MP-FPR's protocol messages. Specifically, path deflation attacks via dropping or delaying of RREQ and ACK messages, family path intersection attacks via UPDATE messages, data DoS attacks via targeting the DATA, QUERY or ACK messages, can all be prevented by adopting redundancy mechanisms. In addition, the robustness mechanism may provide incidental benefits to other types of attacks. For example, path deflation attacks via delaying RREQ messages can also benefit, since paths that do not meet established quality levels are inherently discarded, but can be compensated for by building a replacement path, which is the case with PDMS mechanism.

VIII. Experimental Evaluation

In this section we evaluate the effectiveness of the proposed defense mechanisms and demonstrate their viability. First, we

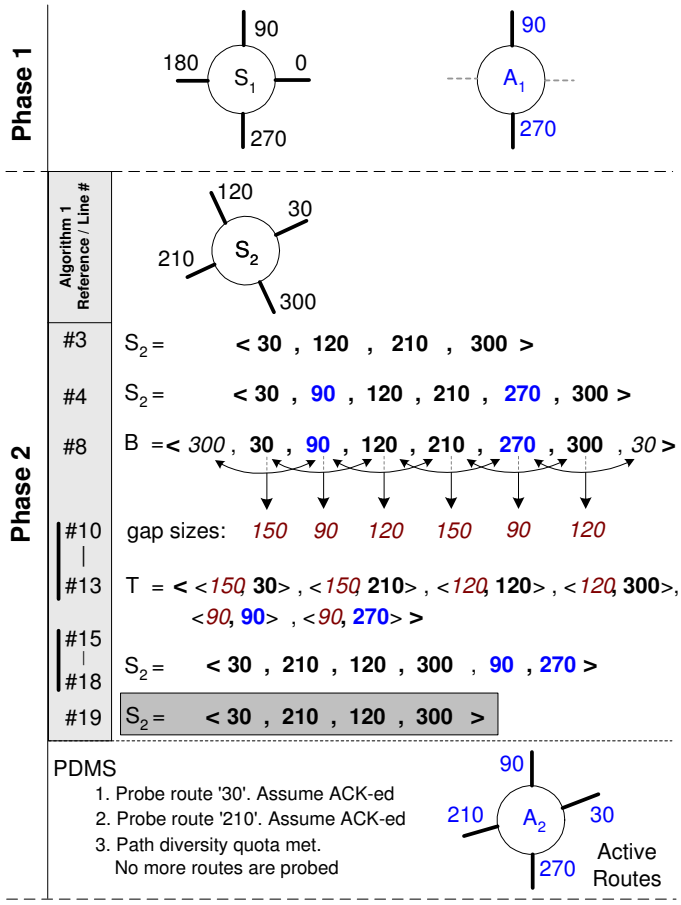


Fig. 6. Priority base route generation example. Settings: phase 1 completed, executing phase $j = 2$, $K = 3$, $N_r = 4$, $\delta = 30°$. For clarity, we express route indexes in degrees, rather than radians. We assume that only two of the candidate routes in phase 1 were ACK-ed and the set of active routes is $A_1 = \{90°, 270°\}$ Algorithm 1 is being applied and a new sequence of route indexes to be probed is generated: $S_2 = \langle 30°, 210°, 120°, 300°\rangle$. Observe that priority is given to routes $30°$ and $210°$ as they are first elements in probing sequence S_2, since these are in vicinity of the unacknowledged routes from phase 1. Subsequently, routes $30°$ and $210°$ are probed iteratively. If both routes are ACK-ed, the set of active routes becomes $A_2 = \{30°, 90°, 210°, 270°\}$ of cardinality 4, which meets the path diversity quota and phase 2 is interrupted. Otherwise, phase 2 continues with probing of routes $120°$ and $300°$.

overview the experimental settings and outline the metrics used in this quantitative analysis. Next we present the experimental overhead analysis of the TESLA integrity mechanisms, which is the solution of choice conform Section VI. Lastly, we detail the experimental findings for the selective-forwarding resilience mechanisms, i.e. k-RPEF, PDMS and k-EF.

A. Simulation Settings

The experiments were performed using the SIDnet-SWANS simulator [30], [1] for WSN. SIDnet-SWANS is an open-source large scale sensor network simulator, which facilitates fast algorithmic implementation on a sensor network comprising a large number of sensor nodes. SIDnet-SWANS is built on the scalable architecture of JiST-SWANS [2], which in turn is based on a high-performance JiST (Java in Simulation Time) engine. When compared to other popular options for sensor network simulation such as ns-2, SIDnet-SWANS enabled us to prepare and perform a large body of experiments in a relatively short amount of time in an environment comprising hundreds of simulated sensor nodes. On the other hand, as far as network stack correctness is concerned, it carries adapted version of ns-2's MAC802.15.4 protocol and same signal propagation models.

Network Configuration. The simulated environment consists of a set of 750 homogeneous nodes having the following configuration: (1) 20 kbps transmission/reception rate, (2) MAC802.15.4 protocol, (3) 5 seconds idle-to-sleep interval (i.e., nodes that are not actively involved in routing enter a low energy consumption state after 5 seconds of continuous idling, in order to preserve battery power), and (4) power consumption characteristics based on Mica2 Motes specifications [3]. To reduce the simulation time while preserving the validity of the observations, nodes were configured to use a small battery with an initial capacity of 35 mAh, for a projected lifespan of several tens of hours under moderate load.

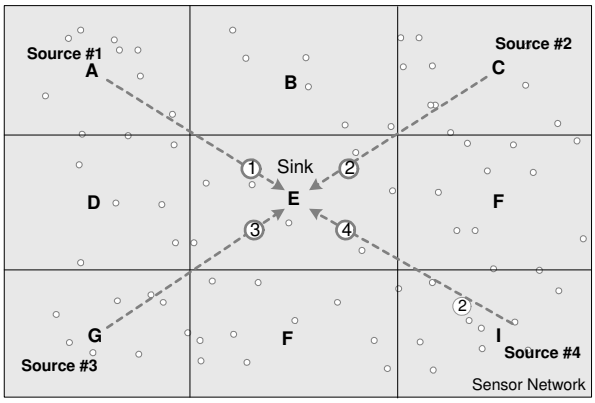
Application Settings. The tested scenario consists of four distinct, long-term, continuous, point-to-point queries rooted at a common sink node. The sink is centrally located within the network. The four corresponding source nodes are evenly distributed around the sink node, namely within the regions A, C, G and respectively I of a grid-based partitioning of the network as shown in Figure 7. This configuration has two advantages: (1) it provides approximatively 90% spatial coverage of the relay area to the network resources (nodes) and (2) it creates a context of four physically adjacent families of routes, which enables investigating of the family path intersection attacks via selective forwarding of UPDATE-messages – which violates the disjointness property of paths pertaining to *different* source-sink families of routes. To further support the latter advantage, the four queries are injected in the network *sequentially*, in the order shown in Figure 7, at 10 minute simulated time intervals. The path diversity quota has been set to $N_r = 30$ routes, and the PDMS's path offset $\delta = 4°$ for a maximum of $N_p = 90$ pool of candidate routes.

Each experiment captures 8 hours of simulated time. Data transmission interval of the point-to-point queries to the designated sink is 4 seconds. As part of the experimental setup, we have gradually increased the set of attacking nodes, which are randomly and uniformly selected from the network, ranging from 5% to 30% of the total sensors in the network.

B. Metrics

Recall that, according to the adversarial model presented in Section III, attacks are classified as *control-level* and *data-level* attacks.

The *control-level* attacks target the main control messages in MP-FPR, namely QUERY, UPDATE, RREQ and ACK. These attacks can either block a user's query from being executed, or disrupt the energy efficiency of the MP-FPR protocol during query processing. For the former, we monitor the *successful query dissemination rate*, expressed as the ratio



Point-to-Point Queries: (1) A -> E; (2) C -> E; (3) G -> E; (4) I -> E;

Fig. 7. Spatial Partitioning of the Network with respect to the experimental point-to-point queries

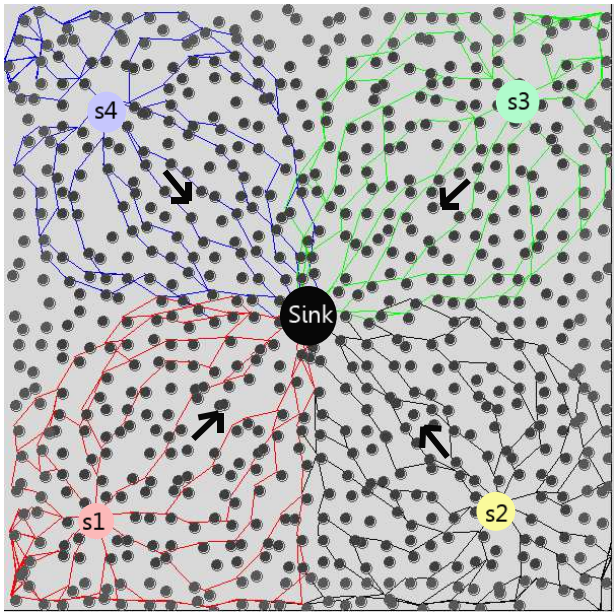


Fig. 8. SIDnet-SWANS snapshot depicting spatial distribution of the experimental point-to-point queries

between the number of queries received at the corresponding source nodes for processing and the total number of queries submitted through the sink node. Considering the energy-efficiency, one important mechanism of attack is the path-diversity deflation attack. To assess path diversity, the effective number of established end-to-end paths are monitored. The other types of control-level attacks: path deflection, family path intersection, wild path and field-line hopping can only be effectively quantified by assessing the disruption of the energy consumption patterns. Correspondingly, we monitor the average residual energy levels $\overline{E}$ in the entire network, normalized relative to the capacity of a fully charged battery E_{max}. The effectiveness of the workload balancing paradigm and its associated energy consumption distribution is measured by means of the standard deviation of the percentage-representation of the residual energy reserves E_σ. Namely, if $E_i(t) \leq E_{max}$ is the residual energy level of a sensor node sn_i at a discrete time t, then the average energy level in a network of N nodes is $\overline{E}(t) = \frac{1}{N}\sum_{i=1}^{N} E_i(t)/E_{max}$. The standard deviation of the energy level is computed as follows:

$$E_\sigma(t) = \sqrt{(E_i(t) - \frac{1}{N}\sum_{j=1}^{N} E_j(t))^2} \quad (1)$$

The *data-level* attacks concern user's perceived experience of the delivery of the DATA-stream from an integrity, reliability and performance perspective. Integrity mechanisms, which concern data pollution or data stream invalidation attacks, provide generalized protection to all DATA messages, hence we do not fractionally track this criteria. From a deliverability reliability standpoint – a primarily focus for data DoS attacks – we rely on the packet-delivery ratio $\eta = n_{rcv}/n_{exp}$, established between the number of packets actually received n_{rcv} by the sink node and the total number of packets sent n_{exp} by the source node and expected at the sink over an interval of time. In multipath settings, the delivery ratio accounts for the successful transmission of one (of the possible many) copies of a packet. Also, the (depreciation of the) packet delivery latency is also monitored as part of the overhead analysis.

C. Evaluation of TESLA for Integrity and Authentication

For demonstrating the effectiveness of TESLA for integrity and authentication, we mounted a path deflection attack via altering of electrostatic charge information in network via either QUERY or UPDATE messages. Path deflection is the most representative attack to be considered because (1) it is an attack that targets unique characteristics of electrostatic field-persistent routing, (2) it requires very little resources to mount and (3) it can yield most damaging effects over the energy consumption patterns. The analysis focused on TESLA because, conform Section VI, it represents the best choice considering not only applicability domain, but also practical overhead.

Note that TESLA is also an effective solution against path diversity deflation attacks by preventing forgery of path diversity quota in QUERY messages, or the route index information in the ACK/RREQ messages. Also, family-path intersection, wild-path conditions and field line hopping can be fully prevented as well. From a user-experience perspective, TESLA is an effective solution against data polluting or data stream invalidation attacks, and it can also be used to prevent data DoS attacks carried through forging route index information in DATA messages.

Energy balancing and data delivery rate performance evaluation The path deflection attack is constructed as follows: forged-charges are generated and randomly placed in various areas of the network through the UPDATE messages. Various levels of attack efforts are considered, by varying the number of forged charges between 4 and 24, the upper bound value being enough to create major loss of connectivity in the network, as the experiments will show.

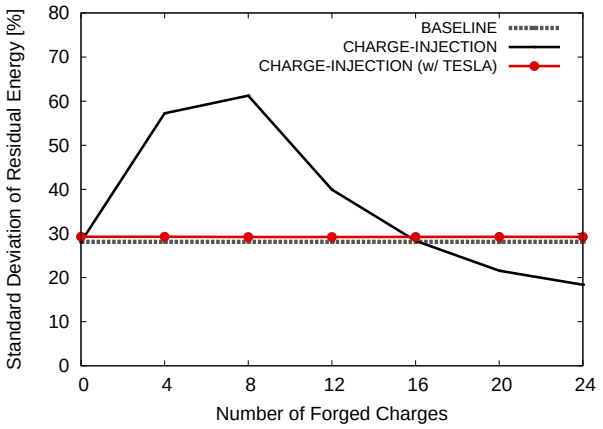


Fig. 9. Impact of path deflection attack via charge forgery to residual energy balance and effectiveness of TESLA defensive mechanism

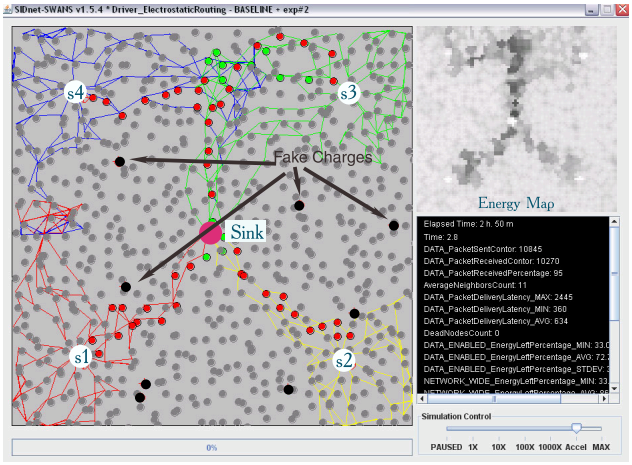


Fig. 10. SIDnet-SWANS snapshot depicting routes' distributions as a result of severe path deflection attacks carried by inserting of eight forged charges. The energy-effect may be observed in the energy-map at the top-right corner of the snapshot, where darker areas represent depleted areas due to perusal of commonly used relay nodes

Figure 9 illustrates the primary impact of inserting invalid charge information in the network: disruption of the energy balancing the MP-FPR is designed to achieve. As it can be seen, MP-FPR is very sensitive to this type of attack: even few number of forged charges, for example 4 such charges, are enough to drastically affect the evenness of the energy consumption, as the standard deviation of residual energy reserves nearly doubles, according to Figure 9. The reason behind is the severe path deflection and agglomeration of routes in narrow physical areas, as a result of the repulsive effect of multiple forged charges[4]. In these conditions, most, if not all, of the alternate paths within a family merge and converge towards a single path type of routing in the relay area. MP-FPR effectivelly degrades towards a single-path routing behavior. For example, Figure 10 snapshots a SIDnet-SWANS run-time instance showing the effective distribution of routes resulting from a charge forgery attack with 8 large magnitude charges.

When a larger body of forged charges are considered, i.e. more than 8 such charges, there exists an apparent improvement of the energy-balance, as it can be observed in Figure 9. This observation surfaces, in fact, an extreme side effect of charge forgery attack: user perceived data DoS. Namely, it is possible that field lines are deflected enough that *all* of the associated routes are too long to be accepted in the route construction phase. The net result is a complete isolation between affected source nodes and their targeted sink. This lack of connecting routes prevents the data-stream from being sent to the sink, resulting in energy-savings by not performing the required workload. To demonstrate that this is the case, we capture the impact over the data-delivery rate in Figure 11. As it can be observed, data-delivery rate drops because of this effect. Correspondingly, network wide average of residual energy levels improves by up to 12%, conform Figure 12, when 24 forged charges are randomly injected in the network.

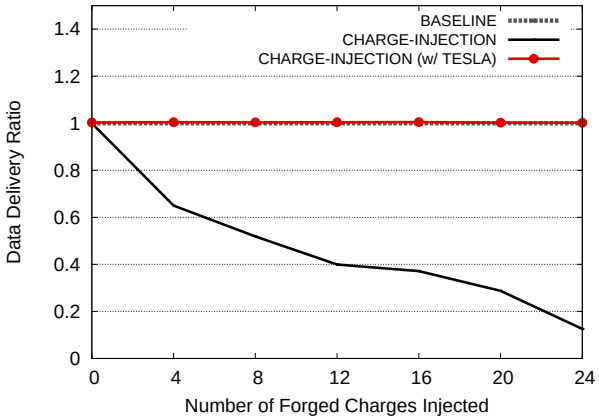


Fig. 11. Impact of path deflection attacks over data stream delivery ratio

TESLA energy and latency overhead evaluation. Figures 9, 11 and 12 demonstrate that TESLA not only provides the required protection against all path-related attacks, namely path deflection, path diversity deflation, family path intersection, wild-paths and field-line hopping, but the energy-overhead is minimal and independent of the dimension of the attack. Namely, it can be observed that TESLA's impact over the energy-balancing mechanism is below 3%, whereas, conform Figure 12, the impact over the network-wide average residual energy levels is maintained below 5%.

The overall data-stream delivery latency is increased when TESLA is being used. However, this is due to the key-generation process that takes place at the source node prior to message transmission, as well as due to on-route key-verification process. Figure 13 demonstrates that the TESLA mechanism increases the end-to-end data delivery latency by a factor of three – a 1-1.5 seconds latency increase over the

[4]In this work we assume the worst case scenario in which forged and real charges have the same polarity, leading to a finer partitioning of the physical space among all resulting field lines, real and forged

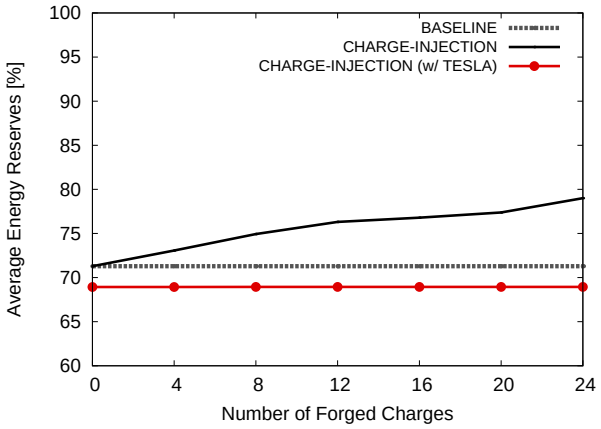


Fig. 12. Average residual energy levels with and without protection against path deflection attacks

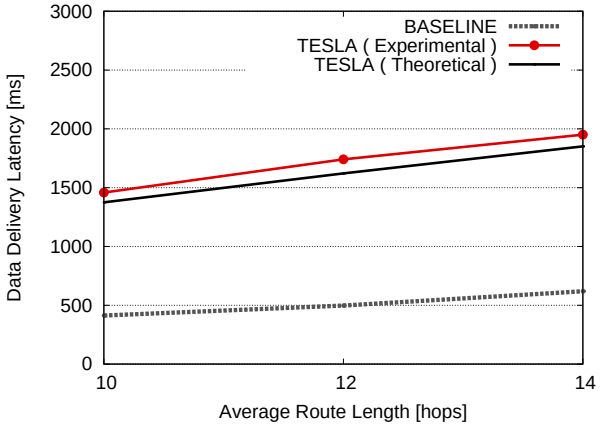


Fig. 13. End-to-end data delivery latency overhead with TESLA

unsecured MP-FPR alternative for routes with 10-14 hops in length respectively. We have varied the hop-count indirectly by increasing the network size, i.e. the number of nodes and deployment area, and kept the spatial distribution of the point-to-point queries unchanged. Given this arrangement, average route lengths of 10, 12 and 14 hops have been achieved from networks of 750, $1,000$ and $1,250$ nodes respectively.

The result presented in Figure 13 confirms that the latency overhead increases linearly with path length, as the analysis in Section IV indicated. To this end, Figure 13 includes the theoretical end-to-end delivery latencies based on the results of Table XI for the path-length considered. It is important to note that the experimental results indicate an approximatively 5% additional latency overhead vs. the theoretical expectations. This is due to several realistic factors that are taken into consideration during simulation, such as transmission delays due to contention in wireless medium – phenomenon that is more pronounced near the sink node where all routes converge.

D. Effectiveness of k-RPEF Against Selective Forwarding

Selective forwarding of QUERY messages Attacks carried during the query dissemination phase target QUERY messages

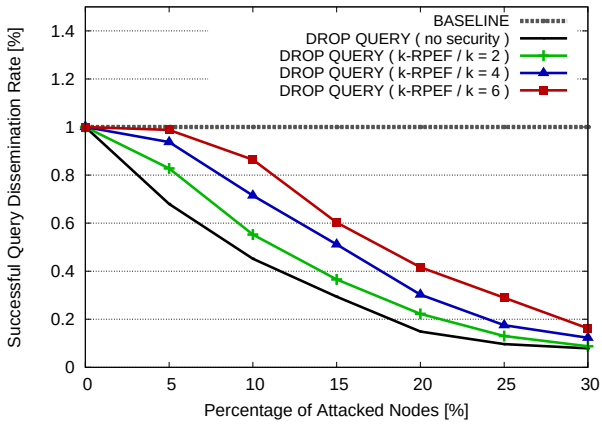


Fig. 14. Impact of selective forwarding of QUERY messages and effectiveness of the k-RPEF defense mechanism

while on-route to the nodes in charge for their processing, i.e. the source nodes. Figure 14 shows that targeting the QUERY messages represents an easy and effective way to block query processing capabilities in the network. For example, by targeting 5% of the sensor nodes, an attacker can expect to impact 30% of the queries submitted. To demonstrate the effectiveness of the k-RPEF replication mechanism, we test against settings with degrees of replication of $k = 2, 4$ and 6. For example, when 6 replicas of QUERY messages are sent, MP-FPR proves to become nearly insensitive to the same small-base of attacks against QUERY messages (5%), with fewer than 1% query dissemination failures. Overall, we note an approximate reduction of successful attacks by 5% for every additional path used for replication, slightly lower under very intense attack settings of more than 25% compromised nodes. This information is relevant for deciding the number of replicas and multi-paths a query message will be sent along, when specific security needs and risk factors are known. Since query submission is an infrequent event, the number of k-RPEF multi-paths can be increased solely based on the security requirement, as the impact on the energy reserves is negligible.

Selective forwarding of ACK messages Dropping ACK messages leads to a similar outcome as to the attacks carried via selective forwarding of RREQ messages, as comparing Figure 15 with Figure 18 demonstrates. Namely, with only a base of 5% of compromised nodes, the effective number of routes have been reduced by nearly 50%, slightly worse than the selective forwarding of RREQ messages.

One fundamental distinction between ACK and RREQ messages in the MP-FPR protocol is that ACK message are not tightly coupled to a particular field line to be forwarded along, hence replicas can be created and forwarded along distinct paths. To this end, Figure 15 demonstrates a significant improvement provided by the k-RPEF mechanism, ranging from approximatively 30% improvement when the degree of replication is $k = 2$, to nearly 100% improvement as the degree of replication is increased to $k = 6$. We can also observe a linear dependency of the improvement to the number

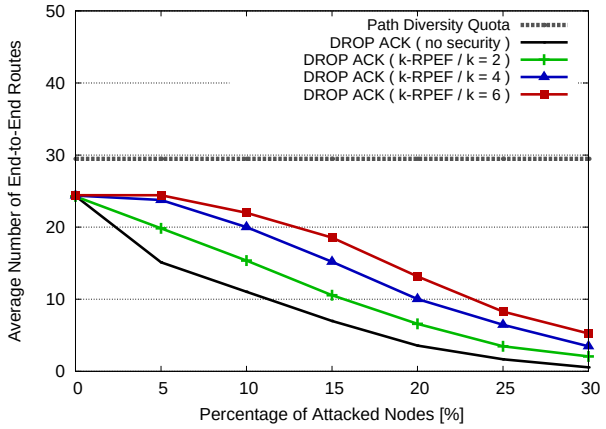


Fig. 15. Impact of selective forwarding of ACK messages to path diversity and effectiveness of k-RPEF defense mechanism

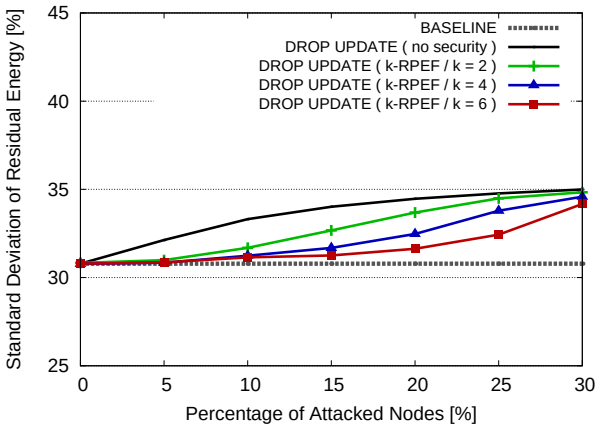


Fig. 17. Impact of selective forwarding of UPDATE messages to energy consumption balancing and the effectiveness of the k-RPEF mechanism

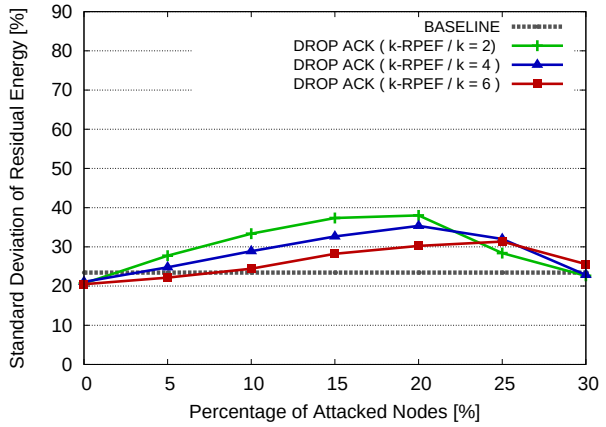


Fig. 16. Impact of selective forwarding of ACK messages to energy consumption balance and effectiveness of k-RPEF defense mechanism

of replicas, each additional replica providing a benefit of 15%, on average, from a resilience perspective to these types of attacks.

The selection of the degree of replication k also impacts the energy balancing, as illustrated in Figure 16. Namely, larger number of replicas promote larger set of routes that improve energy consumption balancing at a rate of approximatively 8% for each additional replica, consistent for attacks comprised of less than 20% nodes. When the attacking base is increased beyond the 20% mark, an apparent improvement of the energy balancing situation similar with the one discussed under the RREQ message dropping manifests.

Because the original MP-FPR protocol sends ACK messages via the SGP mechanism, i.e. along a unique path, it has a higher risk of loosing end-to-end connectivity if the attacks target ACK messages. For example, a single compromised node along the SGP route will compromise the entire route and consequently the entire acknowledgment phase. In practice, it is either the case that (1) no ACK message is lost and end-to-end connectivity is achieved with unaffected families of routes, or (2) *all* ACK messages are being dropped and no routes are established. In both cases, energy is maintained balanced: (1) due to lack of effective workload and (2) due to diverse families of routes. When it comes to the unprotected MP-FPR alternative, the energy imbalance "improved" monotonically as the adversarial activity amplified, due to increased likelihood of end-to-end connectivity loss, reason for which we omit its inclusion in Figure 16.

Selective forwarding of UPDATE messages UPDATE messages contain charge information based on which the non-braiding property of the electrostatic field lines is maintained. Dropping UPDATE messages undermines this property, leading to family path intersection attacks, where increased and uneven energy consumption manifests in the areas were paths pertaining to distinct families of routes start braiding. The effect is more pronounced under high data rate streams where temporary queuing and risk of wireless contention are higher. For this type of experiment we have increased the data delivery transmission rate from $.25$ messages per second to 1 message per second, at each of the four source nodes. Figure 17 illustrates the impact of the wild-path condition attack over the residual energy balancing property. As it can be observed, attacks carried during route establishment phase may yield up to 15% degradation of energy consumption balancing for the data-rate considered. It is important to note that the relative proximity of the source nodes determines the fraction of paths that may intersect and consequently can further impact the level energy imbalance.

Figure 17 also demonstrates that employing the k-RPEF mechanism effectively alleviates the family path intersection attacks. Namely, when the degree of replication is set to $k = 6$, the degradation of energy balancing is maintained below 2% for bases of attacks that cover up to 15% of the nodes, and below 5% degradation when 20% of nodes are compromised.

Sensitivity to degree of replication of k-RPEF. Under long-term queries settings, which represents the motivational basis for the MP-FPR protocol, the amount of traffic gener-

ated by QUERY, ACK and UPDATE messages is minimal, therefore the associated bandwidth and energy costs are insignificant when compared to the large-volume DATA stream. We have, however, demonstrated the incremental benefits of expanding the number of message replicas and sending them along distinct paths (cf. Figures 14, 15, 16, 17). Correspondingly, it can be observed that when the base of attack is reduced, i.e. up to 20% of compromised nodes, increasing the degree of replication k provides an overall benefit of:

- 5% per replica in terms of successful query submissions, considering selective forwarding of QUERY messages,
- 15% for ACK messages flows in terms of number of routes and 8% additional improvement in terms of standard deviation of energy reserves,
- and 8% per replica for UPDATE messages with respect to energy balancing metric.

For all practical purposes, these results may be used as guidelines for selection of the degree of replication given a specific security level requirement.

Considering all the experimental results that were gathered, it can also be consistently observed that the benefit of increasing the degree of replication when the base of attacking nodes is larger than 20% diminishes. This is a consequence of minimal connectivity with respect to a particular message-flow as a result of larger density of compromised nodes in the relay area – situation in which detection/isolation mechanisms are additionally required.

E. Effectiveness of PDMS Against Selective Forwarding

Selective forwarding of RREQ messages. We have simulated path-diversity deflation attacks via selective forwarding of RREQ messages. We note that these experimental results are also representative considering the alternative instrumentation mechanism of significantly delaying of RREQ messages, where paths exhibiting high latencies are not acknowledged. Both mechanisms have an identical adversarial outcome: reduced set of routes, which PDMS will compensate for.

Figure 18 demonstrates the high sensitivity to path diversity deflation attacks, as even with a small base of 5% compromised nodes, the number of paths is effectively reduced by 40% as compared to the non-adversarial settings. Enabling PDMS functionality significantly improves the resilience to route establishment attacks, as for the same base of attacking nodes, the reduction of alternative paths is of only 6%. Consequently, the attacker needs to consider tripling the attacking base, i.e. targeting approximatively 15% sensor nodes instead of 5% nodes, to achieve the same damaging effect as in the unprotected MP-FPR. From a different perspective, under the same adversarial conditions, PDMS scheme enables achievement of up to 140% richer families of routes as compared to MP-FPR under an unprotected adversarial context.

Figure 18 illustrates an additional benefit of PDMS: improving path diversity even under non-adversarial conditions. Namely, even when there are no compromised nodes, MP-FPR yields an average of 17% fewer routes than the user-specified quota ($N_r = 30$ in these settings). This is because

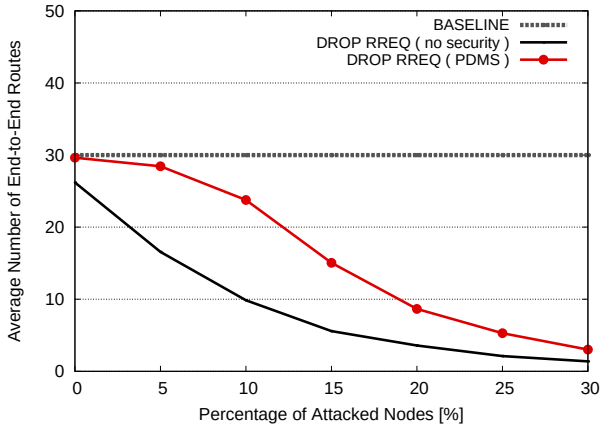


Fig. 18. Impact of selective forwarding of RREQ messages to path diversity and the effectiveness of PDMS reactive mechanism

MP-FPR discards routes that do not meet the end-to-end latency requirements (cf. Section II), such as overly long paths caused due to bandwidth starvation, long field lines or link quality issues and it does not compensate for. PDMS implicitly addresses this issue by persisting in probing routes until the path diversity quota is being met, as PDMS is oblivious of the underling reasons for which certain routes are not acknowledged. Therefore, PDMS represents, in addition, a feature enhancement of the original MP-FPR. The end-benefit can also be observed in Figure 19, according to which the PDMS scheme achieves a 12% improvement in terms of energy balancing over MP-FPR in non-adversarial environments (zero compromised nodes).

The direct consequence of attacks carried during route establishment is a reduction of the effectiveness of the workload balancing. Figure 19 illustrates the depreciation of energy-balancing as the number of compromised nodes is increased, where it can be observed that there is an 110% increase in standard deviation of the residual energy levels when only 10% of the nodes are compromised. PDMS helps maintaining even energy consumption distribution, achieving below 15% depreciation under the same scenarios – a significant improvement over the unprotected MP-FPR. The workload imbalance tops with 175% depreciation when 20% nodes maliciously drop RREQ messages, and ”recover” as the number of attacks is further increased. We recall that the apparent recovery is due to the loss of end-to-end connectivity. When absolutely no routes can be established between the source and sink nodes due to very large base of compromised nodes, the data stream becomes virtually absent and the afferent messages are dropped at the source. Energy savings are being achieved in the relay-area due to the lack of the data stream workload. To demonstrate that this is the case, we analyze in sequel the impact of attacks carried via selective forwarding of RREQ messages over the data delivery ratio.

As it can be observed in Figure 20, the sensitivity to message-dropping of RREQ messages is significantly reduced when compared to the reduction in path diversity under the

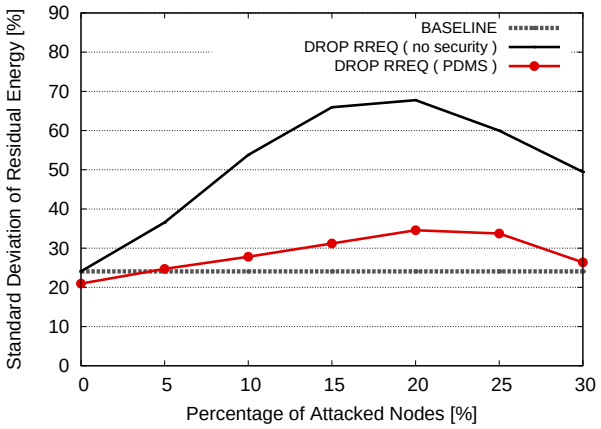


Fig. 19. Impact of selective forwarding of RREQ messages to energy consumption balance and effectiveness of PDMS reactive mechanism

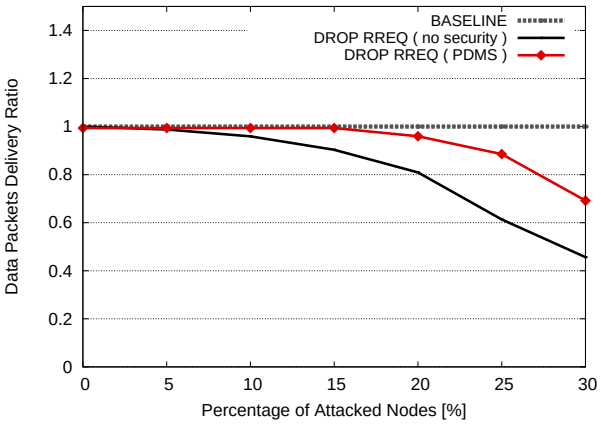


Fig. 20. Impact of selective forwarding of RREQ messages to end-to-end data-stream delivery ratio and the effectiveness of PDMS reactive mechanism

same settings. Namely, when 5% of nodes are compromised, the impact to message dropping is below 1%. This is because the diminution of path-diversity does not affect message delivery, but the total absence of connecting routes does. As it can be observed, when the base of attacks is increased to 30% nodes, the average number of disconnected source-to-sink topologies is around 50%. The PDMS enables higher data-message delivery ratios since the family of routes it yields is consistently larger and the risk of non-connectivity is consequently lowered. PDMS forces an attacker to consider a much larger base of attacking nodes, an average of 20% more, to render PDMS scheme just as ineffective in achieving end-to-end connectivity as with the unprotected MP-FPR, with respect to the data stream deliverability.

Compensatory effect of PDMS to k-RPEF during attacks via selective forwarding of ACK messages Both k-RPEF and PDMS mechanisms provide protection against path diversity deflation under adversarial conditions. However, these two mechanisms are fundamentally different: k-RPEF is a *proactive* mechanism, whereas PDMS is *reactive*. Namely, k-RPEF

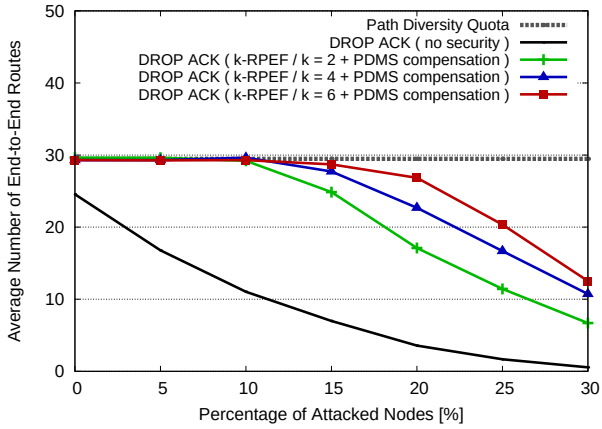


Fig. 21. Impact of selective forwarding of ACK messages to path diversity and the effectiveness of k-RPEF and PDMS solution mix

attempts to reduce the risk of failing to acknowledge a route, while PDMS attempts to build a new route if one has already failed. Since dropping either of ACK and RREQ messages leads to a route construction failure, PDMS will compensate for both in an attempt to meet the path diversity quota. That is, PDMS, when employed, will react to dropping of ACK messages as well. While we have analyzed k-RPEF and PDMS solutions in isolation, we do make note of this compensatory effect of the PDMS mechanism to the k-RPEF. Therefore, we are compelled to present an experimental analysis where both of these methods are concomitantly employed.

Figure 21 illustrates the improvement in path diversity when PDMS mechanism is enabled to provide compensation to the standalone k-RPEF mechanism. As it can be seen, this combination provides a virtually perfect defense against selective forwarding of ACK messages when the base of compromised nodes is below 10% as path diversity remains unaffected. Moreover, the PDMS component enables MP-FPR to reach the path diversity quota even under this adversarial scenario. It takes a large base of compromised nodes, i.e. at least 30% of the total number of sensor nodes, to achieve comparative protection of k-RPEF running in isolation against 20% of compromised nodes. From the perspective of sheer resilience to adversarial activity, PDMS improves the performance of k-RPEF, on average, by 90%.

It is important to mention that PDMS, in isolation, cannot provide any benefit against selective forwarding of ACK messages. This is due to the SGP mechanism employed for relaying ACK messages in the original MP-FPR, as it was previously discussed. That is, if the SGP established sink-to-source path is compromised, *all* ACK messages will be dropped, including those acknowledging routes that PDMS attempts to build as replacement. In other words, compromising the unique route in SGP mechanism effectively nullifies the PDMS's benefits with respect to selective forwarding of ACK messages.

Energy balancing also benefits by enabling the PDMS to operate in conjunction with the k-RPEF solution. As Figure 22

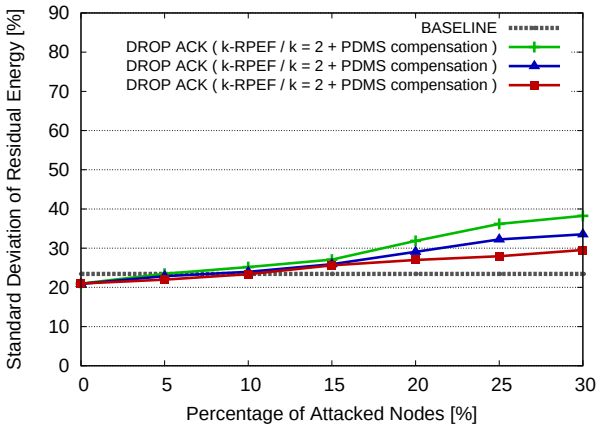


Fig. 22. Impact of selective forwarding of ACK messages to energy consumption balance and the effectiveness of k-RPEF and PDMS mix

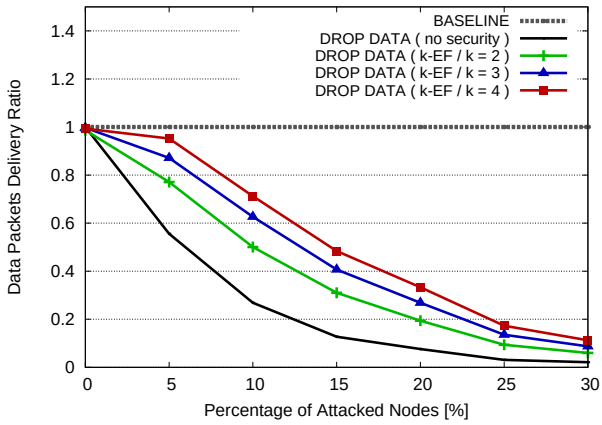


Fig. 23. Impact of selective forwarding of DATA messages to delivery reliability of the data stream and the effectiveness of the k-EF approach

demonstrates, considering a degree of replication of $k = 6$ and an attacking base of 20%, the disruption of energy balancing is of only 16%, i.e. a nearly 50% improvement when compared to the equivalent performance of running k-RPEF in isolation (cf. previous results in Figure 16).

F. Effectiveness of k-EF Against Selective Forwarding

Selective forwarding of DATA messages. Lastly, we study the impact level of data DoS carried via selective forwarding of DATA messages, as well as the efficacy of applying a multipath strategy via k-EF mechanism. It involves using subsets of acknowledged routes, rather than on-demand paths as in k-RPEF. Due to the high-volume of data traffic, replication of such traffic must be limited in order to avoid: (1) wasting resources and (2) bandwidth saturation, especially considering the proximity of the sink node where data flows converge. To this end, we have tested scenarios with degree of replication of $k = 2, 3$ and 4 only.

Figure 23 illustrates the consequence of increasing the number of attacking nodes that target DATA messages: a 45% degradation in DATA packet delivery with a only a small base of 5% nodes, and nearly 90% degradation when the number of compromised nodes is increased to 15%. This vulnerability is particularly important as the user-payload within dropped DATA messages cannot be recovered. Adopting a multipath approach proves to be beneficial in this situation as well: at the minimum, the effect is reduced by a factor of two, i.e. from 45% to 23% message drops when only 2 replication paths are used, and less than 2% when 4 replication paths are used, considering 5% compromised nodes. This relative improvement is consistent regardless of the number of compromised nodes. From an attacker standpoint, the effort required to achieve the same net effect as over an unprotected MP-FPR nearly doubles, considering, for example, 4 replication paths.

The cost of providing protection against data DoS via selective forwarding of DATA messages is reflected in increased energy consumption. For example, assuming a secure environment, i.e. number of compromised nodes is zero, Figure

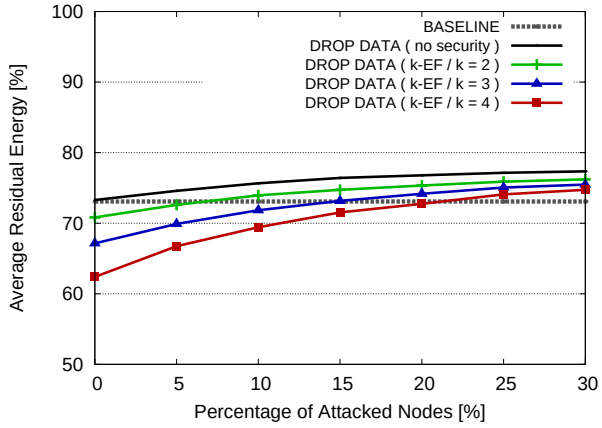


Fig. 24. Residual energy levels overhead of k-EF defense mechanisms against selective forwarding of DATA messages

24 shows an overhead varying between 3% and 15% as the number of multipaths is increased from $k = 2$ to $k = 4$. While the overhead is small, it can be much higher if the rate of transmission of data messages increases, currently set at $.25$ messages per seconds. It is also important to note that the number of compromised nodes does not have a direct negative impact over the energy consumption. It is, however, the case that energy savings are achieved when DATA messages are being dropped along a path due to an undesirable reduction of the workload. As it can be observed in Figure 24, the residual energy reserves increases monotonically corresponding to the reduction of the successful delivery of data messages from Figure 23.

Sensitivity to the degree of replication of k-EF. Relaying a large-volume DATA stream from source nodes towards a sink node has an energy and bandwidth associated cost that cannot be ignored. Specifically, Figures 23 and 24 represent the benefit, respectively the cost, of increasing the degree of replication. A cost-benefit analysis is application and/or query specific, as it needs properly weighting of the user-requirements with respect with the sensitivity to data delivery,

the time-span during which this information can be collected, the size and transmission rate of the payload, and ultimately the security risks the network is exposed to. Due to the complexity of the variables involved, we resume to present a set of guidelines.

As an example, if the application domain for which the user collects data comprise correlation analysis or outlier detection in which completeness of the data stream has high priority, under reduced security risk scenarios, the system may be configured to use a higher degree of replication. For example, if $k = 4$, i.e. 4 distinct paths are employed to relay copies of a given DATA message, under 5% compromised nodes settings, it is expected a success rate of data-stream delivery of 98% (cf. Figure 23). However, under the same settings, the maximum time-span for information delivery is projected to be reduced by 15%, considering DATA messages transmission rate of $.25$ messages per second. The projection is based on a corresponding reduction of the average residual energy reserves (cf. Figure 24), expectedly lower under increasing data rates.

Overall, each increment of the degree of replication has an added benefit of approximatively 5% improvement of successful DATA stream delivery, at a cost of 1% energy consumption under the DATA message transmission rates considered, yielding a cost-benefit ratio of 1:5. If the DATA flow volume will increase beyond the experimental settings we have considered, the cost-benefit ratio will consequently lower.

IX. Related Work

Recent work on the security of sensor networks has focused on proposing key management schemes that can be used to bootstrap other services [25], [20], [19], [24], [51], addressing general attacks such as Sybil [55] and replication [59] attacks, as well as identifying basic attacks in wireless sensor networks [37].

The security of geographical routing protocols using physical nodes' locations was studied in [9] for sensor networks and in [45], [68] for ad-hoc networks. Most of the works focus on preventing malicious modifications of the destination location in packets, verifying neighbor location information, and preventing message dropping. Another main area of work in securing geographic routing is the protection of the location service, which includes [76], [23].

Security of a gradient based routing approach, namely the potential-field routing for sensor networks, has been investigated in [67]. This work, however, distinguishes from our approach in the following aspects: (1) the work surveys a generic list of attacks and countermeasures that do not focus on the specifics of the potential-field routing, while we address specific risks introduced by the MP-FPR protocol in all phases of the protocol operation, from query dissemination and charge allocation to route establishment and data forwarding, and analyze these risk factors through extensive experimental analysis; (2) although potential-field routing and electrostatic field-based routing are both instances of the gradient based routing, their implementation is fundamentally different: the former is a *stateful* protocol, where routes are established based on distance metrics obtained by means of hop-counting, while MP-FPRdoes not maintain routing information and relies only on the distribution of discrete charge information for forwarding purposes; (3) field-based routing has been proposed initially in the context of large scale, dense mesh networks and there is no focus on energy consumption and workload distribution, whereas MP-FPR generalizes the usability of gradient based routing to arbitrary distributions with possible low densities of nodes and focuses on the energy aspect.

Geographic routing remains a promising and active area of research due to intrinsic benefits of exploiting location relationships for routing purposes. A complete survey of geography-based single-path routing approaches can be found in [64], whereas a newer approach that particularly considers the challenges of large scale sensor networks is presented in [38]. Other works have also recognized the benefits of using multipath routing in large-scale sensor networks for improving workload balancing and delivery robustness. For example, trajectory-based forwarding approaches, which rely on multiple non-braided paths via parametric curves for single source and sink scenarios, have been presented in [22], [31]. A natural extension to multiple sink, multiple-path is challenging because route disjointness cannot be easily guaranteed when adopting parametric trajectory models, therefore field, potential and gravity-based routing methodologies, which exploit physical phenomena properties to facilitate the creation of non-braiding paths, have been recently investigated [75], [49], [65]. Despite the broad interest in gradient based routing, very little work has been done to address the security aspect of such advanced protocols, which constitutes the motivational support for this body of work.

X. Conclusions and Future Work

In this article, we have presented an in-depth analysis regarding the feasibility of providing security semantics to Field Persistent Routing (MP-FPR) – an instance of the electrostatic field based routing for location-aware sensor networks. We have identified the attacking model and the core system properties that uniquely characterize MP-FPR's settings. Several cryptographic mechanisms have been investigated for providing integrity and authentication primitives, considering both public (TinyECC) and symmetric (PIKE) key cryptography as possible solutions, as well as a hybrid approach (TESLA). Subsequently, we have investigated an orthogonal problem that concerns the attacks carried via selective forwarding of certain protocol messages. Correspondingly, three complementary solutions were proposed that exploit the native multi-path nature of MP-FPR, in order to improve resilience to such attacks: k-EF, k-RPEF and PDMS. Lastly, we recognized the importance of providing a reactive mechanism for attack detection and isolation – a broader topic that requires a separate in-depth investigation that will be pursued as a future work.

Since the MP-FPR mechanism stresses the importance of energy-efficiency and energy-consumption balancing for ex-

tending the useful lifetime of WSNs, a particular attention has been given to changes to the energy-consumption patterns induced by the security primitives. Accordingly, in addition to performance metrics such as packet-delivery latencies and success ratio, memory, bandwidth and processing overhead, we have also taken into account the overall energy-overhead expressed as the network-wide cumulative residual energy, as well as the standard deviation of the nodes' energy levels as a measure of energy-consumption balancing. We have experimentally demonstrated that MP-FPR energy provisions can be significantly affected under an adversarial environment, however, effective security solutions that exploit MP-FPR's multi-path routing model can be implemented with minimal overhead.

REFERENCES

[1] http://www.eecs.northwestern.edu/~ocg474/SIDnet.html.
[2] http://jist.ece.cornell.edu/index.html.
[3] http://www.xbow.com.
[4] Imote2: High-performance wireless sensor network node.
[5] Mica2dot mote platform.
[6] Micaz: Wireless measurement system.
[7] Telosb mote platform.
[8] Tmote sky: Reliable low-power wireless sensor networking eases development and deployment.
[9] Nael Abu-Ghazaleh, Kyoung-Don Kang, and Ke Liu. Towards resilient geographic routing in wsns. In *Q2SWinet*, 2005.
[10] Kemal Akkaya and Mohamed F. Younis. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3(3):325–349, 2005.
[11] Ian Fuat Akyildiz and Mehmet Can Vuran. *Wireless Sensor Networks*. Willey, 2010.
[12] Waleed Ammar, Ahmed ElDawy, and Moustafa Youssef. Secure localization in wireless sensor networks: A survey. *CoRR*, abs/1004.3164, 2010.
[13] Yi an Huang and Wenke Lee. A cooperative intrusion detection system for ad hoc networks. In *SASN*, pages 135–147, 2003.
[14] Baruch Awerbuch, Reza Curtmola, David Holmer, Cristina Nita-Rotaru, and Herbert Rubens. Odsbr: An on-demand secure byzantine resilient routing protocol for wireless ad hoc networks. *ACM TISSEC*, 10(4), 2008.
[15] Baruch Awerbuch, Reza Curtmola, David Holmer, Herbert Rubens, and Cristina Nita-Rotaru. On the survivability of routing protocols in ad hoc wireless networks. In *IEEE SECURECOMM*, pages 327–338. IEEE Computer Society Press, 2005.
[16] Johannes Barnickel and Ulrike Meyer. Secsywise: A secure time synchronization scheme in wireless sensor networks. In *ICUMT*, pages 1–8, 2009.
[17] Rainer Baumann, Simon Heimlicher, Vincent Lenders, and Martin May. Heat: Scalable routing in wireless mesh networks using temperature fields. In *WOWMOM*, pages 1–9, 2007.
[18] Haowen Chan. Pike: Peer intermediaries for key establishment in sensor networks. In *In Proceedings of IEEE Infocom*, pages 524–535, 2005.
[19] Haowen Chan and Adrian Perrig. PIKE: Peer intermediaries for key establishment in sensor networks. In *INFOCOM*, 2005.
[20] Haowen Chan, Adrian Perrig, and Dawn Song. Random key predistribution schemes for sensor networks. In *SP*, 2003.
[21] Haowen Chan, Adrian Perrig, and Dawn Xiaodong Song. Random key predistribution schemes for sensor networks. In *IEEE Symposium on Security and Privacy*, pages 197–, 2003.
[22] Maulik Desai and Nicholas Maxemchuk. Polar cooordinate routing for multiple paths in wireless sensor networks. In *WOWCOM*, pages 1–9, 2010.
[23] Jing Dong, Kurt E. Ackermann, Brett Bavar, and Cristina Nita-Rotaru. Mitigating attacks against virtual coordinate based routing in wireless sensor networks. In *WiSec*, pages 89–99, New York, NY, USA, 2008. ACM.
[24] Wenliang Du, Jing Deng, Yunghsiang S. Han, Pramod K. Varshney, Jonathan Katz, and Aram Khalili. A pairwise key predistribution scheme for wireless sensor networks. *TISSEC*, 8(2), 2005.
[25] L. Eschenauer and V. Gligor. A key management scheme for distributed sensor networks. In *CCS*, 2002.
[26] Laurent Eschenauer and Virgil D. Gligor. A key-management scheme for distributed sensor networks. In *ACM CCS*, pages 41–47. ACM Press, 2002.
[27] Rodrigo Fonseca, Sylvia Ratnasamy, Jerry Zhao, Cheng T. Ee, David Culler, Scott Shenker, and Ion Stoica. Beacon vector routing: Scalable point-to-point routing in wireless sensornets. In *NSDI*, 2005.
[28] Saurabh Ganeriwal, Christina Pöpper, Srdjan Capkun, and Mani B. Srivastava. Secure time synchronization in sensor networks. *ACM TISSEC*, 11(4), 2008.
[29] Deepak Ganesan, Ramesh Govindan, Scott Shenker, and Deborah Estrin. Highly-resilient, energy-efficient multipath routing in wireless sensor networks. *Mobile Computing and Communications Review*, 5(4):11–25, 2001.
[30] Oliviu Ghica, Goce Trajcevski, Peter Scheuermann, Zachary Bischoff, and Nikolay Valtchanov. Sidnet-swans: A simulator and integrated development platform for sensor networks applications. In *SenSys*, 2008.
[31] Oliviu Ghica, Goce Trajcevski, Peter Scheuermann, Nikolay Valtchanov, and Zachary Bischof. Controlled multi-path routing in sensor networks using bezier curves. *The Computer Journal*, 54(2):230–254, 2011.
[32] Darrel Hankerson, Alfred J. Menezes, and Scott Vanstone. *Guide to Elliptic Curve Cryptography*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2003.
[33] Charles Harsch, Andreas Festag, and Panos Papadimitratos. Secure position-based routing for vanets. In *VTC Fall*, pages 26–30, 2007.
[34] Tian He, Chengdu Huang, Brian M. Blum, John A. Stankovic, and Tarek F. Abdelzaher. Range-free localization schemes for large scale sensor networks. In *MOBICOM*, pages 81–95, 2003.
[35] Don Johnson, Alfred Menezes, and Scott A. Vanstone. The elliptic curve digital signature algorithm (ecdsa). *International Journal of Information Security*, 1(1):36–63, 2001.
[36] Sung ju Lee. Split multipath routing with maximally disjoint paths. In *in Ad hoc Networks, in Proc. of IEEE ICC*, 2001.
[37] Chris Karlof and David Wagner. Secure routing in wireless sensor networks: Attacks and countermeasures. In *WSNA*, 2003.
[38] Anne-Marie Kermarrec and Guang Tan. Greedy geographic routing in large-scale sensor networks: A minimum network decomposition approach. In *MobiHoc*, 2010.
[39] Issa Khalil, Saurabh Bagchi, Cristina Nita-Rotaru, and Ness B. Shroff. Unmask: Utilizing neighbor monitoring for attack mitigation in multihop wireless sensor networks. *Ad Hoc Networks*, 8(2):148–164, 2010.
[40] N. Koblitz. Elliptic curve cryptosystems. *Mathematics of Computation*, 48:203–209, 1987.
[41] Ajay Koul, R. B. Patel, and V. K. Bhat. Double split based secure multipath routing in adhoc networks. In *ARTCom*, pages 835–839, 2009.
[42] H. Krawczyk, M. Bellare, and R. Canetti. Hmac: keyed-hashing for message authentication. *RFC*, 2104:1–12, 1997.
[43] Praveen Kumar, Joy Kuri, and Pavan Nuggehalli. Connectivity-aware routing in sensor networks. In *Sensorcomm, IEEE*, 2007.
[44] H Lee, Y Choi, and H Kim. Implementation of tinyhash based on hash algorithm for sensor network. In *Proc. of World Academy of Science, Engineering, and Technology*, 2005.
[45] Tim Leinmüller, Christian Maihöfer, Elmar Schoch, and Frank Kargl. Improved security in geographic ad hoc routing through autonomous position verification. In *VANET*, 2006.
[46] Colin Lemmon, Siu Man Lui, and Ickjai Lee. Geographic forwarding and routing for ad-hoc wireless network: A survey. In *NCM*, pages 188–195, 2009.
[47] Vincent Lenders, Martin May, and Bernhard Plattner. Service discovery in mobile ad hoc networks: A field theoretic approach. *Pervasive and Mobile Computing*, 1(3):343–370, 2005.
[48] Vincent Lenders, Martin May, and Bernhard Plattner. Density-based anycast: a robust routing strategy for wireless ad hoc networks. *IEEE/ACM Transactions in Networking*, 16(4):852–863, 2008.
[49] Jinbao Li, Shouling Ji, Hu Jin, and Qianqian Ren. Routing in multi-sink sensor networks based on gravitational field. In *ICESS*, pages 368–375, Washington, DC, USA, 2008. IEEE Computer Society.
[50] An Liu and Peng Ning. Tinyecc: A configurable library for elliptic curve cryptography in wireless sensor networks, 2008.
[51] Donggang Liu, Peng Ning, and Rongfang Li. Establishing pairwise keys in distributed sensor networks. *TISSEC*, 8(1), 2005.

[52] Sergio Marti, Thomas J. Giuli, Kevin Lai, and Mary Baker. Mitigating routing misbehavior in mobile ad hoc networks. In *MOBICOM*, pages 255–265, 2000.

[53] S. P. Miller, B. C. Neuman, J. I. Schiller, and J. H. Saltzer. Kerberos authentication and authorization system. In *In Project Athena Technical Plan*, 1988.

[54] Asis Nasipuri, Robert Castañeda, and Samir Ranjan Das. Performance of multipath routing for on-demand protocols in mobile ad hoc networks. *MONET*, 6(4):339–349, 2001.

[55] James Newsome, Elaine Shi, Dawn Song, and Adrian Perrig. The sybil attack in sensor networks: analysis & defenses. In *IPSN*, 2004.

[56] N. T. Nguyen, A. Wang, P. Reiher, and G. Kuenning. Electric-field-based routing: a reliable framework for routing in MANETs. *SIGMOBILE Mobile Computing Communication Review*, 8(2):35–49, 2004.

[57] D. Niculesu and B. Nath. Trajectory based forwarding and its applications. In *MOBICOM*, 2003.

[58] Panagiotis Papadimitratos and Zygmunt J. Haas. Secure message transmission in mobile ad hoc networks. *Ad Hoc Networks*, 1(1):193–209, 2003.

[59] Bryan Parno, Adrian Perrig, and Virgil Gligor. Distributed detection of node replication attacks in sensor networks. In *SP*, 2005.

[60] Adrian Perrig, Ran Canetti, J. D. Tygar, and Dawn Xiaodong Song. Efficient authentication and signing of multicast streams over lossy channels. In *IEEE Symposium on Security and Privacy*, pages 56–73, 2000.

[61] Adrian Perrig, Robert Szewczyk, Victor Wen, David Culler, and J. D. Tygar. Spins: Security protocols for sensor networks. In *Wireless Networks*, pages 189–199, 2001.

[62] Asad Amir Pirzada and Chris McDonald. Establishing trust in pure ad-hoc networks. In *ACSC*, pages 47–54, 2004.

[63] Sylvia Ratnasamy, Brad Karp, Li Yin, Fang Yu, Deborah Estrin, Ramesh Govindan, and Scott Shenker. Ght: a geographic hash table for data-centric storage. In *WSNA*, pages 78–87, 2002.

[64] Stefan Ruehrup. Theory and practice of geographic routing. In *Ad Hoc and Sensor Wireless Networks: Architectures, Algorithms and Protocols*. Bentham Science, 2009.

[65] Stefan Rührup, Hanna Kalosha, Amiya Nayak, and Ivan Stojmenovic. Message-efficient beaconless georouting with guaranteed delivery in wireless sensor, ad hoc, and actuator networks. *IEEE/ACM Transactions Networking*, 18(1):95–108, 2010.

[66] Kimaya Sanzgiri, Bridget Dahill, Brian Neil Levine, Clay Shields, and Elizabeth M. Belding-Royer. A secure routing protocol for ad hoc networks. In *ICNP*, pages 78–89, 2002.

[67] Divya Sharma. Security of field based routing. *Student Thesis SA-2008-08*, 2008.

[68] Joo-Han Song, Vincent W. S. Wong, and Victor C. M. Leung. Secure position-based routing protocol for mobile ad hoc networks. *Ad Hoc Networks*, 5(1):76–86, 2007.

[69] J. Spencer. *The Strange Logic of Random Graphs*. Springer-Verlag, 2008.

[70] Bharath Sundararaman, Ugo Buy, and Ajay D. Kshemkalyani. Clock synchronization for wireless sensor networks: a survey. *Ad Hoc Networks*, 3(3):281–323, 2005.

[71] Hailun Tan. On mitigating malicious behavior against routing in wireless networks. In *WCNC*, 2007.

[72] Stavros Toumpis. Mother nature knows best: A survey of recent results on wireless networks based on analogies with physics. *Computer Networks*, 52(2):360–383, 2008.

[73] Goce Trajcevski, Oliviu C. Ghica, Peter Scheuermann, Marco Zuniga, Rene Schubotz, and Manfred Hauswirth. Improving the energy balance of field-based routing in wireless sensor networks. In *IEEE Globecom*, 2010.

[74] Rebecca Wright, Patrick D. Lincoln, and Jonathan K. Millen. Efficient fault-tolerant certificate revocation. In *In ACM Conference on Computer and Communications Security*. ACM CCS, 2000.

[75] Chengjie Wu, Ruixi Yuan, and Hongchao Zhou. A novel load balanced and lifetime maximization routing protocol in wireless sensor networks. In *VTC Spring*, pages 113–117, 2008.

[76] Xiaoxin Wu and Cristina Nita-Rotaru. On the security of distributed position services. In *SecureComm*, 2005.

[77] Sencun Zhu, Chao Yao, Donggang Liu, Sanjeev Setia, and Sushil Jajodia. Efficient security mechanisms for overlay multicast-based content distribution. In *ACNS*, pages 40–55, 2006.

[78] Wen Tao Zhu and Yang Xiang. Argus: A light-weighted secure localization scheme for sensor networks. In *ATC*, pages 164–178, 2009.