Detecting Route Attraction Attacks in Wireless Networks

Mustafa Y. Arslan*, Konstantinos Pelechrinis†, Ioannis Broustis*,
Srikanth V. Krishnamurthy*, Prashant Krishnamurthy†, Prasant Mohapatra‡

*University of California, Riverside: {marslan, broustis, krish}@cs.ucr.edu
†University of Pittsburgh: {kpele, prashant}@mail.sis.pitt.edu
‡University of California, Davis: prasant@cs.ucdavis.edu

Abstract—Selecting high performance routes in wireless networks requires the exchange of link quality information among nodes. Adversaries can manipulate this functionality by advertising fake qualities for links; by doing so, they can attract routes and subsequently launch pernicious attacks. Our measurements suggest that malicious route attraction can fatally impact throughput. We design a framework that is effective against both independent and colluding attackers. In the latter case, we consider both local and remote colluders. With local collusion, malicious nodes exchange and advertise fake routing information to increase the probability of being selected as relays. Remote collusion refers to nodes residing in distant parts of the network that (i) create sybil identities in a local neighborhood and / or (ii) utilize link quality reports to advertise fake links. Our framework combines packet signing and frequency hopping to accurately detect the adversaries. We implement the framework on our testbed and conduct experiments to assess its efficacy. We observe that our framework provides significant throughput benefits by detecting attackers with 90% accuracy.

I. INTRODUCTION

Selecting high performance routes requires nodes to assess the quality of the individual links, typically in terms of Packet Delivery Ratio (PDR) [1], [2]. To calculate the PDR, routing protocols rely on the periodic transmission of *probe* packets. Based on the number of received probes, a node reports the PDR on its incident links to its neighbors.

One vulnerability of PDR reports is that an attacker can report fake PDR values for the links to its neighbors. As an example, it can advertise low (lower than the actual) PDR values, thereby evading relay operations and causing packets to follow lower-quality paths [3]. Such an attack however, may not be as threatening since sources may still find alternative paths of good quality. In our work, we focus on the opposite, more challenging malicious strategy: an attacker can advertise high (higher than the actual) PDR values for its links, thereby increasing the probability of becoming a relay. We call these nodes (that manipulate the PDR to "attract" the routes) *magnets* and the specific attack, a *route magnet attack* [1]. As long as a magnet can attract a route, it can easily launch any attack on packets (e.g., the blackhole attack [4]).

In this paper we design and implement RoMaD, a Route Magnet Detection framework that identifies such malicious PDR manipulation and thus, can help avert paths from magnets. We study the topological characteristics that (a) render the network vulnerable to the attack, and (b) trigger the online detection and mitigation of the magnets. We highlight our main contributions in what follows:

1. Experimental study of strategies for attracting paths based on PDR broadcasts: We perform an in-depth study of both independent and colluding route magnets via measurements on an indoor / outdoor wireless testbed. We identify three collusion methods. First, two or more magnets can locally collude against legitimate nodes; they can exchange topological information and report fake link qualities. Second, colluding magnets that are in distant parts of the network can exchange credentials (i.e. authentication keys), and launch sybil attacks in local neighborhoods. Third, distant magnets *with no common neighbors* can use the probing functionality to create the illusion of fake "shortcut" links between a source and its destination [2].

2. Designing a framework for the detection of the attack: We design and implement RoMaD, a unified route magnet detector, which accurately detects all of the aforementioned attacks, based on online observations. Our solution consists of two components: NRC (Nonce Report Component) and MCC (Multiband Challenge Component). NRC ensures the truthfulness of probe reports through nonces and constructs a list of potential magnets; MCC detects attackers by challenging suspect nodes on different channels and with variable power levels.

3. Evaluating the efficacy of RoMaD: We implement RoMaD and evaluate it on a large-scale testbed at UC Riverside [5]. We observe that RoMaD accurately detects the magnets in all the considered cases; we also discuss the false positive and false negative scenarios that can arise.

While previous work (e.g., [6], [7]) allude to using nonces and frequency challenges, they lack an implementation on an actual testbed. We are the first to integrate, implement and evaluate the efficacy of these approaches on a real experimental platform. The distinguishing aspects of our study are:

• **First,** existing solutions on routing metric manipulation are *reactive*: they assume that the metric is correctly calcu-

---

[1]The attack resembles sinkhole attacks and we are not the first to identify this general class of attacks. We use the term route magnet as a naming convention.

[2]These attacks differ from traditional wormhole attacks (see Section III).

lated before secure countermeasures are applied. In contrast, our scheme is *proactive*: it *a priori* secures the metric calculation, i.e., before route discovery and maintenance.

We believe that a proactive approach is essential to effectively tackle attackers that manipulate link quality reports. To illustrate this with an example, let us assume a network that employs the DSDV routing protocol [8]. DSDV proactively discovers new routes based on periodic routing advertisements from nodes. The probes used to compute the PDR are also sent periodically even if there is no active data transmission in the network. Coupled with the probes, DSDV discovers new routes (based on PDR) and advertises this information to other nodes. If countermeasures are not applied at the probing stage, the PDR (*as manipulated by the attackers*) would result in manipulated route dissemination in the network. For this reason, the reactive nature of the previous approaches (e.g., [9], [10], [11]) renders them ineffective in addressing malicious PDR manipulation. RoMaD, instead, takes a proactive approach to detect the manipulation at its early stages (while incurring some overhead in detecting the attackers).

• **Second,** we are the first to consider colluding magnets that use diverse attack strategies. Previous work on magnets assume only individual attackers [6], [15].

• **Third,** unlike most prior work we tackle route magnets that are *insiders* (i.e., compromised nodes).

The rest of the paper is structured as follows. In Section II, we provide brief background on link quality based routing and discuss relevant previous work. In Section III, we profile the attack strategies that magnets can use. We present the design of RoMaD in Section IV. In Section V, we evaluate RoMaD. We discuss future work in Section VI and our conclusions form Section VII.

II. BACKGROUND AND RELATED WORK

PDR-based routing metrics: Modern routing metrics account for the quality of individual links. The Expected Transmission Count, ETX [1], of a link corresponds to the expected number of transmissions (including retransmissions) needed for the successful reception of a packet on the link. In particular, $ETX = 1/(p_f \cdot p_r)$, where p_f and p_r are the PDR values on the forward and the reverse directions of the link, respectively. The ETX cost of a route is the sum of the ETX values of the links on the route. The source node selects the route with the minimum ETX cost. Similarly, other metrics such as ETT and WCETT [2] use PDR information to compute the metric.

All of the above metrics rely on a real-time probing functionality in order to measure the PDR. To achieve this, each node periodically broadcasts probe packets every τ seconds. Nodes report the number of probes received from each neighbor during a window of ω seconds. This information is piggybacked in future probes; it is used to compute the PDR for both directions of a link and thereby, its corresponding metric. While a node can directly measure the reverse PDR, p_r , for neighbor links (i.e., the PDR of the link *from* a neighbor), it relies on the reports from its

neighbors to learn about the forward PDR, p_f (i.e., the PDR of the link *to* a neighbor). If malicious nodes report very high PDR values for their links (affecting the p_f for their neighbors and resulting in a lower ETX), they can coerce source nodes into selecting them as relays. Hence, the goal of a route magnet is to assume the role of a relay on routes that are selected based on any of the above metrics, by reporting fake, high PDR values. Note that this vulnerability is also inherited by any other metric that leverages a similar probing mechanism to measure the PDR.

Secure routing: There have been studies on specific versions of route attraction attacks. In [9], SAODV, a secure version of AODV [12] is proposed to prevent the modification of the hop count metric by adversaries. ARIADNE [10] is a secure version of DSR [13]; it also protects routing messages from being altered by adversaries. SEAD [11] protects the metric and sequence number fields in DSDV [8]. All of these studies inherently assume that the routing metric is truthfully calculated and secure measures are used only to subsequently protect the metric. In our work however, the PDR-based metric can be manipulated even before the route discovery process. In such cases, the methods in above studies would try to protect a metric information that is not truthfully calculated in the first place! Hence, it is essential to secure the route setup operations proactively (as we do in this work), rather than trying to avert adversaries after route establishment.

Studies on route magnets: In [3], a solution is proposed to tackle the problem of selfish nodes reporting a *lower* PDR; the goal of this attack is to discourage routes from passing through these selfish participants. However, a source may find alternative paths that may still provide good performance. Zeng et al. [6] propose SLQM to prevent adversaries from reporting inflated PDR values, through the use of nonces. However, the solution does not consider colluding attackers and as we will show later, it can easily be bypassed by colluders. Moreover, the authors do not implement their scheme to sufficiently quantify the impact of the attack in a real setting. [14] studies PDR manipulation in multicast protocols. The proposed method does not handle sybil attackers and is subject to *framing attacks* where attackers can falsely claim that a benign node is malicious. In [15], the authors propose a method to detect the *sinkhole* attack in sensor networks, by comparing reported link qualities between nodes. The sinkhole attack is similar to the attack launched by route magnets, where malicious nodes try to attract routes by faking link qualities. However, specific solutions proposed in [15] do not apply to our setting and have the following limitations:

• [15] assumes that the link qualities are symmetric between two nodes; however as is well known, this is not always the case in wireless networks. We show via real experiments that asymmetric links result in certain issues that render [15] ineffective. In contrast, RoMaD can effectively operate in scenarios with diverse set of link qualities.

• [15] assumes that attackers do not collude i.e., there is

a single independent attacker that launches the attack. In contrast, RoMaD can effectively cope with various forms of collusion.

- Unlike [15], we demonstrate the impact of the attack and the viability of our solution on a real wireless testbed.

Other relevant studies: Newsome et al. [7] propose the use of radio resource testing to challenge sybil suspects by listening on the assigned channels. However, this challenge procedure is limited only to detecting the presence of sybil devices. In contrast, our work has a broader scope since we also ensure trustworthy PDR reporting.

There are efforts on detecting wormhole [16], blackhole, grayhole [4] and jellyfish attacks [17]. Hu et al. [18] propose the use of temporal and geographical leashes to detect the wormhole links. Using temporal leashes requires precise clock synchronization among the nodes. Geographic leashes rely on the physical location of a node, which might not be trivial to compute in adversarial settings. Hu and Evans [19] use directional antennas to detect wormholes; however, the study relies on such specialized hardware, which may not be available in practice. In [20], the authors use graph theory to detect the wormholes. Eriksson et al. [21] investigate the use of link layer timing to prevent wormhole connections between two nodes. Note that unlike in our work, in all of the above studies, malicious devices are considered to be "outsiders", rather than compromised nodes that participate in network operations.

III. Profiling Route Magnets

Route magnets can attract routes using various methods. We profile each method to assess its effectiveness. We do not present detailed experimental results here due to space constraints and simply summarize our main observations. A more extensive set of results can be found in [22].

Threat model: We consider as malicious, the nodes that try to manipulate the route discovery process through either fake PDR reports, remote collusion, or combinations thereof. The name "magnet" is chosen to represent the specific malicious action of "attracting" routes and not to represent the particular attack *on the packets themselves* (e.g., blackhole, jellyfish). These other attacks that target flows are only possible if the attacker is present on the specific route; we address the specific case where the attackers try to cause routes to go through them with the intention of launching additional attacks on information flows. In other words, RoMAD focuses on detecting the magnets rather than the particular attacks on packets.

Throughout the rest of this paper, we assume that magnets are compromised regular nodes (i.e., insiders) in the network. Therefore, in terms of most hardware capabilities, they are no different than any other node. The only difference we envision is the Ethernet (or some other out of band) link used for collusion (explained in detail later).

The attack strategies that we consider adopt the common tactic of advertising fake, high PDR values. However, they differ in terms of the specifics of the attack strategy. In
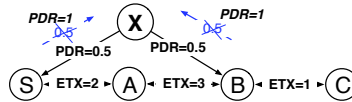

Figure 1. The independent magnet X advertises fake PDR values (benign values are crossed out) in order to attract the route from S to C.

most cases, we consider two colluders. Our solution is applicable to cases with more than two colluders; however, we recognize that a more in-depth investigation is required and will consider this in future work.

Network assumptions: We consider a network that employs public key infrastructure (PKI). Specifically, each node is assumed to have a public-private key pair used for encryption and authentication. The certificates are verified and managed by a certification authority (CA). The message contents encrypted with the public key of a node can only be decrypted using the private key of the particular node.

We did not observe significant throughput overhead due to encrypting the probes since these are typically small packets (100 - 200 bytes). Our testbed consists of Soekris nodes having 266 MHz processors and 256 MB RAM on board. We believe that probe encryption can easily be deployed in today's wireless networks, which have nodes with much faster processors and increasing amounts of available memory.

As mentioned before, RoMaD addresses attackers that manipulate PDR-based routing metrics. Since these metrics utilize periodic probe broadcasts, they are not suitable for networks with energy constraints (e.g., sensor networks). We believe that RoMaD is most applicable to 802.11 networks where PDR-based routing metrics have already been implemented and demonstrated to offer throughput benefits [1], [2]. However, as we show in this paper, these metrics offer a new ground for the attackers to impede the benefits and should be complemented with a detection framework such as RoMaD. We now proceed to describe different types of route magnets and the attack strategies that they use.

A. Independent Route Magnets

Malicious devices in this category advertise high PDR values for the links *from* their neighbors. For example, consider the topology in Fig. 1, where node X is the only magnet. The source node S wishes to setup a route to destination C. In benign conditions, S chooses the path $S - A - B - C$. However, if X reports a very high PDR value (e.g., PDR = 1) for the links $S - X$ and $B - X$ (both links have a PDR = 0.5 in reality), it is easy to verify that S will choose the route $S - X - B - C$ (since it is the route with the lowest aggregate ETX value). With this, X successfully attracts the route from S to C. We call this attack **IRM**, for Independent Route Magnets. Note that although X tries to manipulate S by advertising fake qualities for its attached links, S may still select a route that does not include X. This is because X cannot advertise fake PDR values for links other than its own. Hence it may be possible that the aggregate ETX value for other routes is still lower.
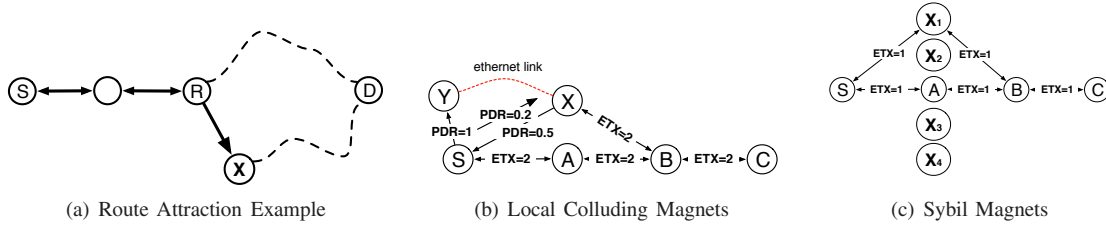
(a) Route Attraction Example (b) Local Colluding Magnets (c) Sybil Magnets

Figure 2. Local colluders can exchange information via Ethernet. Sybil magnets can attract routes without manipulating the PDR.

When is IRM effective? We conduct experiments to assess the ability of independent magnets to attract routes in different topological scenarios. We observe that by attracting routes, even if no other attack is launched in conjunction (like a blackhole attack), there can be up to a 90% throughput reduction compared to benign cases (the experiment is described in [22]). The main reason for this throughput reduction is that the quality of the links comprising the route degrades (as in Fig. 1). In addition, we observe that:

a. IRM is most effective on asymmetric links: Our measurements suggest that IRM is especially effective when applied on links where the PDR on the link $<magnet$ - $neighbor>$ is significantly higher than that in the reverse direction. This is because on such links: (a) the magnet can easily lie about the PDR *from* neighbors by reporting high values, and (b) most of the magnet's reports are actually received by neighbors, due to their good incident links. Note here that this property of magnets renders the solution in [15] ineffective. Since the magnets make the link "look like" symmetric via PDR manipulation, [15] perceives this as a benign operation.

b. The hop distance between a source and its destination affects the potency of IRM: Independent magnets can only manipulate the PDR on their neighbor links and thus their effect is localized. To illustrate, node X is trying to attract the route from S to D and become the relay instead of node R (see Fig. 2(a)). The magnet cannot fake the PDR for the nodes on the path from R to D and similarly from X to D. As one might expect, with shorter paths the probability of attracting the route is higher. In our experiments, we observe that when the source and the destination are separated by many hops, it is harder for the magnet to attract the route. This is because with longer routes, the path cost is the sum of the ETX values of a higher number of links (not under control of the magnet). Since one independent magnet can manipulate at most two links on a path, it typically does not cause node S to pick the path with X. With few hops, however, magnet's ETX manipulation significantly affects the best path decision.

B. Colluding Route Magnets

Simple solutions can prevent IRMs from being effective. As an example, benign nodes could mark the probes with random nonces and require their neighbors to report the actual nonces (or a linear function of such nonces) from the probes that they receive. This makes it difficult for IRMs to lie about the received probes from neighbors. However,

colluding magnets can easily bypass this strategy as we explain in the following.

Local Colluders: If two or more magnets (having a common neighbor) cooperate, they can exchange information about the nonces that they have each deciphered from the common neighbor(s). In other words, if magnets were to *collude*, they could cooperatively report high PDR values. We call these devices "local colluders", and the attack **CRM** (for Colluding Route Magnets).

As an example, consider the topology in Fig. 2(b). Nodes X and Y are two local colluders; S requires neighbors to report the sum of all the nonces that they receive. The link $S - X$ has PDR = 0.2, while the link $S - Y$ has a PDR = 1. For X to join the route from S to C, it has to report PDR = 1 for the link $S - X$. For this, X has to provide the sum of all the nonces included in the probes that have been transmitted by S during the last probing interval. Since the colluder Y has received all the probes (PDR = 1), it can send all the nonces to X; X can then embed this information in probes, thereby fooling S about the quality of the link $S - X$. Recall that colluding magnets utilize a private Ethernet link to exchange information.

Sybil Magnets: Colluding magnets, being insider adversaries, can share their cryptographic keys. With this, magnets can create a large number of identities in a local neighborhood and use them to increase the probability of being selected as relays. Such a magnet can be one physical device with many virtual wireless interfaces, and each interface could be used to transmit probes with different MAC addresses. Upon receiving the probes, legitimate nodes will falsely believe that the fake identities correspond to real neighbors. We call these magnets "sybil magnets" and the attack **SRM**, for Sybil Route Magnets. The number of fake sybil identities that a magnet can create is at most equal to the number of compromised nodes in the network. This is because, in order to create a sybil identity, the magnet needs to have an additional set of predicates (i.e., keys) that are only obtainable by compromising an additional node.

Fig. 2(c) depicts a representative example of SRM. Here, S seeks to find the route to C. A and X both have high quality links to S (ETX = 1). In the benign case, it is equally likely that S selects A *or* X as the next hop. However, if X creates four identities, S observes that it has 5 neighbors all of which have links with $ETX = 1$. With this, X has now increased the probability of being selected on a path to 0.8 (instead of 0.5 in benign settings). Note that, IRM and CRM are not effective in cases where the magnets have

high quality links with their neighbors. In such scenarios, the PDR manipulation does not give any advantage to the magnets since the benign PDR for the links are already quite high. However, in such cases, the magnets can employ SRM to attract paths.

Distant Magnets: Colluding magnets can also mount wormhole-like attacks on link quality aware routing protocols. In traditional wormhole attacks, outside attackers create fake links between two legitimate nodes by replaying the transmissions of one node in the neighborhood of the other. In our threat model, the attackers are insiders; they announce the fake link between themselves and the routes are computed based on the qualities of the links rather than simply the hop counts. Thus with collusion, an effect similar to the wormhole is created albeit in a different way.

There are two strategies that distant colluders can use. First, colluding insider magnets can exchange their keys and use these when sending probes to their neighbors. As an example, let us assume nodes A and B are located in two "distant" parts of the network. If A uses B's credentials while sending probes, A's neighbors will falsely conclude that they are neighbors with B. With this, such victim nodes will be deluded into believing that there exist shortcuts to other valid nodes via the magnet (since it advertises the neighbor list of B). This is likely to increase the chances of magnets attracting a path. Second, two magnets can advertise a fake link between themselves even without exchanging credentials. This can still result in fake shortcuts between legitimate nodes. We call these types of magnets together "distant magnets" and the strategy **DRM** for Distant Route Magnets. As we show later, these attacks are extremely harmful; once the magnets create the fake link, they can attract many routes and can easily hurt performance by dropping packets.

Impact of Colluding Magnets on Performance: Thus far, we have not shown the impact of magnets on performance (e.g., throughput). For the throughput to be affected, the quality of the path should change from that in benign settings (e.g. poorer quality links or longer paths) as in Fig. 1. With colluding magnets, the quality may be similar to that of in benign settings (as in Fig. 2(c)). However, once they attract a path, the magnets can launch other attacks to significantly impact the performance. We conduct comprehensive experiments with magnets in Section V.

IV. Detection Framework

RoMaD consists of two components: namely NRC and MCC. NRC includes nonces in probe packets; it can be used to (a) detect IRMs and (b) compile a list of suspect colluding magnets via simple hypothesis tests. MCC uses multiple channels and transmission power levels to challenge the suspect nodes determined by NRC. Since a single wireless interface can be tuned to one frequency at a time, it is expected that SRMs will not be able to simultaneously respond to challenges on more than one channel. Similarly, as discussed later, CRMs will be greatly limited in their

---

Algorithm 1 NRC operation at B
P is the list of probes received from A

$sum \leftarrow 0$
$vector \leftarrow 0$ {initialize a vector of 0s}
$seq \leftarrow last\ sequence\ number\ used$
for $p \in \mathbb{P}$ **do**
    $sum \leftarrow sum + p.nonce$
    $i \leftarrow p.seq\_no\ mod\ (w/\tau)$
    $vector.i + 1^{th}\ bit \leftarrow 1$ {set $i + 1^{th}$ bit to 1}
end for
$seq + +$
$nonce \leftarrow random\ int$
{store [seq, nonce] and send report}

---

abilities to exchange nonce information. In addition, the use of different power levels help detect DRMs.

A. NRC: Nonce Report Component

With NRC, each node includes a pre-specified linear function (the sum in our implementation) of the nonces that it receives from its neighbors, when it reports the PDR. The recipient verifies the included sum; should the verification fail, the transmitter is flagged as an IRM. To illustrate the semantics of NRC, consider two neighbors A and B. A broadcasts probes once every τ seconds, while B reports the corresponding PDR for the last ω seconds. In every probe, A embeds the following: (i) a nonce value that is randomly picked and (ii) a sequence number that is incremented by one. A also maintains a local copy of the tuple of the sequence number and the nonce associated with each probe.

The recipient, B, maintains the following: (a) a moving sum of the nonces that it receives from A during the last probing window and (b) the tuple of the sequence number and the nonce included in each received probe. Whenever B reports the PDR for the link $A - B$, it includes the sum of the nonces (from the probes sent by A), along with a bit vector indicating the sequence numbers of the received probes. B sets the ith significant bit to 1, if a probe with sequence number $seq \equiv i - 1 \pmod{\omega/\tau}$ is received; if not, the particular bit is set to 0.

A verifies the sum of nonces reported by B in conjunction with B's bit vector. For example, if B reports a vector wherein all the bits are set to 1, then B must have received all the probes sent by A. Node A, in this case, computes an expected sum by adding the nonces in all the locally stored tuples. A then verifies this expected sum against the sum of nonces reported by B. Note that A can learn about the PDR of the link $A - B$ by accounting for the bits set to '1' in the reported bit vector. We present the pseudocode for nodes B and A in Algorithm 1 and 2, respectively.

Despite the use of nonces, IRMs can still evade detection by NRC. One strategy is to overhear the probe reports sent by legitimate neighbors to infer the missing nonces. In order to address this, we leverage PKI. Specifically, each node is required to sign the reports with its private key and encrypt the signed report using the destination node's public key. With this, magnets will not be able to decode the sum of nonces contained in overheard probes, since they do not have

Algorithm 2 NRC operation at A
p is the probe (report) received from B

$num\_rx \leftarrow 0, expectedSum \leftarrow 0$
$i \leftarrow 1$
for $i \leq p.vector.length;\ incr\ i$ **do**
  if $vector.i^{th}\ bit = 1$ **then**
    $nnc \leftarrow assoc.nonce$
    $\{assoc\ has\ seq\_no \equiv i - 1\ (\mathrm{mod}\ w/\tau)\}$
    $expectedSum \leftarrow expectedSum + nnc$
    $num\_rx + +$
  end if
end for
if $expectedSum \neq claimedSum$ **then**
  B is misbehaving
else
  $pdr\_to\_B \leftarrow num\_rx * \tau/w$
end if

L	List of potential magnets
p_f and p_r	Forward PDR and reverse PDR of a link
p_f^{ad}	Forward PDR advertised by a node
p_f^{act}	Actual forward PDR for a link found using MCC
α	MCC tolerance parameter
challenger	The node initiating MCC challenge
responder	The node being challenged by MCC

Table I
TERMS AND NOTATIONS

the private key of the destination. In addition to detecting IRMs, NRC is also leveraged for hypothesis tests with MCC to detect colluding magnets. We discuss this next.

B. MCC: Multiband Challenge Component

The primary goal of MCC is the detection of colluding magnets. MCC leverages multiple channels and transmission power levels to challenge suspect nodes. The key insight behind MCC is that a wireless interface cannot simultaneously operate on two different channels. We enlist the definitions that we use in Table I. In what follows, we present the operations of MCC.

The 'high forward PDR' hypothesis test: The challenger identifies the nodes that it wishes to challenge using a hypothesis test. If a node were to challenge all of its neighbors, the overhead due to MCC would be excessive. Thus, it is important to first identify a subset of the neighbors L who could be potential magnets. To construct this subset, NRC computes the average forward PDR, p_f^{avg}, over all links of a node (after IRMs are filtered out). If a neighbor reports a forward PDR much higher than that of the other nodes, then NRC includes it in L. Specifically, these are neighbors that report a forward PDR $p_f^{ad} \geq p_f^{avg} + \vartheta * var$, where ϑ is a threshold parameter and var is the variance of p_f values. The hypothesis test is based on the fact that magnets typically advertise high PDR to increase the probability of being selected as relays.

The above hypothesis alone cannot identify potential SRMs since they can operate without inflating the PDR (see Section III). However, since sybil identities belong to the same physical node, these identities will report the same

Algorithm 3 MCC operation

---
while all channels are challenged with less than m packets **do**
  $channel \leftarrow random\ channel$
  $node \leftarrow channel.assigned\_node$
  $reply \leftarrow challenge(node, channel)$
  if $reply$ **then**
    $node.success\_count + +$
  end if
end while
for $node \in L$ **do**
  compute p_f^{act} for $node$ as $p_f^{act} = k/(m \cdot p_r)$
  if $p_f^{ad} - p_f^{act} \geq \alpha$ **then**
    node is misbehaving
  end if
end for

PDR statistics. Therefore, coupled with the forward PDR hypothesis, NRC also monitors if there is more than one node reporting the same set of nonces. If there are such nodes, NRC includes them in L as well.

The challenge procedure: The challenger assigns a different channel to each node in L and broadcasts this information on the default channel. The nodes that receive this announcement configure their radios to their assigned channels and wait for challenge packets. The challenger then randomly selects a channel and transmits one or more challenge packets. Each such packet has an associated random nonce (we leverage NRC). If the responder receives the packet successfully, it is expected to report the nonce back to the challenger. Using this procedure, at random instants, the challenger sends m packets to each node in L and, keeps track of the number of successful replies. It then calculates the difference between p_f^{ad} and p_f^{act} (see Table I) for each responder. If $p_f^{ad} - p_f^{act} < \alpha$, the particular responder is considered to be legitimate; otherwise it is flagged as a magnet. The challenger computes p_f^{act} for each responder as $p_f^{act} = k/(m \cdot p_r)$ where k is the number of successfully received responses (out of m). We present the operation of MCC in Algorithm 3.

Discarding sybil identities: As discussed in Section III, SRMs report a high p_f^{ad} value by using multiple virtual identities. Hence, NRC includes all the identities of a sybil node in list L. MCC then assigns a different channel to each of these identities. The fact that nodes are equipped with a single radio interface prevents SRMs from listening on more than one channel at a time [3]. Hence, if a sybil magnet stays on one of its assigned channels, the challenger will detect and discard the other fake identities since these will not be able to reply to the challenge packets. Note that with MCC, the challenger cannot actually detect the physical sybil magnet; however, it can discard the fake identities, thereby decreasing the success of SRM to a great extent. An alternate strategy for the magnet could be to "hop" between the assigned channels and try to be on the same channel as the challenger. However, due to the random channel

---
[3]The approach can easily be extended to cases where nodes have multiple wireless interfaces. A node will be required to simultaneously respond on more than one channel.

selection, the probability of tuning to the same channel as the challenger is $1/c$, where c is the number of virtual identities used by the magnet. Clearly, the probability decreases with larger number of sybil identities.

Detecting local colluders: CRMs exchange nonce information using their Ethernet link. If the colluders report a p_f^{ad} greater than the threshold in the hypothesis test, they are included in list L. During the MCC challenge, one strategy for the colluders could be to reply to challenges addressed to their colluding partner. This can especially be a feasible strategy when one of the colluders maintain a high-PDR link to the challenger node (e.g., Fig. 2(b)). However, the challenge on different channels makes it difficult for a suspect to respond on behalf of its colluder. Since a node can only be on one channel at a time, it cannot respond to the challenges issued to its colluding partner if it stays on its own channel. Conversely, if it chooses to be on the channel assigned to the colluding partner, it misses the challenges on its own channel. Thus, the assignment of different channels discourages colluders from hopping between channels.

Detecting distant magnets: The challenge phase for detecting DRMs is slightly different. It uses a different hypothesis test and employs multiple transmission powers.

The 'topological inconsistency' hypothesis test: Maintaining the average forward PDR of the links is not sufficient to hypothesize about attacks launched by DRMs. These magnets can easily inject "shortcut" links without being classified as suspects. To cope with these magnets, we require each node to encrypt the nonce information (described earlier), but send out the node IDs of its neighbors without encryption. This connectivity information is also essential for the routing functionality with ETX [1].

Each node monitors the reports from its neighbors to determine its two-hop neighborhood. If a subset of nodes in the two-hop neighborhood are reachable *through only one* neighbor, that immediate neighbor is suspected to be a distant magnet and included in L. We point out that even with this simple test, RoMaD captures the topological anomalies imposed by distant magnets and successfully detects the adversaries, as shown in Section V.

The power challenge procedure: To detect DRMs, MCC assigns channels in the following manner. A channel is assigned to one of the suspects in L and the remaining channels are assigned to the two-hop neighbors that are only reachable through that suspect node. MCC then uses an appropriately chosen higher transmission power (assuming that the default power is less than the maximum) in an attempt to reach the two-hop neighbors and verify their existence via challenge packets. We explain this with an example in Fig. 3. Assume that nodes X and Y advertise a fake link between themselves. Let us also assume that Y obtains X's PDR statistics and advertises them. When A receives probes from Y, it observes that X is a two-hop neighbor that is reachable only through Y. Thus, the topology inconsistency test tags Y as a suspect and MCC assigns channels to X and Y. The challenges are now issued with a higher power to see if X is indeed a legitimate two-
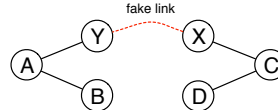

Figure 3. Example topology for power challenge.

hop neighbor. If A does not receive enough replies from X, it considers Y to be malicious.

The power challenge does not target the two-hop neighbor that is reachable through the suspect node. Instead, it determines whether the one-hop suspect is a magnet. If the higher power transmissions do not reach the two-hop distance and the challenge packets fail for the two-hop neighbor, the one-hop suspect is categorized as a distant magnet. This property discourages DRMs; it forces them to advertise fake links between relatively nearby regions of the network. This limits the reach of the fake link and thus, reduces the chances that a magnet is chosen as a relay.

One could argue that since nodes are flagged as malicious by other peers (challengers), *framing attacks* are possible, wherein malicious nodes could flag honest nodes as magnets. However, note that RoMaD is not affected by such attacks. In particular, each node performs its own challenge of a suspect and decides whether the latter is a magnet or not. In other words, only the specific challenger avoids paths through a node considered as a magnet. Thus, there is no opportunity for the attackers to *blame* legitimate nodes.

V. Evaluating our Framework

Our testbed (see [5] for details) consists of 42 Soekris net5501 nodes. For each experiment on the testbed, we initiate a series of benign flows and deploy the malicious devices. We simultaneously activate 5 different source-destination pairs. We measure the aggregate end-to-end throughput under the following settings: (a) when every node is benign, with and without RoMaD, (b) when the magnets are active without RoMaD, and (c) RoMaD is performing against the magnets. We repeat experiments with each of these three settings 20 times with randomly picked flows and magnets. We collect measurements from the 5 flows and calculate the total throughput (averaged over 20 runs). All nodes set their transmission powers to the default value of 15 dBm. We mainly use saturated UDP traffic with 1500 byte packets. We use DSDV (with ETX) as the routing protocol and set $\tau = 1$ sec and $\omega = 10$ sec. We use the *iperf* tool for generating traffic. Since we observe that the results with and without RoMaD in benign settings are almost identical (given the low overhead as discussed later), for clarity we only show the results for the benign case without RoMaD. Next, we evaluate RoMaD against each attack.

1. Performance with independent magnets: We activate 5 flows and one independent magnet simultaneously in each experiment and initiate 300 sec. of saturated UDP traffic. The magnet drops all packets after attracting the routes (i.e., blackhole attack). The cumulative distribution function (CDF) of the throughput samples is presented in Fig. 4. Without RoMaD, nearly 10% of the samples
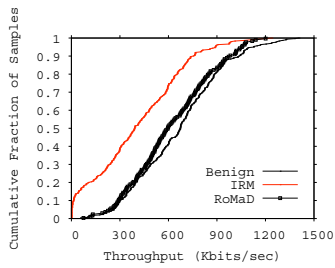
Figure 4. RoMaD increases the median throughput by 40% under IRM.
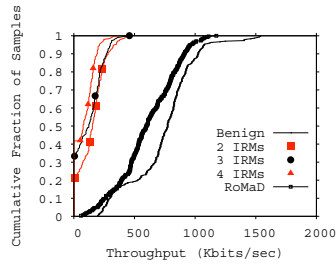

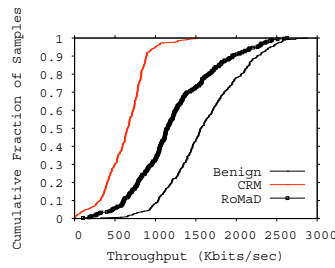Figure 5. RoMaD can effectively combat increasing number of magnets.

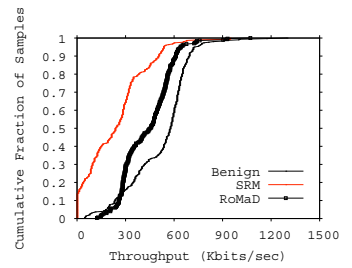Figure 6. RoMaD increases the median throughput by 76% under CRM.

Figure 7. RoMaD increases the median throughput by 92% under SRM.
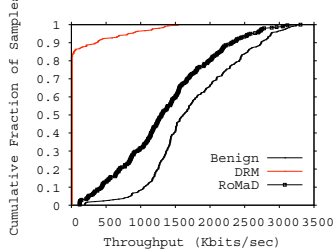

Figure 8. RoMaD increases the maximum achievable throughput by 380% for 90% of the samples, under DRM.
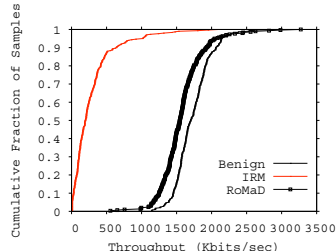

Figure 9. In a bottleneck setting, RoMaD increases the median throughput by $\approx$ 680% when under attack.
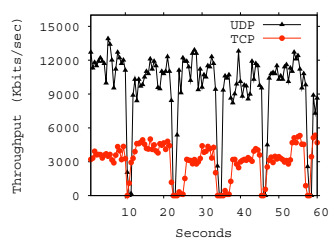

Figure 10. After a channel switch, TCP and UDP can reach the same throughput as before the switch.
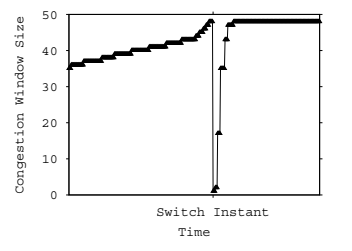

Figure 11. After a channel switch, TCP enters the slow start phase.

correspond to a zero throughput (with a median throughput of 390 Kbits/sec). RoMaD successfully detects the magnet and restores the throughput to a large extent. The median throughput achieved in benign settings is 672 Kbits/sec; RoMaD achieves a median throughput of 547 Kbits/sec. RoMaD increases the overhead in terms of increased probe packet size and the MCC challenge. However, the throughput improvements outweigh this overhead.

Next, we perform a similar set of experiments with different numbers of IRMs. We present the CDF of the throughput in Fig. 5. We observe that without RoMaD, increasing the number of magnets results in a higher percentage of dropped packets due to the increased chance of attracting the routes. RoMaD can effectively combat increased numbers of IRMs. In benign settings, the median aggregate throughput is 770 Kbits/sec; RoMaD restores the median throughput to 600 Kbits/sec when under attack.

2. Performance with colluding magnets: We activate 5 flows and a pair of local colluding magnets. The colluders exchange the nonces that they receive and report a joint fake PDR value for common neighbors. The magnets drop all packets after attracting the routes. The flows initiate 300 sec. of saturated UDP traffic. We observe from Fig. 6 that RoMaD is effective against local colluders and improves throughput by avoiding the routes via the magnets. However, the throughput cannot be completely restored due to the overhead with MCC.

We perform another set of experiments with a sybil magnet using two different virtual identities. Further, the magnet launches a blackhole attack. To evade detection during the challenge, the magnet switches between its assigned channels at random times. Fig. 7 shows that without RoMaD, a sybil magnet drops nearly 20% of the packets. However, the MCC challenge effectively detects the fake identities, eliminates them from the paths and

thereby restores the throughput to a large extent. The median throughput achieved in benign settings is 576 Kbits/sec. RoMaD restores the median throughput from 235 to 456 Kbits/sec in adverse settings.

Next, we activate 5 flows and two pairs of DRMs. The magnets advertise a fake link of ETX = 1 between each other and implement the blackhole attack. For this set of experiments, the nodes operate at 15 dBm and increase the power to 19 dBm (max. supported by our cards) during the MCC power challenge. Fig. 8 presents CDF of the throughput. Without RoMaD, the magnets attract up to 84% of the packets and cause zero throughput. RoMaD is effective and the power challenge phase eliminates the shortcut links. In benign settings, the median throughput is 1611 Kbits/sec whereas RoMaD achieves a median throughput of 1300 Kbits/sec in attack scenarios.

3. Performance with a bottleneck scenario: When the magnets attract routes, they become bottlenecks for the flows. Thus, magnets hurt performance even by *simply* attracting victim flows. To examine this more carefully, we choose three source-destination pairs all active at the same time; the flows use disjoint routes in benign settings and can coexist without imposing much interference on each other. We introduce an IRM, which attracts the three flows and becomes a bottleneck. Note that the magnet does not implement the blackhole attack. We present the throughput for 300 sec. of TCP traffic in Fig. 9. Without RoMaD, the congestion at the magnet reduces the TCP throughput by 88%. RoMaD successfully detects the magnet and restores the throughputs.

In the experiments so far, we evaluate the performance of RoMaD over long flow durations. The throughput with RoMaD includes the *macroscopic* overhead of the MCC challenge over a long duration. We observe that RoMaD's overhead is lower when it combats IRM (Fig. 4), than in

cases with colluding magnets. This is because IRMs are immediately detected via the nonces in NRC and thus the MCC challenge is not activated frequently. Next, we want to evaluate the *microscopic* impact of the MCC challenge.

4. Evaluating MCC overhead: As discussed in Section IV, the nodes switch channels to respond to challenges and return to the default channel when the challenge is over. We evaluate the temporary interruption of traffic at a relay node due to channel switching. The relay node is configured to switch to a different channel, stay there for 1 sec. and return to the default channel. This cycle is repeated every 10 sec. We deliberately choose these durations to better visualize the impact of MCC (the reported penalties are significantly lower in practice). Fig. 10 presents the time series of the throughput both for a UDP and a TCP flow. We observe that UDP can quickly regain the throughput that was achieved before the switch. The recovery time is slightly longer with TCP due to timeouts and congestion control. However, as shown in Fig. 11, it can quickly ramp up the congestion window to the value before the switch.

5. Impact of the overhead of RoMaD: RoMaD incurs overhead when deployed in a network where every node is benign. As explained in Section IV, NRC monitors the PDR reports and builds a list of suspect nodes. During the PDR report monitoring, each node has to periodically sign and verify the reports which has an overhead on CPU usage. We found that these operations however, only had an insignificant impact on the throughput.

When the nodes are challenged, there is an overhead due to temporal loss of connectivity at relay nodes (evaluated above). To keep the frequency of challenges minimal, RoMaD only challenges nodes that report much higher than average PDR values (as discussed earlier). We observe in our experiments with benign nodes, that RoMaD invokes MCC challenge very rarely. This is because in such settings, each node truthfully reports the actual PDR and these do not exhibit a high deviation from the average. Therefore, the performance with and without RoMaD in benign settings is very similar. Thus, for purposes of clarity we do not explicitly plot these results.

6. Evaluating false positives and negatives: We observe rare cases where MCC wrongly tags a neighbor as malicious. This is possible when the links to the neighbor are of poor quality. If the quality of the link to the challenger fluctuates, the benign node may fail the challenge. Fig. 12 depicts the false positive rate with RoMaD (i.e., the ratio of suspects that were erroneously tagged as malicious out of the total number of suspects) for a set of ϑ and α values. We observe that: (a) For a given ϑ, the false positive rate decreases with increasing α. This is due to the increase in the allowed difference between the measured and reported PDR values for the suspect nodes. (b) As ϑ increases, a smaller set of nodes are included in the suspect list and therefore, the challenge produces fewer false positives. For appropriately picked α and ϑ, RoMaD can achieve false positive probabilities as low as 3.7%.
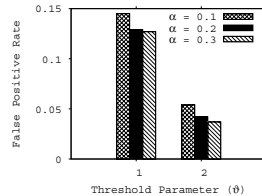
False positives with the power challenge phase can occur

Figure 12. False positive rates of MCC frequency challenge for varying α and ϑ.
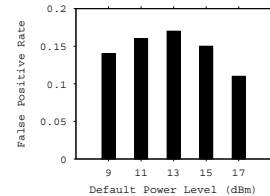
Figure 13. False positive rates of MCC power challenge for varying default power levels.

if some nodes in the two-hop neighborhood are reachable via only one benign neighbor. In Fig. 13, we show the false positive rates for different default power levels (increased to 19 dBm during the power challenge). For 9 dBm, the nodes are connected with very short range links; therefore most of the suspected two-hop neighbors are validated with a 19 dBm power level during the challenge phase. As the default power level increases and the topology gets more connected, there is an increase in suspicious cases which in turn leads to increased false positives. However, once the connectivity increases beyond a certain degree, two-hop neighbors become legitimate one-hop neighbors and the false positives decrease. Note that the current implementation is based on a rather "aggressive" approach towards identifying magnets. Thus we observe very low false negatives at the expense of moderate false positives (10% to 17%); fake links are detected with $> 95\%$ accuracy throughout our experiments.

A false positive simply eliminates a benign node from being chosen on a path; thus the overall impact on throughput is typically not significant. We would like to point out that we observe very few false negatives (only 2 among all our experiments) in the MCC challenge phase. In these cases, both colluding magnets maintain high quality links with their neighbors and can therefore handle the challenge packets. However, note that in such settings, IRM and CRM strategies do not provide an advantage to the magnets since the link quality cannot be manipulated to be better. We find that the MCC challenge phase is extremely effective against sybil magnets (recall Fig. 7); we do not observe a case where they are able to evade detection.

Limitations of our work: With RoMaD, there is no coordination between legitimate devices with regards to which node will perform a challenge. For example, it is possible that two or more nodes decide to perform MCC on certain suspects at the same time. Since this requires the assignment of channels to suspects, a distributed decision should be made on who will perform the challenge and for which set of nodes. We plan to design such a protocol in our future work.

We recognize that MCC introduces a new vulnerability; an attacker can initiate frequent challenges to its neighbors. This causes a neighbor to consistently respond to these challenges thereby suffering a disruption in throughput for prolonged periods. This problem can be alleviated by imposing a maximum rate at which the MCC challenge can be invoked. We will consider this possibility and examine the trade-offs between fast detection and opening the door for the new attack in future work.

VI. Discussions

Combining different attacks: In our evaluations, we consider each type of attack in isolation. We do so to capture the specific impact of each type of attack. We recognize that an attacker could launch these attacks in conjunction to be more effective. Our approach will still hold in such cases and we believe can detect attackers. This is because NRC identifies potential attackers based on the deviation from the average PDR in the neighborhood (no assumption is made on the attack type). Thus, even if an adversary combines attacks of different types, it is likely to be flagged by NRC and subsequently challenged by MCC. We will experimentally validate this in further experiments in future work.

Efficacy of the Power Challenge: We wish to point out that the power challenge is only issued on nodes that are flagged by NRC. Thus, even though the false positive rate of the power challenge is about 10 - 17%, the overall false positive rate (considering NRC) is much lower. We do recognize that the efficacy of the power challenge is topology dependent; while it effectively detects attackers in the topologies of our testbed, the possibilities considered are not clearly exhaustive. We will conduct a more extensive study of the power challenge in future work. Note also that false positives typically do not result in significant throughput penalties. Even if a benign node is falsely flagged as a magnet, there maybe other candidate relays with similar link qualities. On the other hand, false negatives have a higher impact on throughput since a magnet (not detected via the challenge) can perform packet manipulation attacks. RoMaD leverages this in its design; the "aggressive" power challenge results in very few false negatives while allowing a moderate degree of false positives. Thus, we expect that RoMaD will not result in significant throughput loss when it incurs false positives.

VII. Conclusions

We profile and address an attack on link quality based routing protocols: by advertising fake PDR values, adversaries (called *magnets*) can attract routes and affect end-to-end throughput. We consider independent attacks and attacks involving collusion. Magnets may collude locally, create sybil identities or establish distant collusion. We design and implement a framework, RoMaD, that can be used as a unified solution against the various flavors of the attack. RoMaD helps create adversary-free routes proactively through (a) tagging PDR probes with nonces, and (b) challenging suspect nodes on different channels and power levels. Our extensive experiments on a 42 node testbed demonstrate that RoMaD can detect the magnets with high efficacy and provide significant throughput benefits.

References

[1] D. S. J. De Couto et al. A High Throughput Path Metric for MultiHop Wireless Routing. In *ACM MobiCom*, 2003.

[2] R. Draves et al. Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks. In *ACM MobiCom*, 2004.

[3] F. Wu et al. Incentive-Compatible Opportunistic Routing for Wireless Networks. In *ACM MobiCom*, 2008.

[4] P. Agrawal et al. Cooperative Black and Gray Hole Attacks in Mobile Ad Hoc Networks. In *ICUIMC*, 2008.

[5] UCR Wireless Testbed. http://networks.cs.ucr.edu/testbed.

[6] K. Zeng et al. Towards Secure Link Quality Measurement in Multihop Wireless Networks. In *IEEE GLOBECOM*, 2008.

[7] J. Newsome et al. The Sybil Attack in Sensor Networks: Analysis and Defenses. In *IPSN*, 2004.

[8] C. E. Perkins et al. Highly Dynamic Destination Sequenced Distance Vector Routing (DSDV) for Mobile Computers. In *ACM SIGCOMM*, 1994.

[9] M. G. Zapata et al. Securing Ad hoc Routing Protocols. In *ACM WiSe*, 2002.

[10] Y. C. Hu et al. Ariadne: a Secure On-demand Routing Protocol for Ad Hoc Networks. In *ACM MobiCom*, 2002.

[11] Y. C. Hu et al. SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks. In *Ad Hoc Networks, 1(1), 175 - 192*, 2003.

[12] E. M. Royer et al. An implementation study of the AODV routing protocol. In *IEEE WCNC*, 2000.

[13] D. B. Johnson et al. DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks. In *Ad Hoc Networking, Ch. 5, pp. 139-172*, 2001.

[14] J. Dong et al. On the Pitfalls of Using High-Throughput Multicast Metrics in Adversarial Wireless Mesh Networks. In *IEEE SECON*, 2008.

[15] I. Krontiris et al. Launching a sinkhole attack in wireless sensor networks; the intruder side. In *IEEE WIMOB*, 2008.

[16] R. Maheshwari et al. Detecting Wormhole Attacks in Wireless Networks Using Connectivity Information. In *INFOCOM*, 2007.

[17] B. Awerbuch et al. An On-demand Secure Routing Protocol Resilient to Byzantine Failures. In *ACM WiSe*, 2002.

[18] Y. C. Hu et al. Packet Leashes: a Defense Against Wormhole Attacks. In *IEEE INFOCOM*, 2003.

[19] L. Hu et al. Using Directional Antennas to Prevent Wormhole Attacks. In *NDSS Symposium*, 2004.

[20] R. Poovendran et al. A Graph Theoretic Framework for Preventing the Wormhole Attack in Wireless Ad Hoc Networks. In *ACM Journal on Wireless Networks*, 2007.

[21] J. Eriksson et al. TrueLink: A Practical Countermeasure to the Wormhole Attack in Wireless Networks. In *IEEE ICNP*, 2006.

[22] RoMaD Technical Report. http://www.cs.ucr.edu/~marslan/romad-tech.pdf.