Wireless Netw (2007) 13:27–59
DOI 10.1007/s11276-006-3723-x

A graph theoretic framework for preventing the wormhole attack in wireless ad hoc networks

Radha Poovendran · Loukas Lazos

Published online: 8 May 2006

Abstract Wireless ad hoc networks are envisioned to be randomly deployed in versatile and potentially hostile environments. Hence, providing secure and uninterrupted communication between the un-tethered network nodes becomes a critical problem. In this paper, we investigate the *wormhole attack* in wireless ad hoc networks, an attack that can disrupt vital network functions such as routing. In the wormhole attack, the adversary establishes a low-latency unidirectional or bi-directional link, such as a wired or long-range wireless link, between two points in the network that are not within communication range of each other. The attacker then records one or more messages at one end of the link, tunnels them via the link to the other end, and replays them into the network in a timely manner. The wormhole attack is easily implemented and particularly challenging to detect, since it does not require breach of the authenticity and confidentiality of communication, or the compromise of any host. We present a graph theoretic framework for modeling wormhole links and derive the necessary and sufficient conditions for detecting and defending against wormhole attacks. Based on our framework, we show that any candidate solution preventing wormholes should construct a communication graph that is a subgraph of the geometric graph defined by the radio range of the network nodes. Making use of our framework, we propose a cryptographic mechanism based on *local broadcast keys* in order to prevent wormholes. Our solution does not need time synchronization or time measurement, requires only a small fraction of the nodes to know their location, and is decentralized. Hence, it is suitable for networks with the most stringent constraints such as sensor networks. Finally, we believe our work is the first to provide an analytical evaluation in terms of probabilities of the extent to which a method prevents wormholes.

Keywords Wormhole attack · Security · Wireless ad hoc networks · Geometric random graphs

R. Poovendran (✉)· L. Lazos
Network Security Lab, Department of EE, University of Washington, Seattle, WA
e-mail: {radha, l_lazos}@ee.washington.edu

1. Introduction

Networking a large number of wireless devices in ad hoc mode will facilitate a wealth of applications not feasible under the conventional base station-to-network node communication model. The absence of infrastructure and the low-cost, on demand deployment makes ad hoc networks ideal candidate solutions for civilian applications such as disaster relief and emergency rescue operations, patient monitoring, and environmental control, as well as military applications such as target identification and tracking, and surveillance networks. On the other hand, an infrastructureless network has to rely on the collaboration among network nodes in implementing most, if not all, network operations. Moreover, due to limited resources of the wireless devices, algorithms and protocols are designed and implemented to allow distributed collaborative communication and computing involving multiple nodes. For example, two nodes that are not within the direct communication range will have to rely on intermediate nodes to exchange messages, thus forming multihop networks.

To implement distributed algorithms and coordinate the cooperation among network nodes, a number of control messages need to be exchanged in every local neighborhood. For example, to deliver protocol status updates, nodes broadcast their up-to-date information. In addition, the inherent broadcast nature of the wireless medium significantly

reduces the energy expenditure for sending an identical message from a single sender to multiple receivers within the same neighborhood. Hence, broadcasting is an efficient and frequent operation in many network functions. However, a wireless ad hoc network may be deployed in hostile environments, where network nodes operate un-tethered. Moreover, the wireless medium exposes any message transmission to a receiver located within the communication range. Hence, in a wireless environment, it is critical to secure any broadcast transmission from a node to its immediate neighbors. A node receiving a broadcast transmission must verify that (a) the message has not been altered in transit (integrity), (b) it originates from a valid and identifiable network source (authenticity), (c) the message is not a replay of an old transmission (freshness) and that, (d) in case of a local broadcast intended only for immediate neighbors, that the source lies within the receiving node's communication range.

Recently, it has become evident that verification of the integrity, authenticity and freshness of a message via cryptographic methods, is not sufficient to conclude that a local broadcast message originated from a one-hop (immediate) neighbor of the receiving node [20,34,46]. In this paper, we investigate a specific type of attack, known as the *wormhole attack* [20,34,46]. Such attacks are relatively easy to mount, while being difficult to detect and prevent. In a wormhole attack, an adversary records information at one point of the network (origin point), tunnels it to another point of the network via a low-latency link (destination point), and injects the information back into the network. Since in the wormhole attack the adversary replays recorded messages, it can be launched without compromising any network node, or the integrity and authenticity of the communication, and hence, the success of the attack is independent of the strength of the cryptographic method used to protect the communication. In addition, the lack of communication compromise makes this type of attack "invisible" to the upper network layers [20]. As a consequence, using a wormhole attack, an adversary can lead two nodes located more than one hop away into believing that they are within communication range and into exchanging information as if they were immediate neighbors.

Several approaches have been presented for defending against the wormhole attack [6,19–21,46,47]. The solutions proposed attempt to bound the distance that any message can travel [20] or securely discover the set of one-hop neighbors [6,19,21,46,47]. In this paper, we show that any defense mechanism against the wormhole attack can be interpreted by a graph theoretic framework. We make the following contributions.

Our contributions: We present a graph theoretic framework for modeling of the wormhole attack and state the necessary and sufficient conditions for any candidate solution to prevent such an attack. We show that any previously proposed methods [6,19–21,46,47] or future solutions have to satisfy our conditions in order to prevent wormholes. In addition, we propose a cryptographic mechanism based on keys only known within each neighborhood, which we call local broadcast keys (LBKs), in order to secure the network from wormhole attacks and show that our solution satisfies the conditions of the graph theoretic framework. We present a centralized method for establishing LBKs, when the location of all the nodes is known to a central authority (base station). Furthermore, we propose a decentralized mechanism for LBK establishment that defends against wormholes with a probability very close to unity. Based on *Spatial Statistics* theory [11], we provide an analytical evaluation of the level of security achieved by our scheme to support our claims.

Compared to previously proposed methods [6,20,21], our solution does not require any time synchronization or highly accurate clocks. In addition, our method requires only a small fraction of the network nodes to know their location. Finally, our approach is based on symmetric cryptography rather than expensive asymmetric cryptography and hence is computationally efficient, while it requires each node to broadcast only a small number of messages thus having a small communication overhead. Due to its efficiency, our method is applicable to ad hoc networks with very stringent resource constraints, such as wireless sensor networks.

In Section 2, we describe the wormhole attack and present its graph theoretic formulation. In Section 3, we state our network model assumptions. Section 4 presents the idea of LBK's and the mechanisms to establish them. In Section 5, we describe how to secure the broadcasting of the LBKs. In Section 6, we present the performance evaluation of our algorithm. Section 7 presents related work, and Section 8 presents our discussion. In Section 9, we present our conclusions.

2. Problem statement

In this section, we present the wormhole attack model and illustrate how a wormhole attack can significantly impact the performance of network protocols, such as routing, and applications of wireless ad hoc networks, such as monitoring. We then abstract the problem using graph theory and provide the necessary and sufficient conditions to prevent the wormhole attack. Throughout the rest of the paper, we will use the terms wormhole attack and wormhole problem interchangeably to refer to a network with wormhole links.

2.1. Wormhole attack model

To launch a wormhole attack, an adversary initially establishes a low-latency link between two points in the network. We will refer to the attacker's link as *wormhole link* or simply

wormhole. Once the wormhole link is established, the attacker eavesdrops on messages at one end of the link, referred to as the *origin point*, tunnels them through the wormhole link, and replays them at the other end of the link, referred to as the *destination point*.

If the distance separation between the origin point and destination point is longer than the communication range of the nodes, any node at the origin point will rely on multi-hop paths to communicate with nodes at the destination point. Hence, the attacker can use the low-latency link to re-broadcast recorded packets at the destination point faster than they would normally arrive via the multi-hop route. A low-latency link can be realized with a wired connection, an optic connection, a long-range, out-of-band wireless directional transmission, or even a multi-hop combination of any of the aforementioned types of connections, as long as the latency in the wormhole path is less than or equal to the latency in the legitimate multi-hop path.

In a wormhole attack, the devices and wormhole links deployed by the adversary do not become part of the network. The devices used to mount the attack do not need to hold any valid network Ids and, hence, the adversary does not need to compromise any cryptographic quantities or network nodes in order to perform the attack. Any key used by valid network nodes for encryption remains secret, and the integrity and authenticity of the replayed messages is preserved. The lack of need to compromise any valid network entity makes the wormhole attack "invisible" to the upper layers of the network [20]. Furthermore, the adversary need not allocate computational resources for compromising the communications, thus making the wormhole attack very easy to implement.

The assumption of not compromising the network communications is a reasonable one since if the adversary were to gain access to cryptographic keys used in the network, it would have no need to record messages at one part of the network, tunnel them via a direct link, and replay them to some other part of the network. Instead, the adversary can use the compromised keys to fabricate any message and inject it into the network as legitimate. Using compromised keys to *impersonate* a valid node, and fabricate and inject bogus messages into the network, known as the Sybil attack [13,33], is overall a different problem than the wormhole attack and is not addressed in this paper. We present our reasoning on assuming non-compromise of cryptographic keys and nodes in our discussion in Section 8.

Finally, in our wormhole attack model, we assume that the adversary does not launch any Denial-of-Service (DoS) attacks against network entities. The goal of the adversary is to remain undetected and, hence, DoS attacks, such as jamming of the communication medium as well as battery exhaustion attacks, are not performed by an adversary mounting a wormhole attack. We now present examples on the impact of a wormhole attack on network protocols.

2.2. Wormhole threat against network protocols

Wormhole attack against routing protocols: Ad hoc network routing protocols can be classified into *periodic* protocols [4,32,36] and *on-demand* protocols [22,37]. In periodic protocols, every node is aware of the routing path towards any destination at any given time and periodically exchanges information with its neighbors to maintain the best network routes. In on-demand protocols, a routing path is discovered only when a node wants to send messages to some destination. A wormhole attack can affect both categories of routing protocols in the following ways.

Periodic protocols: Periodic protocols are based on the distance vector routing algorithm, which was initially proposed for wired networks [2]. In distance vector routing, each node stores a routing table that contains for each possible destination the associated routing cost, usually in number of hops, and the corresponding next hop towards that destination. Periodically, or when a route change occurs, each node broadcasts its routing table in order to inform its neighbors about possible route changes. Every node that receives a route update adjusts its own routing table based on the broadcast received from the neighboring nodes.

As an example, consider Fig. 1(a) which shows an ad hoc network of 13 nodes. In Fig. 1(a), a node s_i is connected to a node s_j if the distance between them is less than the communication range r. Consider an attacker establishing a wormhole link between nodes s_9 and s_2, using a low-latency link. When node s_9 broadcasts its routing table, node s_2 will hear the broadcast via the wormhole and assume it is one hop away from s_9. Then, s_2 will update its table entries for node s_9, reachable via one hop, nodes $\{s_8, s_{10}, s_{11}, s_{12}\}$, reachable via two hops, and broadcast its own routing table. Similarly, the neighbors of s_2 will adjust their own routing tables. Note that nodes $\{s_1, s_3, s_4, s_5, s_7\}$ now route via s_2 to reach any of the nodes $\{s_9, s_{10}, s_{11}, s_{12}\}$.

On-demand protocols: A wormhole attack against on-demand routing protocols can result in similar false route establishment as in the case of periodic protocols. Consider the route discovery mechanism employed in DSR [22] and AODV [37] protocols. A node *A* initiates a route discovery to node *B* by broadcasting a *route request* message. All nodes that hear the *route request* message will re-broadcast the request until the destination *B* has been discovered. Once the destination *B* is reached, node *B* will respond with a *route reply* message. The *route reply* message will follow a similar route discovery procedure, if the path from *B* to *A* has not been previously discovered. If an attacker mounts a wormhole link between the *route request* initiator *A* and the destination *B*,

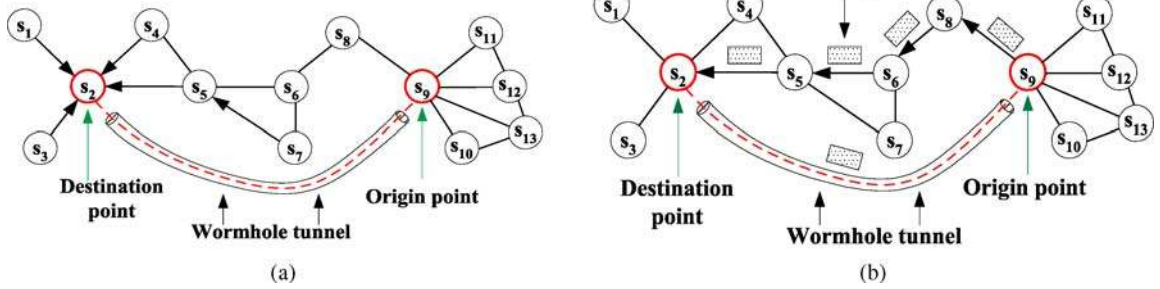


Fig. 1 (a) Wormhole attack on a distance vector-based routing protocol. An adversary establishes a wormhole link between nodes s_9 and s_2, using a low-latency link. When node s_9 broadcasts its routing table, node s_2 hears the broadcast via the wormhole and assumes is one hop away from s_9. Then, s_2 updates its table entries for node s_9, now reachable via one hop, and nodes $\{s_8, s_{10}, s_{11}, s_{12}\}$, now reachable via two hops, and broadcasts its own routing table. Similarly, the neighbors of s_2 adjust their own routing tables. Nodes $\{s_1, s_3, s_4, s_5, s_7\}$ now route via s_2 to reach any of the nodes $\{s_9, s_{10}, s_{11}, s_{12}\}$. (b) Wormhole attack against an on-demand routing protocol. An adversary establishes a wormhole link between nodes s_9 and s_2, and node s_9 wants to send data to node s_2. The adversary forwards the route request broadcasted from node s_9, via the wormhole link to node s_2. Node s_2 replies with a route reply, and the adversary forwards the reply to node s_9, via the wormhole link. Nodes s_2, s_9 establish a route via the wormhole link, as if they were one hop neighbors. Similarly, if any of the nodes $\{s_1, s_3, s_4, s_5, s_7\}$ wants to send data to any of the nodes $\{s_9, s_{10}, s_{11}, s_{12}\}$, the routing paths established include the wormhole link

and if *A, B* are more than one hop away, then a one-hop route via the wormhole will be established from *A* to *B*.

As an example, consider Fig. 1(b) which is the same topology as in Fig. 1(a). Consider that the attacker establishes a wormhole link between nodes s_9 and s_2 and assume that node s_9 wants to send data to node s_2. When node s_9 broadcasts the *route request*, the attacker will forward the request via the wormhole link to node s_2. Node s_2 will reply with a *route reply* and the attacker using wormhole link will forward the reply to node s_9. At this point, nodes s_2, s_9 will establish a route via the wormhole link, as if they were one hop neighbors. Similarly, if any of the nodes $\{s_1, s_3, s_4, s_5, s_7\}$ wants to send data to any of the nodes $\{s_9, s_{10}, s_{11}, s_{12}\}$, the routing paths established will include the wormhole link.

From our examples and the existing literature [20], we note that the existence of wormhole links impacts the network routing service performance in the following three ways: (1) nodes can become sinkholes [25] without even being aware that they are victims of a wormhole attack (as noted in both Figs. 1(a), and (b), nodes s_2, s_9 become sinkhole nodes and attract all traffic from surrounding nodes). Hence, a significant amount of traffic is routed through the wormhole link and the attacker can control and observe a significant amount of traffic flow without the need to deploy multiple observation points. (2) If an attacker kept the wormhole link functional at all times and did not drop any packets, the wormhole would actually provide a useful network service by expediting the packet delivery. However, by selectively dropping packets, the attacker can lower the throughput of the network. (3) Furthermore, by simply switching the wormhole link on and off, the attacker can trigger a route oscillation within the network, thus leading to a DoS attack, driving the routing service to be unusable.

Wormhole attack against local broadcast protocols: In many applications, nodes need to communicate some information only within their neighborhood. For example, in localization protocols [27,43,44], nodes determine their location based on information provided by the neighbors. In wireless sensor networks, sensors performing monitoring (for example tracking the movement of an object), may broadcast local measurements to a *central node or clusterhead* that estimates target related parameters, such as location and velocity of the target. In such applications, false local information can lead to significant performance degradation of the estimation algorithms. Currently, all the tracking algorithms assume that the input data is noisy and at times may use cryptographic mechanisms to verify the authenticity of the data.

As an example, consider the setup in Fig. 2, where sensor node s_1 is responsible for triggering an alarm in region *A*, if the temperature in region *A* rises above a certain threshold. Let's assume that sensor s_1 makes use of a majority-based algorithm that triggers the alarm if the majority of its immediate neighbors report temperature measurements above a specific threshold. Assume that an attacker records the temperature broadcasts from region *B* and re-broadcasts the data to region *A* via the wormhole link. If the number of distinct measurements replayed via the wormhole link exceeds the collected distinct measurements from region *A*, the temperature in region *A* may never impact the decision to trigger the alarm in *A*.

From the above examples, we note that in order to prevent the wormhole attack, there must be some mechanism to ensure that any transmission received by a node *s* indeed originates from a valid one-hop neighbor of *s* that is located within its communication range. We now show that these ideas can be formalized using a graph theoretic framework.

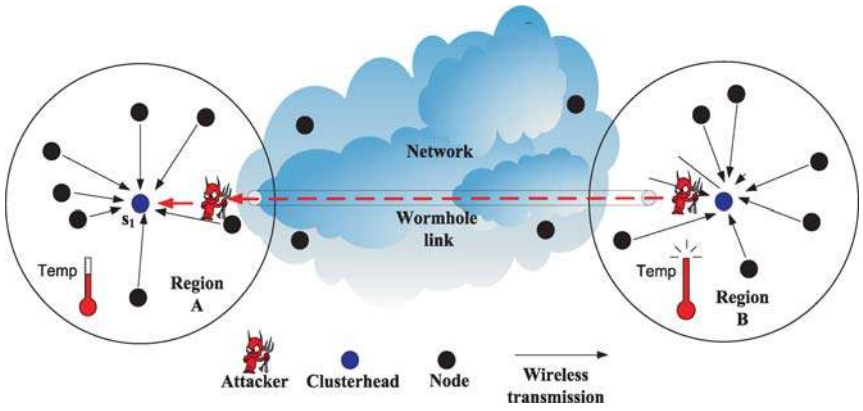


Fig. 2 Wormhole attack against a local broadcast protocol. Node s_1 is responsible for triggering an alarm in region A, if the majority of nodes in region A report a temperature above a certain threshold. Region B has a higher temperature than the threshold. An attacker records the temperature broadcasts from region B and re-broadcasts the data to region A via the wormhole link. If the number of distinct measurements replayed via the wormhole link exceeds the collected distinct measurements from region A, the temperature in region A will never impact the decision to trigger the alarm in region A

2.3. Graph theoretic formulation of the wormhole problem and its solution

Consider an ad hoc network deployed with any node i having a communication range r. Such a network can be modeled as a geometric graph [35], defined as follows:

Geometric graph: Given a finite set of vertices $V \subset \mathcal{R}^d (d = 2$, for planar graphs), we denote by $G(V, r)$ the undirected graph with vertex set V and with undirected edges connecting pairs of vertices (i, j) with $\|i - j\| \leq r$, where $\|\cdot\|$ is some norm on $\mathcal{R}^d$ [35]. The entries of the edge, or connectivity matrix, denoted by e, are given by:

$$e(i, j) = \begin{cases} 1, & \text{if} \quad \|i - j\| \leq r \\ 0, & \text{if} \quad \|i - j\| > r. \end{cases} \tag{1}$$

Geometric graphs have long been considered a useful model for deriving insightful analytic results in wireless ad hoc networks [3,9,14,15]. The network protocols developed for ad hoc networks are *implicitly* designed based on the geometric graph model. For example, routing algorithms assume that for two nodes that are not within communication range, a multi-hop route must be constructed. In addition, the networking protocols define one-hop neighbors of an arbitrary node s as those nodes that can directly hear any broadcast transmission from node s. However, the existence of wormhole links violates the model in (1) by allowing direct links longer than r, thus transforming the initial geometric graph $G(V, r)$ into a logical graph $\tilde{G}(V, E_{\tilde{G}})$, where arbitrary connections can be established. Hence, even a single non-trivial wormhole will always result in a communication graph with increased number of *ones* in the binary connectivity matrix compared to the connectivity matrix of the wormhole-free communication graph. We now formalize the wormhole problem based on the geometric graph property expressed in (1).

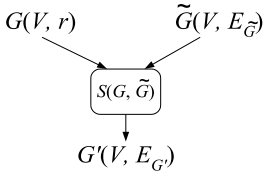


Fig. 3 The wormhole embedded graph theoretic model. The wormhole-infected graph $\tilde{G}(V, E_{\tilde{G}})$ is transformed via a solution $S(G, \tilde{G})$ into a communication graph $G'(V, E_{G'})$, with $E_{G'} \subseteq E_G$

Wormhole problem: A network is vulnerable to the wormhole attack if there exists at least one edge $e(i, j)$ such that $e(i, j) = 1$ for $\|i - j\| > r$, where r is the communication range of nodes.

Any candidate solution to the wormhole problem should construct a communication graph $G'(V, E_{G'})$, where no link longer than r exists. Any edge $e(i, j)$ of the communication graph $G'(V, E_{G'})$ satisfies (1), and hence, the communication graph solving the wormhole problem will always be a subgraph of the geometric graph of the network, i.e. $G'(V, E_{G'}) \subseteq G(V, r)$. Figure 3 graphically represents the extraction of the wormhole-free communication graph $G'(V, E_{G'})$ from the wormhole-infected graph $\tilde{G}(V, E_{\tilde{G}})$ via the application of a transformation $S : G \times \tilde{G} \to G'$, when the geometric graph $G(V, r)$ is known.

Note that the wormhole infected graph $\tilde{G}$, the geometric graph G, and the communication graph G', have the same set of vertices V since, as mentioned in Section 2.1, the devices deployed by the adversary launching a wormhole attack do not become part of the network (they do not acquire valid network identities). Also, note that the sets of edges E_r, $E_{G'}$, $E_{\tilde{G}}$ are determined based on fixed node locations. If the nodes of the network are mobile, the set of edges on each graph may

change according to the node locations at any given time. Despite the changing network topology, at any time and for a given location, any valid solution to the wormhole problem should construct a communication graph that is a subgraph of the geometric graph. We now formalize the necessary and sufficient condition for solving the wormhole problem in the following theorem.

Theorem 1. *Given a geometric graph G(V, r) defined as in* (1)*, and an arbitrary logical graph* $\tilde{G}(V, E_{\tilde{G}})$*, a transformation* $S : G \times \tilde{G} \rightarrow G'$ *of* $\tilde{G}(V, E_{\tilde{G}})$ *into a communication graph* $G'(V, E_{G'})$ *is a solution to the wormhole problem iff the set of edges of* G' *is a subset of the set of edges of the G(V, r), i.e.* $E_{G'} \subseteq E_G$.

Proof: Assume that $G' = S(G, \tilde{G})$ prevents the wormhole attack. Let C_X denote the connectivity matrix of graph X. If $E_{G'} \nsubseteq E_G$, there exists a pair of nodes (i, j) for which: $C_G(i, j) = 0$ and $C_{G'}(i, j) = 1$. For such node pairs, $e(i, j) = 1$, with $\|i - j\| > r$, and the communication range constraint is violated. Hence, in order for $S(G, \tilde{G})$ to prevent the wormhole attack, it follows that $E_{G'} \subseteq E_G$.

The converse follows immediately. If $E_{G'} \subseteq E_G$, then $C_{G'}(i, j) \leq C_G(i, j), \forall\, i, j \in V$. Hence, there is no edge $e'(i, j) \in E_{G'}$ such that $e'(i, j) = 1, \|i - j\| > r$, and the graph G' is wormhole free. □

Note that a trivial graph G' with no links ($E_{G'} = \emptyset$) satisfies the conditions of the Theorem 1. However, to ensure communication between all network nodes, we seek solutions that construct a connected subgraph of G. A necessary but not sufficient condition for a connected subgraph to exist is that the geometric graph G is also connected.

We also note that the transformation $G' = S(G, \tilde{G})$ requires the knowledge of the geometric graph $G(V, r)$, defined by the location of the vertices, and the communication range r. When nodes do not have a global view of the network (know the location of other nodes), to verify Theorem 1, an alternative way to construct a connected subgraph of the geometric graph $G(V, r)$ must be developed. If the geometric graph can be constructed, all wormhole links can be eliminated using Corollaries 1, 2.

Corollary 1. *We can identify and eliminate the wormhole links of a logical graph* $\tilde{G}(V, E_{\tilde{G}})$ *by performing an exclusive or (XOR) operation between the connectivity matrices of* $\tilde{G}$ *and the geometric graph* $G(V, r)$*, corresponding to the set of vertices V and communication range r.*

To illustrate how we can identify the wormhole links using Corollary 1, consider the network of Fig. 1(a). Each row i of the connectivity matrix denotes the links of node i (we have assumed that links between nodes are bi-directional). Using the notation $C_X(i)$ for the row vector of matrix C_X corresponding to the node s_i, the row vectors corresponding to node s_2, for the connectivity matrices C_G, and $C_{\tilde{G}}$ are

$$C_{\tilde{G}}(2) = [1\,0\,1\,1\,1\,0\,0\,0\,1\,0\,0\,0\,0],$$
$$C_G(2) = [1\,0\,1\,1\,1\,0\,0\,0\,0\,0\,0\,0\,0].$$

By performing an XOR operation between $C_{\tilde{G}}, C_G$, we can identify all wormhole links and corresponding nodes that are affected by the non-zero entries in matrix $(C_{\tilde{G}} \oplus C_G)$. In Fig. 1(a), the second row of the matrix $C_{\tilde{G}} \oplus C_G$ resulting from the XOR operation is

$$(C_{\tilde{G}} \oplus C_G)(2) = [0\,0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0]\,, \tag{2}$$

and a wormhole link exists between node s_2 and node j for which $(C_{\tilde{G}} \oplus C_G)(2, j) = 1$. In our example, the wormhole link between node s_2 and node s_9 is successfully identified.

Note that according to Theorem 1 any connected subgraph of $G(V, r)$ is sufficient to prevent any wormhole attack. For a subgraph of $G(V, r)$ an XOR operation may identify valid links of $G(V, r)$ as wormhole links. However, along with the false positives, all the wormhole links are detected. For example, consider a subgraph $G'(V, E_{G'}) \subset G(V, r)$ for the network of Fig. 1(a), for which node s_2 is not connected to node s_3. For the subgraph G', the second row of the connectivity matrix is

$$C_{G'}(2) = [1\,0\,0\,1\,1\,0\,0\,0\,0\,0\,0\,0\,0],$$
$$(C_{\tilde{G}} \oplus C_{G'})(2) = [0\,0\,1\,0\,0\,0\,0\,0\,1\,0\,0\,0\,0].$$

By performing an XOR operation between $C_{\tilde{G}}, C_{G'}$, we identify all wormhole links (link from node s_2 to node s_9) and some false positives (link from node s_2 to node s_3). Eliminating both the wormhole links and the false positives to construct graph G' is an acceptable solution as long as G' is a connected graph. We summarize the wormhole elimination in Corollary 2.

Corollary 2. *We can identify and eliminate the wormhole links of a logical graph* $\tilde{G}(V, E_{\tilde{G}})$ *by performing an exclusive or (XOR) operation between the connectivity matrices of* $\tilde{G}$ *and any subgraph* $G'(V', E'_G)$ *of G(V, r), where G(V, r) is the geometric random graph corresponding to the set of vertices V and communication range r.*

Theorem 1 and Corollaries 1, 2, provide the necessary framework to detect and prevent any wormhole attack. We will specifically utilize them in the context of *geometric random graphs*, since we assume that our network is randomly deployed. Based on our graph theoretic formulation, the wormhole problem can be reduced to the problem of

constructing a communication graph that is a connected subgraph of the geometric graph, without the explicit knowledge about the geometric graph. Before we present our solution on constructing a subgraph of the geometric random graph, we describe the needed network model assumptions.

3. Network model assumptions

Network setup: We assume that the network consists of a large number of nodes, randomly deployed within the network region $\mathcal{A}$. We also assume that a small fraction of network nodes, *called guards*, is assigned special network operations. Network nodes are deployed with a density ρ_s while guards are deployed with a density ρ_g, with $\rho_s \gg \rho_g$.

Antenna model: We assume that the guards can transmit with higher power than regular nodes and/or are equipped with different antenna types. Specifically:

(a) *Network nodes:* We assume that network nodes are equipped with omnidirectional antennas and transmit with a power P_s. The directivity gain of the node antenna is $D_s = 1$.
(b) *Guards:* We assume that guards can transmit with a power $P_g > P_s$. We also assume that guards can be equipped with either omnidirectional or directional antennas, with a directivity gain $D_g \geq 1$.

Based on the antenna model assumptions, both symmetric as well as asymmetric modes of communication between different network nodes are possible. Let the signal attenuation over space be proportional to some exponent γ of the distance d between two nodes, times the antenna directivity gain $D \in \{D_s, D_g\}$, i.e. $\frac{P_s}{P_r} = cD^2d^\gamma$, with $2 \leq \gamma \leq 5$, where c denotes the proportionality constant and P_r denotes the minimum required receive power for communication. If r_{nn} denotes the node-to-node communication range and r_{ng} denotes the node-to-guard communication range, then [1],

$$\begin{aligned} \frac{P_s}{P_r} &= cD_s^2(r_{nn})^\gamma = c(r_{nn})^\gamma, \\ \frac{P_s}{P_r} &= cD_sD_g(r_{ng})^\gamma = cD_g(r_{ng})^\gamma. \end{aligned} \tag{3}$$

From (3), it follows $r_{ng} = r_{nn}(D_g)^{\frac{1}{\gamma}}$. Similarly, if r_{gn} denotes the guard-to-node communication range (guards transmit with $P_g > P_s$ and hence, $r_{gn} > r_{ng}$), the guard-to-guard communication range r_{gg} is equal to $r_{gg} = r_{gn}(D_g)^{\frac{2}{\gamma}}$. For notational simplicity, we will refer to the node-to-node communication range as $r_{nn} = r$, the guard-to-node communication range as $r_{gn} = R$, and the guard directivity gain as D. Table 1 summarizes the four possible communication modes with appropriate ranges indicated.

Table 1 The four communication modes between nodes and guards. Each entry denotes the range of communication for that mode

	Receiver	
Sender	Node	Guard
Node	r	$rD^{\frac{1}{\gamma}}$
Guard	R	$RD^{\frac{2}{\gamma}}$

The assumption that guards are able to transmit with higher power than network nodes is a reasonable one, especially for low-power networks such as sensor networks. A typical sensor has a communication range from 3–30 m with a transmission power of $P_s = 0.75$ mW [31]. Hence, guards need to transmit with a power $P_g = 75$ mW to achieve a communication range ratio $\frac{R}{r} = 10$ when $\gamma = 2$ without even the use of directional antennas.

Note that we have assumed that the communication range of both the guards and the nodes does not vary with direction and the environment (unit disk graph model). This assumption has been made to facilitate the derivation of analytical expression, quantifying the level of security achievable by our method.[1] Clearly, while the unit disk model provides theoretical performance bounds, knowledge of the statistics of the variation of the communication range is needed to provide a more robust approach. We discuss the effect of the variation of the communication range due to the heterogeneity of the wireless medium in Section 6 and present performance evaluation analysis that takes the variation into account.

Resource constraints: We assume that network nodes are resource limited in the following ways:

(a) Due to hardware limitations (lack of GPS receiver), nodes may not know their location at all times. In addition, due to limited resource-constraints, generic nodes may not attempt to determine their location. However, we assume that guards do know their location either through GPS [18] or through some other localization method [43,44].
(b) We also assume that due to hardware limitations, there is no time synchronization between the network nodes or the guards. In addition, nodes are not equipped with hardware able to perform highly accurate time measurements in the nanoseconds.
(c) Due to computational power limitations, network nodes cannot perform expensive asymmetric cryptographic operations such as digital signatures [12,42]. Instead, they rely on efficient symmetric cryptography to generate, manage, and distribute cryptographic quantities and execute cryptographic operations, such as encryption/decryption, authentication, and hashing. We also

[1] The unit disk graph model has been used to represent ad hoc networks with identical devices being deployed in order to derive insightful theoretical results in diverse research topics, such as security [9,14], network connectivity [3,15], routing [16,23,24], and topology control [48].

assume that nodes and guards can be pre-loaded with needed cryptographic quantities before deployment.

System parameters: Since both guards and network nodes are randomly deployed, it is essential that we appropriately choose the network parameters, namely the guard density ρ_g and the guard-to-node communication range R, for a given deployment area $\mathcal{A}$, so that guards can communicate with nodes.

The random deployment of the network nodes and guards can be modeled after a *Spatial Homogeneous Poisson Point Process* [11]. The random placement of a set U of guards with a density $\rho_g = \frac{|U|}{\mathcal{A}}$ ($|\cdot|$ denotes the cardinality of a set) is equivalent to a sequence of events following a homogeneous Poisson point process of rate ρ_g. Given that $|U|$ events occur in area $\mathcal{A}$, these events are uniformly distributed within that area. The random deployment of a set S of nodes with a density $\rho_s = \frac{|S|}{\mathcal{A}}$, is equivalent to a random sampling of the deployment area with rate ρ_s [11].

Based on *Spatial Statistics* theory [11], if GH_s denotes the set of guards heard by a sensor s (i.e., being within range R from s), then the probability that a node hears exactly k guards is given by the Poisson distribution

$$P(|GH_s = k|) = \frac{(\rho_g \pi R^2)^k}{k!} e^{-\mu_g \pi R^2}. \tag{4}$$

Based on (4), we can compute the probability that every node of the network hears at least one guard as

$$P(|GH_s| > 0, \forall_s \in S) = \left(1 - e^{-\rho_g \pi R^2}\right)^{|S|}. \tag{5}$$

Using (5), we can determine the desired guard density ρ_g or guard-to-node communication range R, so that each node hears at least one guard with a probability p,

$$\rho_g \geq \frac{-\ln\left(1 - p^{\frac{1}{|S|}}\right)}{\pi R^2}, \quad R \geq \sqrt{\frac{-\ln\left(1 - p^{\frac{1}{|S|}}\right)}{\pi \rho_g}}. \tag{6}$$

Both inequalities in (6) are independent from the node density ρ_s. Hence, once the deployment region is sufficiently covered by guards, nodes can be deployed as dense as desired with $P(|GH_s| > 0, \forall\, s \in S)$ remaining constant. The detailed derivation of (5) is presented in the Appendix A.

Probability of hearing a given number of guards: Assume now that we require each node to hear at least k guards ($|GH_s| = k$). That probability is given by

$$P(|GH_s| \geq k, \forall_s \in S) = \left(1 - \sum_{i=0}^{k-1} \frac{(\rho_g \pi R^2)^i}{i!} e^{-\rho_g \pi R^2}\right)^{|S|}. \tag{7}$$

Note that (7) allows the choice of parameters ρ_g, R so that a node will hear at least k guards with a given probability. Since all random variables are non-negative, the expected number of guards heard by each node, $E(|GH_s|) = \rho_g \pi R^2$, is significantly higher than k. For example, for $R = 20$, to allow every node to hear at least 4 guards with probability $P(|GH_s| \geq 4, \forall_s \in S) = 0.99$, we need a guard density of $\rho_g = 0.02$. For $\rho_g = 0.02$, $E(|GH_s|) = 25.13$. Hence, $P(|GH_s| \geq k, \forall_s \in S)$ is a stricter requirement than $E(|GH_s|) \geq k$. Derivations of $P(|GH_s| \geq k, \forall_s \in S)$ and $E(|GH_s|)$ are presented in Appendix A.

4. Local broadcast keys

As we showed in Section 2.3, broadcasted messages that are destined only to the local neighborhood are timely replayed in regions that are not within the communication range of the source of the messages. Since the replayed messages are both authentic and decryptable at the destination point of the attack, a wormhole link is established between the nodes at the origin point of the attack and the nodes at the destination point, as if the nodes were one-hop neighbors. Hence, wormhole links violate the communication range constraint by allowing nodes that are not within communication range to directly communicate. In order to prevent the establishment of wormhole links, we showed that any candidate solution should construct a communication graph that is a subgraph of the geometric graph of the network.

A wormhole attack is successful when the replayed messages that are destined only to the local neighborhood are decryptable and can be authenticated outside that neighborhood. Once the attacker replays broadcasted messages outside the local neighborhood in a timely manner, nodes at the ends of the wormhole link are led to believe that they are one-hop neighbors. However, if only the nodes within a local neighborhood can decrypt and/or authenticate the messages broadcasted within that neighborhood, nodes out of communication range of each other will not conclude that they are one-hop away. Hence, the communication graph constructed by securely identifying the one-hop neighbors is a subgraph of the geometric graph of the network and the wormhole attack is eliminated.

In order for a broadcast message intended for one-hop neighbors to be decryptable only by the one-hop neighbors, each node should be able to encrypt broadcast messages with keys only known to all of its one-hop neighbors. We call such keys *Local Broadcast Keys* (LBKs). Hence, the problem of eliminating wormhole links reduces the problem of allowing nodes to establish LBKs with their one-hop neighbors. Once the LBKs are established, the resulting communication graph will be a subgraph of the geometric graph of the network.

In this section, we first define local broadcast keys and constructively show that LBKs lead to a wormhole-free communication graph that is a subgraph of the geometric graph of the network. We then present one centralized and one decentralized mechanism for establishing LBKs, followed by a probabilistic analysis of the level of security achieved.

4.1. Definition and correctness

Definition. For a node i, we define the neighborhood N_i as $N_i = \{j : \|i - j\| \leq r\}$. Given a cryptographic key K, let U_K denote the set of nodes that hold key K. We assign a unique key K_i called *Local broadcast key* LBK of i, to all $j \in N_i$ so that $U_{K_i} = N_i$ and $K_i \neq K_j, \forall i \neq j$. Hence, by definition, all one-hop neighbors of node i possess the LBK of node i. We follow the convention that any message from node i to j is encrypted with K_i, though either K_i or K_j can be used between nodes i, j. Hence, a link between nodes i, j exists iff $i \in N_j$ or $j \in N_i$.

Theorem 2. *Given $K_i, N_i, \forall i \in V$, where V is the set of vertices defined by network nodes, and an arbitrary logical random graph $\tilde{G}(V, E_{\tilde{G}})$, the edge matrix $E_{G'}$. defined by*

$$e_{G'}(i, j) = \begin{cases} 1, & if \quad i \in U_{K_j} \cup j \in U_{K_i} \\ 0, & if \quad Else, \end{cases} \tag{8}$$

yields the desired wormhole-free graph $G'(V, E_{G'})$, such that $E_{G'} \subseteq E_G$, where G(V, r) is the geometric random graph defined in (1).

Proof: By the definition of $E_{G'}$, there exists a link $e_{G'}(i, j) = 1$ if and only if the two nodes hold at least one LBK. But, according to the definition of LBK, a node $i \in U_{K_j}$ iff $i \in N_j$, which in turn implies that i, j satisfy (1), which defines the links of the geometric graph $G(V, r)$. Hence, $e_{G'}(i, j) = 1$, iff $\|i - j\| \leq r$, $E_{G'} = E_G$ and, therefore, $G' \equiv G$. According to Theorem 1, if a transformation $S(G, G)$ results in a graph $G'(V, E_{G'})$ such that $E_{G'} \subseteq E_G$, then G' is a *wormhole-free* graph. □

As a side remark, we note that since $G' \equiv G$ and if G is connected, then G' is also connected. Also, given that LBKs are established for any network nodes, the wormhole attack can be prevented even in the absence of any location information. The LBK solution reconstructs the geometric graph $G(V, r)$ by encrypting the information exchange and disclosing the decryption keys only to direct neighbors. However, the challenge of establishing LBKs in a network may or may not require location information. In what follows, we present two mechanisms by which we can assign local broadcast keys to the nodes of the network.

4.2. Local broadcast key establishment mechanisms

4.2.1. Key distribution from a central authority

Wireless ad hoc networks have been visualized to operate under both centralized and decentralized control depending on the applications and the services that they provide. Though our research mainly focuses on decentralized systems, for completeness, we first show how LBKs can also be established in centralized systems.

Assume that a central authority has a global view of the network topology (knows the location of all nodes) and that a security association has been established between every node, and the central authority (every node shares a pairwise key with the central authority). Similar assumptions have been made in the centralized wormhole prevention scheme presented in [47].[2] It is quite simple to see that the central authority can construct the geometric graph $G(V, r)$ using the location of the nodes and the communication range constraint r. Once the geometric graph $G(V, r)$ is constructed, the central authority can distribute a unique LBK to each node and its one-hop neighbors, via the secure channel established based on the security association shared with each node. Once the LBKs have been established, any broadcast encrypted with the LBK of a node s_i can only be decrypted by the one-hop neighbors of s_i. Hence, using wormhole to replay messages at one neighborhood encrypted with the LBK of another will not introduce any vulnerability.[3]

The centralized authority-based LBK establishment mechanism exhibits drawbacks that are commonly noted in any centralized solution. First, the central authority constitutes a single point of failure. Second, in case of a mobile ad hoc network, the base station needs frequent updates of the location of each node in order to maintain an up-to-date geometric graph and update the LBKs according to the changing topology. The LBK update has to be performed via unicast messages from the base station to every node and, hence, can add prohibitively high overhead for the network. Finally, the centralized method requires knowledge of the entire network topology (location of all nodes). A base station can acquire the node location if the network is systematically deployed, or by using a wormhole-resistant localization method [7,27–30]. We now describe a decentralized LBK establishment mechanism that requires only a small fraction of the nodes to have knowledge of their location.

[2] The authors in [47] assume that a base station receives information about the relative position of each node via a channel secured with a group key known to all nodes and the base station.

[3] Since the central authority can reconstruct the geometric graph $G(V, r)$, it can also inform every node about their one-hop neighbors via a secure channel and, hence, prevent the wormhole attack.

4.2.2. Decentralized establishment of local broadcast keys

We present a three-step algorithm to allow nodes to establish LBK in a decentralized manner. In step one, every guard G_i broadcasts fractional keys FK_i to the network. Every node collects the fractional keys from all guards that it can hear. In step two, every node broadcasts the Ids of the fractional keys that it holds. If two nodes s_i, s_j share more than *th* fractional keys, they use all common fractional keys to generate a pairwise key K_{s_i,s_j}. In step three, a node s generates an LBK K_s and unicasts it to every node that it shares a pairwise key with. Before we describe the three steps in detail, we present the cryptographic mechanisms of our decentralized LBK scheme.

1) Cryptographic mechanisms

Encryption: To protect the distribution of the fractional keys, all broadcasts from the guards are encrypted with a global symmetric key K_0, preloaded before deployment. In addition, a node s shares a symmetric pairwise key K_{s,g_i} with every guard g_i, also preloaded. Since the number of guards deployed is relatively small, the storage requirement at the node is within the storage constraints (a total of $|U|$ keys), even for memory scarce nodes. For example, mica motes [31] have 128 Kbytes of programmable flash memory. Using 64-bit RC5 [41] symmetric keys and for a network with 200 guards, a total of 1.6 Kbytes of memory is required to store all the symmetric pairwise keys of the node with all the guards.

In order to save storage space at the guard side (guards would have to store $|S|$ keys), the pairwise key K_{s,g_i} is derived by a master key K_{g_i}, using a pseudo-random function [45] h and the unique node Id_s, $K_{s,g_i} = h_{K_{g_i}}(Id_s)$. Hence, given an Id_s, a guard can compute its pairwise key with any node whenever needed, without having to store any pairwise keys.

Guard ID authentication: The use of a global symmetric key K_0 does not provide any authentication on the source of the message. Hence, any guard or node holding the global key can broadcast fractional keys encrypted with K_0. Though we have assumed that the global symmetric key K_0 is not compromised and that network entities do not operate maliciously, in order to allow nodes to authenticate the guards within one-hop, we provide a lightweight authentication mechanism.[4] Our scheme is based on *efficient one-way hash chains* [26], that have also been used extensively in broadcast authentication protocols [38,39].

Each guard g_i is assigned a unique password PW_i. The password is blinded with the use of a *collision-resistant* hash function such as SHA-1 [45]. Due to the collision resistance property, it is computationally infeasible for an attacker to find a value PW_i', such that $H(PW_i) = H(PW_i')$, $PW_i \neq PW_i'$. The hash sequence is generated using the following equation:

$$H^0 = PW_i, \quad H^q = H(H^{q-1}), \quad i = 1, \ldots, n,$$

with n being a large number and H^0 never revealed to any node. In addition, due to the one-way property of the hash chain, it is computationally infeasible for an adversary to derive values of the hash chain that have not been already published by the guard [26]. Each node is preloaded with a table containing the Id of each guard and the corresponding hash value $H^n(PW_i)$. For a network with 200 guards, we need 8 bits to represent node Ids. In addition, hash functions such as SHA-1 [45] have a 128-bit output. Hence, the storage requirement of the hash table at any node is only 3.4Kbytes. To reduce the storage needed at the guard side, we employ an efficient storage/computation method for hash chains of time/storage complexity $O(\log^2(n))$ and compute any hash chain values when needed [10].

2) Steps of the key establishment scheme

Step 1. Initially, every guard g_i generates a random fractional key FK_i. Guards broadcast their fractional keys encrypted with the global symmetric key K_0. Every broadcast message also contains the coordinates (X_i, Y_i) of the transmitting guard, the next hash value in the hash chain that has not been published, $H^{n-q}(PW_i)$, and the hash chain index q. The broadcast message format is

$$\text{Guard } g_i : \{FK_i \| (X_i, Y_i) \| H^{n-q}(PW_i) \| q\} K_0, \tag{9}$$

where $\{A \| B\}_K$ denotes concatenation of A, B and encryption with key K.

Every node collects the fractional keys from all the guards that it can hear and verifies that $H(H^{n-q}(PW_i)) = H^{n-q+1}(PW_i)$. If a node has not received some intermediate values of the hash chain due to packet loss, it can use the hash index q to re-synchronize to the current published hash value. Assume that the latest hash value of the chain of guard g_i stored by a node s is $H^{n-z}(PW_i)$, with $z < q$. Node s can re-synchronize with the hash chain of guard g_i upon receipt of the hash value $H^{n-q}(PW_i)$ by applying $(q - z)$ consecutive hash operations to $H^{n-z}(PW_i)$.

For all received messages for which the verification of the hash is correct, the node stores the fractional keys FK_i, the coordinates of each guard (X_i, Y_i), the latest published hash values of the chain, $H(H^{n-q}(PW_j))$, and the hash index m. In Fig. 4(a), guards g_1–g_5 broadcast their fractional

[4] The guard authentication mechanism provides a basis for the future enhancement of the system against other type of attacks, such as the Sybil attack [13,33].

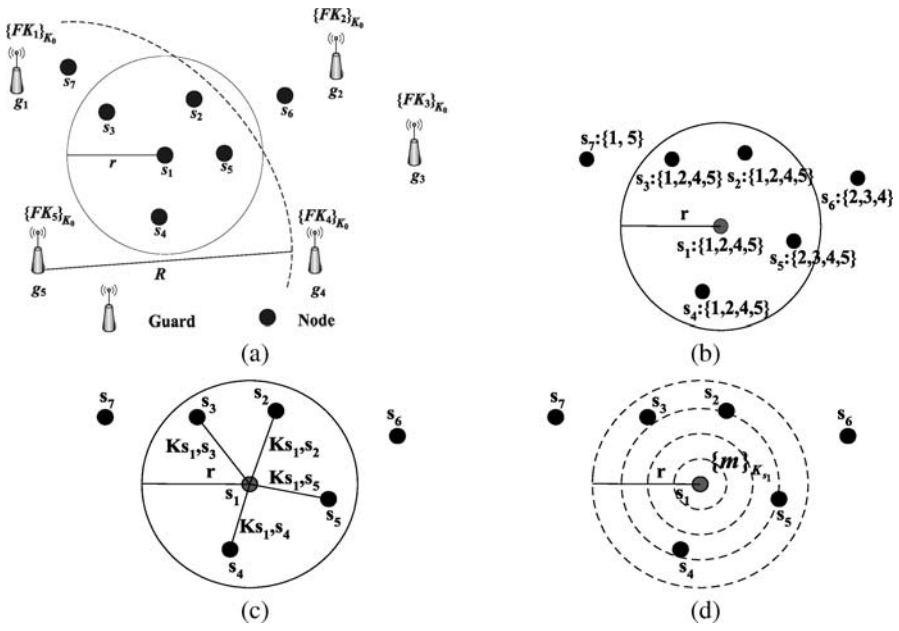


Fig. 4 (a) Guards g_1–g_5 broadcast fractional keys FK_1–FK_5 encrypted with the global broadcast key K_0. The location of the guards and the hash chain value is also included in every broadcast. (b) Nodes announce the Id's of the fractional keys that they hold. (c) Neighbor nodes that have in common at least three fractional keys ($th = 3$) establish a pairwise key. Node s_1 has at least three common fractional keys with all nodes within one hop. (d) Node s_1 establishes a broadcast key K_{s_1} with every one hop neighbor and uses it to broadcast a message m encrypted with K_{s_1}

keys FK_i encrypted with the global broadcast key K_0. Nodes s_1–s_7 decrypt the message with the key K_0, and verify the authenticity of the broadcasting guards.

Step 2. Once the nodes have collected the fractional keys from all the guards that they hear, they broadcast a message indicating the identities of the fractional keys that they hold and a node specific threshold value, encrypted with the global symmetric key K_0. Since every node is aware of the correspondence between the fractional keys that it has acquired and the identities of the guards that provided the fractional keys, the nodes need only broadcast the identities of the guards that they heard, in order to indicate which fractional keys they hold. The identities of the guards uniquely define the identities of the fractional keys broadcasted by those guards.[5]

If two neighbor nodes s_1, s_2 have in common fractional keys $\{FK_1, FK_2, \ldots FK_m\}$ with m above a threshold th, they individually generate a pairwise key, $K_{s_1,s_2} = H(FK_1 \| FK_2 \| \ldots \| FK_m)$, where H is a collision-resistant hash function [26]. A node s_1 can verify the claim of another node s_2 about holding a specific set of keys by challenging the claimant node. If node s_2 claims to hold a set of keys $\{FK_1 \| FK_2 \| \ldots \| FK_m\}$, it should be able to generate the key K_{s_1,s_2}. To verify such a claim, the verifying node s_1 first broadcasts a nonce, encrypted with the key K_{s_1,s_2} generated from the fractional keys corresponding to the guard Ids transmitted by the claimant node s_2. If the claimant node s_2 indeed holds the keys $\{FK_1 \| FK_2 \| \ldots \| FK_m\}$, it will be able to generate the same pairwise key K_{s_1,s_2}, decrypt the nonce, and reply to the verifying node.

For example, if node s_1 is the verifying node and s_2 is the claimant node, s_1 encrypts a nonce η_1 with K_{s_1,s_2} and challenges node s_2 to reply with $J(\eta_1)$, where $J(x)$ is a simple function, such as $J(x) = x - 1$. If node s_2 were to really hold the fractional keys that it advertised, it would generate the pairwise key K_{s_1,s_2}, and hence, will be able to decrypt the nonce and reply with $J(\eta_1)$, encrypted with K_{s_1,s_2}. The message exchange occurring between s_1 and s_2 in Step 2 is

$$s_1 \to s_2 : \{\eta_1\}_{K_{s_1,s_2}} \quad s_1 \to s_2 : \{J(\eta_1)\}_{K_{s_1,s_2}}.$$

Note that we require that the claimant node replies to the challenge η_1 with $J(\eta_1)$ rather than the nonce itself in order to prevent an adversary from replaying the challenge message as a valid response.

In Fig. 4(c), the threshold value is set to $th = 3$. Node s_1 establishes a pairwise key with all its neighbors that have at least three fractional keys in common. Note that s_1 does not share sufficient fractional keys with s_6 and s_7 in order to establish a pairwise key. Hence, even in the presence of a wormhole link between s_1 and s_6 or s_7, non-neighboring nodes will not be able to establish a pairwise key.

Step 3. After pairwise keys have been established with one-hop neighbors, node s_i randomly generates an LBK K_{si} and unicasts it to every neighbor, encrypted with the pairwise key K_{s_i,s_j}. Node s_j stores its LBK K_{s_i}, used for encrypting its own messages, and also stores the LBKs of all its

[5] Note that two guards may individually generate the same *FK*, but given a guard Id, the *FK* is unique

one-hop neighbors that it shares sufficient fractional keys with, in order to decrypt their broadcast messages. We assume that $K_{s_i} \neq K_{s_j}, \forall_{s_j} \neq s_j$. In Fig. 4(d), s_1 has established a LBK K_{s_1} with its neighbors s_1–s_5 and uses it to encrypt the transmission of message m.

Before we present our decentralized local broadcast key establishment scheme in algorithmic form, we analyze the critical problem of allowing nodes to determine the threshold value for establishing pairwise keys with their immediate neighbors.

4.3. Setting the threshold for key establishment

In this section, we examine how the value of the threshold th affects the probability of sharing more than th fractional keys with immediate and non-immediate neighbors. We then propose mechanisms to increase the connectivity with one-hop neighbors while decreasing the probability of non-immediate neighbors to share more than th fractional keys.

4.3.1. Key establishment with immediate neighbors

Let the distance between two nodes s_1, s_2 be $l = \|s_1 - s_2\|$, as in Fig. 5(a). Any guard g_i that lies within the shaded area A_c is heard by both nodes s_1, s_2 and hence, its fractional key FK_i is received by both s_1, s_2. From Fig. 5(a), we can compute the area A_c as follows

$$\phi = \cos^{-1}\frac{l}{2R}, \quad A_c = 2R^2\phi - Rl\sin\phi. \tag{10}$$

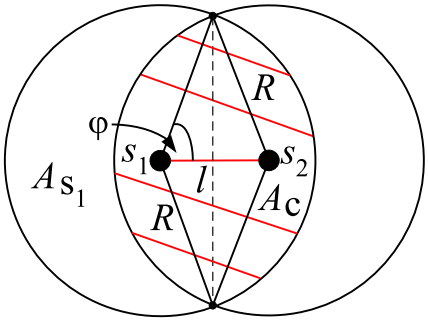
If GH_{A_c} denotes the set of guards located within A_c, the probability P_{key} for two nodes that are at a distance $l \leq r$ to establish a pairwise key is equal to the probability that more than th guards are located in A_c,

$$\begin{aligned} P_{key} &= P(|GH_{A_c}| \geq th) = 1 - P(|GH_{A_c}| < th) \\ &= 1 - \sum_{i=0}^{th-1}\left[\frac{(\rho_g A_c)^i}{i!}e^{-\rho_g A_c}\right]. \end{aligned} \tag{11}$$

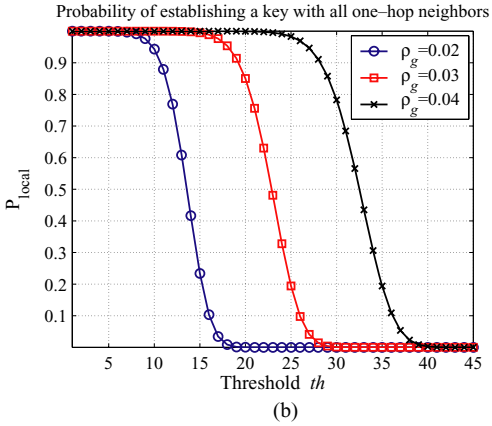
From (11), we compute the probability P_{local} for a node to be connected *to all* the nodes within its neighborhood. Let $P(N_s = i)$ denote the probability for a node s to have i neighbors. Since neighbors' nodes can be located at any distance $0 \leq l \leq r$ from node s, we can derive a lower bound on P_{local} by considering the worst case where every neighbor is located at the circle of radius r centered at the node s. Assuming that every one-hop neighbor is at the boundary of the communication range yields the worst case for P_{local}, since A_c attains its minimum value for $l = r$, and, hence, the probability of finding th guards in A_c becomes the smallest. P_{local} is expressed as

$$\begin{aligned} &P_{local} \\ &\geq \sum_{i=0}^{|S|} P(N_s = i, |GH_{A_c}| \geq th, \forall i) \\ &= \sum_{i=0}^{|S|} P(N_s = i)P(|GH_{A_c}| \geq th, \forall i) \qquad (12) \\ &= \sum_{i=0}^{|S|} P(N_s = i)P_{key}^i \qquad (13) \\ &= \sum_{i=0}^{|S|}\left(\frac{(\rho_s \pi r^2)^i}{i!}e^{-\rho_s \pi r^2}\right)\left(1 - \sum_{j=0}^{th-1}\left(\frac{(\rho_g A_c)^j}{j!}e^{-\rho_g A_c}\right)\right)^i, \qquad (14) \end{aligned}$$

with A_c given by (10) for $l = r$. In the computation of P_{local}, (12) follows from the fact that nodes are independently deployed from guards, (13) follows from the randomness in



(a)



(b)

Fig. 5 (a) Nodes s_1, s_2 are within communication range ($l \leq r$). All guards located in the area A_c are heard to both nodes s_1, s_2. (b) A lower bound on P_{local} for varying guard densities ρ_g and for a node density $\rho_s = 0.5$, when $\frac{R}{r} = 10$

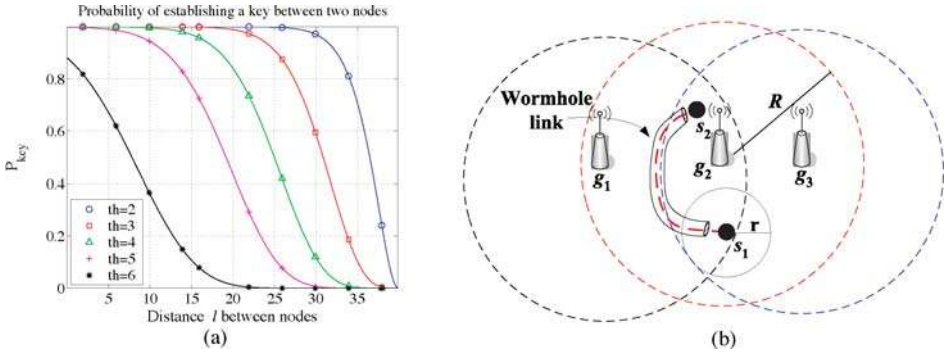


Fig. 6 (a) P_{key} for varying threshold values when $\rho_g = 0.03$. (b) Nodes s_1, s_2 hear guards g_1–g_3. An adversary replays the fractional key Id broadcast information of s_1 at point s_2, and the fractional key Id broadcast information of s_2 at point s_1. If the threshold is set to $th = 3$, sensors s_1 and s_2 are led to believe they are one hop away, establish a pairwise key and communicate through the wormhole link

the guard deployment (finding GH_{Ac} guards in an area A_c is independent on where A_c is located), and (14) follows from (11).

Given parameters r, ρ_s, we can select the threshold th and the parameters R, ρ_g, so that the probability P_{local} is close to unity (i.e., nodes establish pairwise keys with almost all their neighbors). In Fig. 5(b), we show the lower bound on P_{local} vs. th, for varying guard densities ρ_g and for a node density $\rho_s = 0.5$, when $\frac{R}{r} = 10$.

From (14), we can select the threshold th such that P_{local} is very close to unity. For example, for $\rho_g = 0.03$, setting the threshold to $th \leq 15$ will allow one-hop neighbors to share more than th fractional keys with a probability very close to unity. However, if we choose a low threshold value, neighbors more than one-hop away will also have in common more than th fractional keys. Hence, an adversary can establish a wormhole link between nodes more than one-hop away. In the next section, we examine the statistics on establishing keys between non-immediate neighbors.

4.3.2. Avoiding key establishment with non-immediate neighbors

To satisfy the definition of LBKs, nodes more than one hop away must not have more than th fractional keys in common. In Fig. 6(a), we show the probability $P_{key}(l)$ of two nodes to share more than th fractional keys depending on the distance l between them, as expressed by (11).

From Fig. 6(a), we observe that the value of the node-to-node communication range r is critical for the selection of the threshold. For example, if we set $r = 10\,m$ and $th = 5$, two nodes within communication range ($l < 10\,m$) establish a pairwise key with a probability almost unity. Two-hop neighbors located at a distance $l = 2r$ from a node s have a $P_{key} = 0.43$ to share more than $th = 5$ fractional keys. Such a probability value is prohibitively high. In order to reduce the P_{key} for non-immediate neighbors, we examine the reasons why P_{key} is high for distances $l > r$ and propose remedies to avoid key establishment between non-immediate neighbors.

Problem 1. In our analysis in Section 4.3.1, we have considered the threshold to be a global variable, the same for all deployed nodes. However, in a random deployment, not all nodes hear the same number of guards. Hence, for some nodes, the threshold value is too high to allow them to connect to their immediate neighbors, while for other nodes, the threshold value is too low to isolate non-immediate neighbors. To avoid the shortcomings of selecting a global threshold for all nodes, we propose each node to select its own threshold, based on number of guards heard at each node.

Problem 2. The use of omnidirectional antennas can increase the number of non-immediate neighbors vulnerable to the wormhole attack under the following scenario. Consider Fig. 6(b), where nodes s_1, s_2 are not within communication range. Due to the omnidirectionality of the guard antennas, both s_1, s_2 are able to hear the same set of guards $\{g_1, g_2, g_3\}$ and, hence, acquire the same set of fractional keys $\{FK_1, FK_2, FK_3\}$. In Step 2 of our decentralized LBK establishment scheme, the two nodes broadcast the Ids of the fractional keys that they hold, indicating the guards that they hear. Since the two nodes are not within communication range, in the absence of a wormhole they would not be able to establish an LBK. However, consider an adversary mounting wormhole attack that records the fractional key Ids broadcast information of s_1, tunnels it via the wormhole link to s_2, and replays it. Similarly, the adversary records the fractional key Id broadcast of s_2, tunnels it at s_1 and replays it. If the threshold for establishing communication is set to $th = 3$, s_1, s_2 will establish a pairwise key K_{s_1,s_2}, assuming that they are one hop away.

To account for the lack of direction in the distribution of the fractional keys at the expense of increased hardware complexity, guards may be equipped with M directional antennas

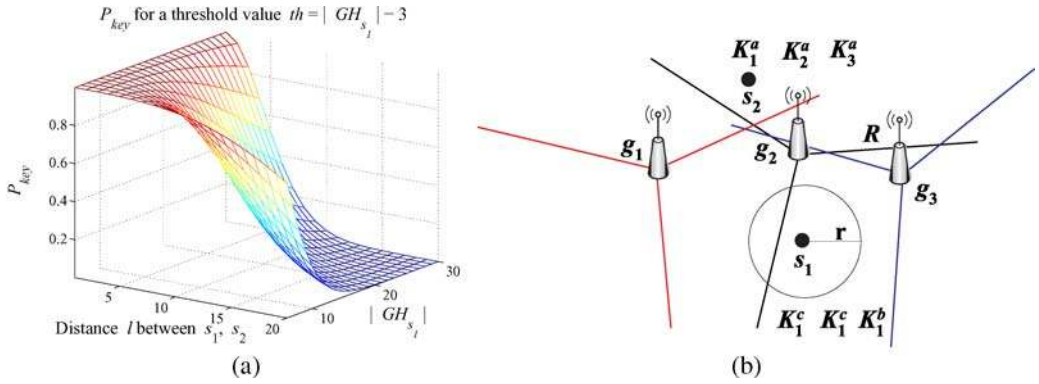


Fig. 7 (a) P_{key} for a varying threshold value equal to $th = |GH_{s_1}| - 3$. (b) Use of directional antennas for the distribution of fractional keys

of beamwidth $\frac{2\pi}{M}$ each. Guards transmit different fractional keys at each antenna sector and, hence, two nodes need to hear the same antenna sectors of the same guards in order to acquire common fractional keys.

4.3.3. Local threshold computation

In the previous section, we argued that setting the threshold globally can prohibit some immediate neighbors from establishing pairwise keys and allow some non-immediate neighbors to share more than *th* fractional keys. Hence, it is preferable that each node locally computes the threshold *th* based on the number of guards that it hears.

Assume that a sensor s_1 can hear $|GH_{s_1}|$ guards and wants to establish a pairwise key with node s_2 located at distance $l \leq r$ from s_1, as in Fig. 5(a). The probability that s_1, s_2 hear *th* common guards, given that $|GH_{s_1}|$ guards are heard by s_1, is equivalent to the probability that *th* guards are located within A_c, given that $|GH_{s_1}|$ of them are located within the area inside the circle of radius R centered at s_1. Due to the random guard deployment, if GH_{s_1} guards are located within a specific region, those guards are uniformly distributed [11]. Hence, if a single guard is deployed within the communication area of a node πR^2, the probability for that guard to be within A_c is $p_g = \frac{A_c}{\pi R^2}$. Since we assume random guard deployment, the event of a guard g_i being within A_c is independent of the event of guard g_j being within A_c. Hence, the probability that more than *th* guards are deployed within A_c, given that a total of $|GH_{s_1}|$ are deployed within πR^2 is,

$$\begin{aligned} P_{key} &= P\left(\left|GH_{A_c}\right| \geq th \middle\| GH_{s_1}\right| = k\right) \\ &= \sum_{i=0}^{k-th} \binom{k}{th+i} p_g^{th+i} (1-p_g)^{k-th-i} \\ &= \sum_{i=0}^{k-th} \binom{k}{th+i} \frac{A_c}{\pi R^2}^{th+i} \left(1 - \frac{A_c}{\pi R^2}\right)^{k-th-i}. \end{aligned} \tag{15}$$

Note that the binomial in (15) cannot be approximated by a Poisson distribution since k may not be much bigger than one and A_c has a comparable size to πR^2. In Fig. 7(a), we show the P_{key}, for different values of guards heard $|GH_{s_1}|$ and different distances between s_1, s_2, when the threshold is set to $th = |GH_{s_1}| - 3$. The selection of $th = |GH_s| - 3$ serves as an example to illustrate the idea of the locally computed threshold. In Section 6, we will provide extensive simulation studies for the selection of *th*.

Using (15), each node s_i can determine the threshold th_{s_i} individually depending on the number of guards that it hears. For example, if node s_i has a threshold of th_{s_i} and node s_j has announced that it holds at least th_{s_i} fractional keys known to s_i, node s_i will challenge s_j with a nonce η_i and s_j will reply with $J(\eta_i)$ encrypted with K_{s_i,s_j}. However, node s_j may hear a different number of guards and, hence, decide upon a different threshold value th_{s_j}. In such a case, s_j will repeat the pairwise key establishment process in order to agree on an additional pairwise key with node s_i. It is also possible that $\min\ (th_{s_i}, th_{s_j}) \leq |\cap (ID_{s_i}, ID_{s_j})| < \max(th_{s_j}, th_{s_j})$ and, hence, only unidirectional secure communication can be established between two one-hop neighbors. To establish only bidirectional links between one-hop neighbors, we can modify the pairwise key establishment condition by selecting a common threshold value th_{s_i,s_j} at both engaging nodes. To achieve maximum network connectivity, nodes, s_i, s_j can set the common threshold value th_{s_i,s_j} equal to the minimum of the two individual thresholds, th_{s_i}, th_{s_j}. However, in such a case, the probability of establishing a wormhole with a non-immediate neighbor grows larger for the node with the higher threshold. To tradeoff connectivity for protection against wormholes, nodes s_i, s_j can set the threshold value th_{s_i,s_j} equal to the maximum of the two individual thresholds, th_{s_i}, th_{s_j}. Two nodes s_i, s_j establish a pairwise key according to the following rule

$$K_{s_i,s_j} = \begin{cases} H(FK_1, FK_2, \ldots FK_m), \\ \quad \text{if } m = |\cap \left(ID_{s_i}, ID_{s_j}\right)| \geq \max\left\{th_{s_i}, th_{s_j}\right\} \\ \theta, \qquad \text{otherwise.} \end{cases} \tag{16}$$

Fig. 8 The decentralized local broadcast key establishment scheme

$U = \{\text{Set of guards}\}, \qquad S = \{\text{Set of nodes}\}$
U : Broadcast $\{FK_i \| (X_i, Y_i) \| H^{n-q}(PW_i) \| q\}_{K_0}$.
S : Verify $H(H^{n-q}(PW_i)) = H^{n-q+1}(PW_i)$, $\forall\, g_i \in GH_s$.
S : Broadcast $ID_{s_i} = \{ID_{g_1} \| ID_{g_2} \| \ldots \| ID_{g_m} \| th_{s_i}\}_{K_0}$, where $m = |GH_s|$.
for all $s_i \in S$
  for all ID_{s_j} heard by s_i
    if $|\bigcap(ID_{s_i}, ID_{s_j})| \geq th_{s_i,s_j}$, Generate: $K_{s_i,s_j} = H(FK_1 \| FK_2 \| \ldots \| FK_m)$
      s_i: $\{\eta_i\}_{K_{s_i,s_j}} \rightarrow s_j$ $s_j : \{J(\eta_i)\}_{K_{s_i,s_j}} \rightarrow s_i$
      if $J(\eta_i)$ valid $\rightarrow N_{s_i} = N_{s_i} \cup \{s_j\}$ end if
    end if
  end for
end for
for all $s_i \in S$
  for all $s_j \in N_{s_i}$
    Send $s_i : \{K_{s_i}\}_{K_{s_i,s_j}}$
  end for
end for

The algorithm in Fig. 8 summarizes our decentralized local broadcast key establishment scheme. In the local threshold computation, each node individually determines its own threshold (a parameter directly related to the success in preventing wormholes) based on the number of guards it hears. However, during the wormhole attack, a node may hear a much higher number of guards compared to its neighbors. In such a case, the node under attack can be misled to compute a threshold value that cannot be met by any of its one-hop neighbors and, hence, be disconnected from, the rest of the network. To address this problem, using our method, the node first detects if it is under wormhole attack. If a wormhole is detected, the node uses a mechanism called Closest Guard Algorithm (CGA) described in Section 5.3 to separate the one-hop guards from the replayed ones. Once the one-hop guards have been determined, the node selects the threshold value based on the guards that are directly heard.

4.3.4. Key establishment using directional antennas.

In Fig. 6(b), we showed how the omnidirectionality of the guards' antennas allows non-immediate neighbors to have more than *th* fractional keys in common. In order to avoid the distribution of the same fractional keys to nodes located more than one-hop away, guards may be equipped with directional antennas.

Each guard has M directional antennas with sectors being $\frac{2\pi}{M}$ wide. At each sector, guards transmit different fractional keys. However, guards include the same hash value of the hash chain to all M messages transmitted at the different antenna sectors. The use of the same hash value in all sectors for every periodic transmission of fractional keys will not allow an attacker to replay a message heard at another antenna sector. If a node s hears sector j of a guard g_i and an attacker replays to s a message transmitted at sector k of g_i, node s will have already received the latest published hash value of the hash chain via the directly heard sector j and will not authenticate the replay of the sector k.

In Fig. 7(b), we show the same network as in Fig. 6(b) with each guard using three directional antennas of beamwidth $\frac{2\pi}{3}$. Although nodes s_1, s_2 hear the same guards g_1–g_3, since they are located in different directions, they acquire different fractional keys. Hence, s_1, s_2 do not share sufficient number of fractional keys for the establishment of a pairwise key, even if an attacker mounts a wormhole link between s_1, s_2.

4.3.5. Communication cost of the decentralized key establishment scheme

In this section, we compute the communication cost of the decentralized LBK establishment scheme in terms of number of messages that are transmitted in the whole network as well as the number of messages transmitted individually by each node. In Step 1, guards broadcast the beacons containing the fractional keys. If U denotes the set of guards deployed in the network, the cost of Step 1 is equal to $|U|$, where $|\cdot|$ denotes the cardinality of the set.

In Step 2, every node broadcasts the identities of the guards that it heard. If S denotes the set of nodes deployed in the network, the number of broadcasts is equal to $|S|$. Once the fractional keys have been broadcasted, each node establishes pairwise keys with all their one-hop neighbors. The challenge response scheme executed for the establishment of the pairwise keys requires the exchange of two messages with each one-hop neighbor, and every node has, on average, $\rho_s \pi r^2$ neighbors. Hence, the communication cost of the challenge response scheme is equal to $2|S|\rho_s \pi r^2$.

In Step 3, every node unicasts the LBK to all its one-hop neighbors. The cost of this step is equal to $|S|\rho_s \pi r^2$

messages. Adding the cost of all three steps yields a network-wide communication cost C for the decentralized key establishment scheme equal to

$$C = |U| + |S| + 3|S|\rho_s \pi r^2. \quad (17)$$

The communication cost C_g for each guard g is equal to one message per LBK establishment (guards may periodically broadcast new fractional keys to update the current LBKs or accommodate changes in the network topology). The communication cost C_s for each node s is computed as follows: each node broadcasts one message to announce the fractional keys that it holds. In addition, each node s exchanges one message with each one-hop neighbor in order to establish a pairwise key when it initiates the key establishment, and one message when the key establishment is initiated by the one-hop neighbors. Finally, each node s needs to unicast its LBK to each of its one-hop neighbors, thus the communication cost for each node is $C_s = 3\rho_s \pi r^2 + 1$.

Note that the network-wide communication cost C and the individual node communication cost have been calculated based on the assumption that two pairwise keys are established between one-hop neighbors. If only one key is established according to (16), the network-wide communication cost reduces to $C = |U| + |S| + 2|S|\rho_s \pi r^2$, and the individual node communication cost reduces to $C_s = 2\rho_s \pi r^2 + 1$.

In the case where the guards are equipped with directional antennas, they transmit a different fractional key at each antenna sector. Hence, each guard needs to transmit $C_g = M$ messages per LBK establishment, where M denotes the number of antenna sectors at each guard. While the node communication cost C_s does not change, the network-wide communication for the case of guards equipped with directional antennas becomes $C = M|U| + |S| + 2|S|\rho_s \pi r^2$.

5. Securing the broadcast of fractional keys

The LBKs prevent wormhole attacks once they have been established. However, we need to ensure that an adversary does not mount a wormhole attack during the broadcasting of the fractional keys. In this section, we provide mechanisms to secure the fractional key distribution from wormholes.

5.1. Wormhole attack against the fractional key distribution

We first show how an adversary can successfully operate a wormhole link between two nodes that are out of communication range by exploiting the fractional key distribution mechanism. Recalling that $R(> r)$ is the range of the guard, consider Fig. 9, where an adversary establishes a bi-directional wormhole link between nodes s_1, s_2, with s_1, s_2 being several hops away. In Step 1 of the decentralized LBK establishment scheme, guards broadcast their fractional keys. The adversary records all messages heard by s_1, s_2 and replays the messages heard by s_1 in the vicinity of node s_2 and messages heard by s_2 in the vicinity of s_1. After the replay, nodes s_1, s_2 have a common set of fractional keys of size $|GH_{s_1} \cup GH_{s_2}|$. Independent of the threshold value selected, s_1, s_2 will share more than th fractional keys since they hear exactly the same sets of guards.

In Step 2 of the LBK establishment scheme, the nodes s_1, s_2 will broadcast the Ids of the fractional keys that they hold. The adversary will forward those messages to both nodes, and since s_1, s_2 share more than th fractional keys, they establish a pairwise key through the wormhole link. Once the pairwise key is established, the two nodes will also share LBKs and the wormhole link will be in operation.

5.2. Detection of the wormhole attack

We now show how a node can detect a wormhole attack during the broadcast of the fractional keys using two properties: The *single message per guard/sector* property and the *communication range constraint* property.

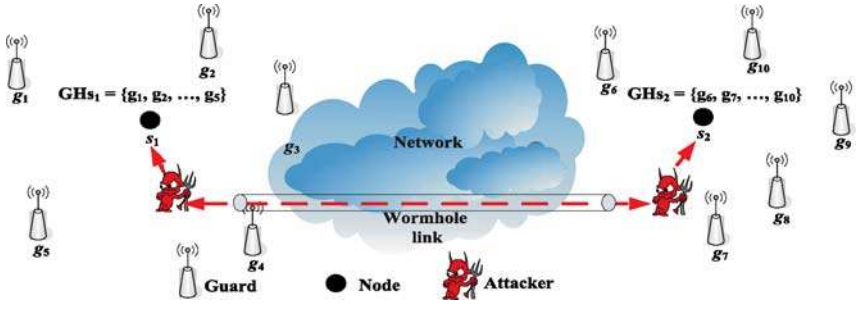


Fig. 9 A wormhole attack scenario. Node s_1 hears broadcasts from guard set $GH_{s_1} = \{g_1, \ldots, g_5\}$ and node s_2 hears broadcast from guard set $GH_{s_2} = \{g_6, \ldots, g_{10}\}$, with $GH_{s_1} \cap GH_{s_2} = \emptyset$. An attacker replays messages from GH_{s_1} in the vicinity of s_2 and messages from GH_{s_2} in the vicinity of s_1. Nodes s_1, s_2 have $|GH_{s_1} \cup GH_{s_2}| > th$ fractional keys in common and hence establish pairwise key K_{s_1,s_2}

5.2.1. Single message per guard/sector property

Claim 1. Single message per guard/sector property: Reception of multiple copies of an identical message from the same guard is due to replay or multipath effects.

Proof: When guards are equipped with omnidirectional antennas, they include a different hash value from the hash chain on every message they transmit. When guards are equipped with directional antennas, they include a different hash value on every message they transmit on the same sector. If a node receives multiple copies of an identical message. it can only be because (a) a malicious entity replays the message or (b) the message arrives multiple times due to multipath effects. If multipath effects are treated as a replay attack, then a node cannot receive the same transmission multiple times, unless it is under a replay attack.[6] □

Based on Claim 1, we can detect wormhole attacks, in case the origin point of the attack is close to the nodes under attack so that the attacker records transmissions from guards that are directly heard to the nodes under attack. Assume that guards use omnidirectional antennas for the transmission of the fractional keys. If an attacker replays a transmission of a guard g_i that is directly heard to node s, the node can detect the attack since it will have received the same fractional key through the direct link at an earlier time.

If the guards use directional antennas and the attacker replays messages from guards directly heard to the node under attack but from a different sector, the attacked node will detect that it is infeasible to hear two sectors of a single guard. Moreover, the hash values being identical for all sectors per transmission, the replay will be detected. Since the direct signal from g_i will reach s earlier than any replay, assuming that the guard transmits in all sectors simultaneously. In addition, the node will acquire the latest published value of the hash chain of g_i through the direct link. Hence, any replay containing an already published hash value will not be authenticated. Note that in the case of directional antennas a node can hear two different sectors if it located at the boundary between two sector regions due to imperfect sectorization or due to multipath effects. We also treat imperfect sectorization as a replay attack, and a node accepts the earliest received message as the authentic one.

In Fig. 10(a), A_s denotes the area where guards heard to node s are located (circle of radius R centered at s), A_0 denotes the area where guards heard at the origin point of the attack are located (circle of radius R centered at O), and the shaded area A_c denotes the common area $A_c = A_s \cap A_0$.

Claim 2. The detection probability $P(SG)$ due to the single message per guard/sector property is equal to the probability that at least one guard lies within an area of size A_c and is given by

$$P(SG) = 1 - e^{-\rho_g A_c}, \quad \text{with}$$
$$A_c = 2R^2\phi - Rl\sin\phi, \quad \phi = \cos^{-1}\frac{1}{2R}, \tag{18}$$

with l being the distance between the origin and the destination.

Proof: If a guard g_i lies inside A_c, it is less than R units away from node s and less than R units away from the origin point of the attack O. Hence, g_i will be heard by node s and its messages will be recorded by the attacker at point O. When the attacker replays the recorded messages at the destination point, node s will detect the attack due to the single message per guard/sector property. Hence, the detection probability $P(SG)$ is equal to the probability that at least one guard lies within A_c. If GH_{A_c} denotes the set of guards located within area A_c, then

$$\begin{aligned} P(SG) = P(|GH_{A_c}| \geq 1) &= 1 - P(|GH_{A_c}| = 0) \\ &= 1 - e^{-\rho_g A_c}, \end{aligned} \tag{19}$$

where A_c was previously computed in (10). □

In Fig. 11(a), we show the detection probability *P(SG)* vs. the guard density ρ_g and the distance $\|s - O\|$ between the origin point and the node under attack, normalized over R, for $\frac{R}{r} = 10$. We observe that if $\|s - O\| \geq 2R$, the single message per guard/sector property cannot be used to detect a wormhole attack since the disks A_s, A_0 do not overlap ($A_c = 0$). For distances $\|s - O\| \geq 2R$ a wormhole attack can be detected using the communication range constraint property detailed next.

5.2.2. Communication range constraint property

The set of guards GH_s heard by a node s has to satisfy the Communication Range constraint (CR). Given the coordinates of node s, all guards heard should lie within a circle of radius R, centered at s. Since node s is not aware of its location, it relies on its knowledge of the guard-to-node communication range R to verify that the set GH_s satisfies the communication range constraint.

[6] Identifying and removing multipath effects is an important problem since it will reduce the false alarm rate. We do not address this problem in this paper.

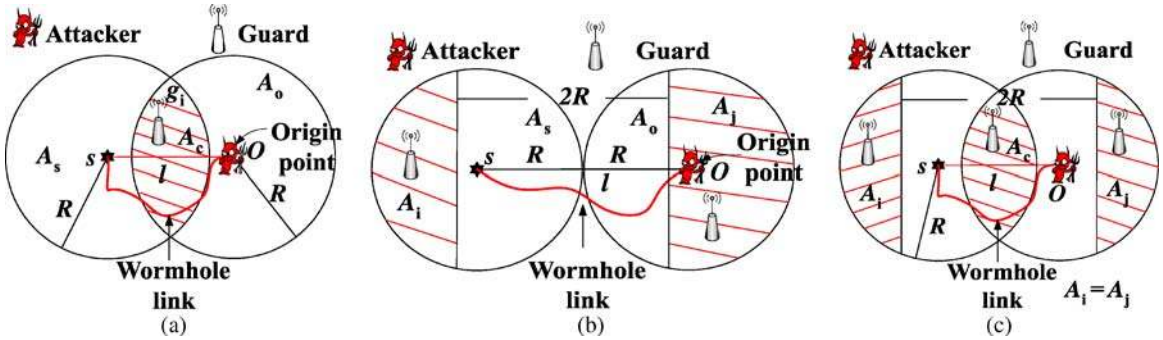


Fig. 10 (a) Single message per guard/sector property: a node s cannot receive multiple copies of the same message. (b) Communication range constraint violation: a sensor cannot hear two locators that are more than $2R$ apart, (c) Combination of the single message per guard/sector and communication range constraint property

Claim 3. Communication Range constraint property (CR): A node s cannot hear two guards $g_i, g_j \in GH_s$, that are more than 2R apart, $(i.e., \|g_i - g_i\| \leq 2R, \forall i, j, i \neq j)$.

Proof: Any guard $g_i \in GH_s$ heard by node s, has to lie within a circle of radius R, centered at the node s (area A_s in 10(a)), $\|g_i - s\| \leq R, \forall i \in GH_s$. Hence, there cannot be two guards within a circle of radius R, that are more than $2R$ apart.

$$\begin{aligned}\|g_i - g_j\| &= \|g_i - s + s - g_j\| \\ &\leq \|g_i - s\| + \|s - g_j\| \leq R + R = 2R. \end{aligned} \quad (20)$$

□

Recall that guards include their coordinates with every transmission of fractional keys and, hence, a node s knows the location of all the guards $g_i \in GH_s$. Using the guards' coordinates, a node can detect a wormhole attack if the communication range constraint property is violated. We now compute the probability *P(CR)* of detecting a wormhole attack using the communication range constraint property.

Claim 4. A wormhole attack is detected using the communication range constraint property, with a probability

$$P(CR) \geq \left(1 - e^{-\rho_g A_i^*}\right)^2, \text{ with}$$

$$A_i^* = d\sqrt{R^2 - d^2} - R^2 \tan^{-1}\left(\frac{d\sqrt{R^2 - d^2}}{d^2 - R^2}\right) \text{ and}$$

$$d = \frac{||s - O||}{2} \quad (21)$$

Proof: Consider Fig. 10(b), where $||s - O|| = 2R$. If any two guards within A_i, A_j have a distance larger that $2R$, the attack is detected. Though $P(CR)$ is not easily computed analytically, we can extract a lower bound on $P(CR)$ by considering the following event: In Fig. 10(b), the vertical lines defining shaded areas A_i, A_j are perpendicular to the line connecting s, O and have a separation $2R$. If there is at least one guard in the shaded area A_i and at least one guard in the shaded area A_j, then $\|g_i - g_j\| > 2R$ and the attack is detected. Note that this event does not include all possible locations of guards for which $\|g_i - g_i\| > 2R$ and, hence, it yields a lower bound.

$$\begin{aligned} P(CR) &= P(\|g_i - g_i\| > 2R, g_i, g_j \in GH_s) \\ &\geq (CR \cap (|GH_{A_i}| > 0 \cap |GH_{A_j}| > 0)) \quad (22) \\ &= P(CR|(|GH_{A_i}| > 0 \cap |GH_{A_j}| > 0))P(|GH_{A_i}| \\ &\quad > 0 \cap |GH_{A_j}| > 0)) \quad (23) \\ &= P(|GH_{A_i}| > 0 \cap |GH_{A_j} > 0|)) \quad (24) \\ &\quad \times \left(1 - e^{-\rho_g A_i^*}\right)^2 \left(1 - e^{-\rho_g A_j^*}\right), \quad (25) \end{aligned}$$

where (22) follows from the fact that the probability of the intersection of two events is always less or equal to the probability of one of the events, (23) follows from the definition of the conditional probability, (24) follows from the fact that when $|GH_{A_i}| > 0 \cap |GH_{A_j} > 0$, we always have a communication range constraint violation $(P(CR|(|GH_{A_i}| > 0 \cap |GH_{A_j}| > 0)) = 1)$, and (25) follows from the fact that A_i, A_j are disjoint areas and that guards are randomly deployed (probability of finding k guards within some area follows the Poisson distribution).

We can maximize the lower bound of $P(CR)$ by finding the optimal values A_i^*, A_j^*, depending on the distance $\|s - O\|$. In the Appendix B we prove that the lower bound in

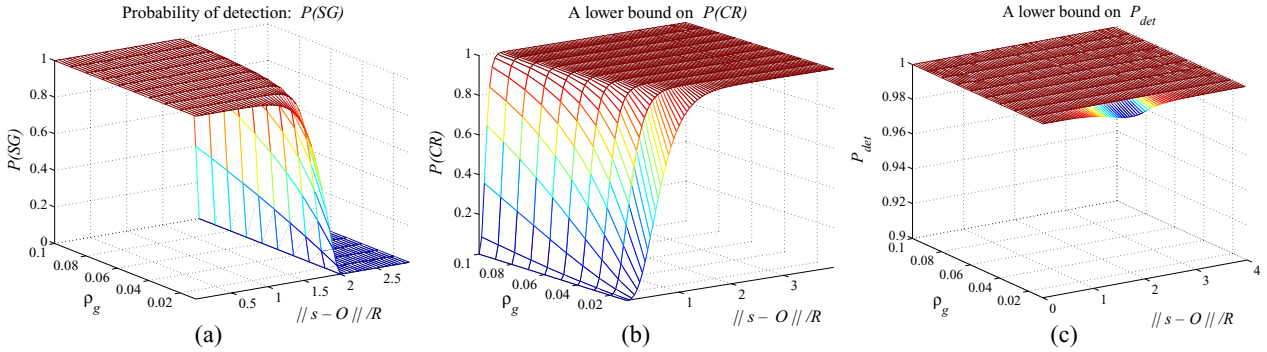


Fig. 11 Wormhole detection probability for $\frac{R}{r} = 10$ based on: (a) single message per guard/sector property: $P(SG)$, (b) Communication range constraint property, a lower bound on $P(CR)$, (c) and a lower bound on the wormhole detection probability P_{det}

(25) attains its maximum value when $A_i^*, = \max_i\{A_i\}$ subject to the constraint $A_i = A_j$ (A_j, A_j are symmetric). We also prove that A_i^*, is expressed by

$$A_i^*, A_j^* = \sqrt{R^2 - d^2} - R^2 \tan^{-1}\left(\frac{d\sqrt{R^2 - d^2}}{d^2 - R^2}\right), \quad \text{and}$$

$$d = \frac{\|s - O\|}{2}. \tag{26}$$

Substituting (26) to (25) yields the required result, $P(CR) \geq (1 - e^{-\rho_g A_i^*})^2$. □

In Fig. 11(b), we show the maximum lower bound on $P(CR)$ vs. the guard density ρ_g and the distance $\|s - O\|$ normalized over R. The lower bound on $P(CR)$ increases with the increase of $\|s - O\|$ and attains its maximum value for $\|s - O\| = 4R$ when $A_i^* = A_j^* = \pi R^2$. For distances $\|s - O\| > 4R$, a wormhole attack is always detected based on the communication range constraint property, since any guard within A_0 will be more than $2R$ apart from any guard within A_s.

5.2.3. *Detection probability P_{det} of the wormhole attack*

We now combine the two detection mechanisms, namely the single message per guard/sector property and the communication range constraint property, for computing the detection probability of a wormhole attack during the broadcast of the fractional keys.

Claim 5. The detection probability of a wormhole attack during the broadcast of fractional keys is lower bounded by $P_{det} \geq (1 - e^{-\rho_g A_c}) + (1 - e^{-\rho_g A_i^*})^2 e^{-\rho_g A_c}$.

Proof: In the computation of the communication range constraint property, by setting $A_i = A_j$ and maximizing A_i regardless of the distance $\|s - O\|$, the areas A_i, A_j, and A_c do not overlap as shown in Fig. 10(c). Hence, the corresponding events of finding a guard at any of these areas are independent, and we can derive a lower bound on the detection probability P_{det} by combining the events.

$$\begin{aligned} P_{det} &= P(SG \cup CR) \\ &= P(SG) + P(CR) - P(SG)P(CR) \\ &= P(SG) + P(CR)(1 - P(SG)) \\ &\geq \left(1 - e^{-\rho_g A_c}\right) + \left(1 - e^{-\rho_g A_i^*}\right)^2 e^{-\rho_g A_c}. \end{aligned} \tag{27}$$

The quantity in (27) is a lower bound on P_{det} since we used the lower bound on $P(CR)$. □

In Fig. 11(c), we show the lower bound on P_{det} vs. the guard density ρ_g and the distance $\|s - O\|$ normalized over R. For values of $\|s - O\| > 4R$, $P(CR) = 1$, and, hence, a wormhole attack is always detected. From Fig. 11(c), we observe that a wormhole attack during the distribution of the fractional keys is detected with a probability very close to unity, independent of where the origin and destination point of the attack are located. The intuition behind (27) is that there is at most $(1 - P_{det})$ probability for a specific realization of the network, to have an origin and destination point where a wormhole attack would be successful. Even if such realization occurs, the attacker has to acquire full knowledge of the network topology and, based on the geometry, locate the origin and destination point where the wormhole link can be established.

5.3. Key establishment in the presence of wormholes

Although a wormhole can be detected using the two detection mechanisms, a node under attack cannot distinguish the valid subset of guards from the replayed ones. Once a wormhole is detected, there needs to be an additional mechanism to identify the set of guards directly heard to the node, from those replayed. We now describe the *Closest Guard Algorithm (CGA)* that resolves the guard ambiguity.

GH'_s : Guards heard by node s
s : **broadcast** $\{CGR\_REQ \| \eta_s\}_{K_0} \| ID_s$
forall $g_i \ni \|g_i - s\| \leq r(D_g)^{\frac{1}{\gamma}}$
g_i : **broadcast** $\{(X_i, Y_i) \| J(\eta_s) \| H^{n-k}(PW_i)\}_{K_{s,g_i}} \| ID_{g_i}$
endfor
s : **identify** $g'_i \in GH'_s$ that replies first with the correct $J(\eta_s)$
s : **set** $GH_s : \{g_i \in GH'_s \ni \|g'_i - g_i\| \leq 2R\}$

Fig. 12 The pseudo-code for the Closest Guard Algorithm (CGA). A node under a wormhole attack uses the CGA to separate the valid set of guards (one-hop) from the replayed ones

Closest Guard Algorithm (*CGA*)*:* Assume that a node s authenticates a set of guards GH'_s, but detects that it is under attack. To determine the valid set of guards (guards within one hop from s), node s executes the following three-step algorithm:

Step 1. The node s broadcasts a message containing a Closest Guard Reply Request *CGR_REQ* and a nonce η_s encrypted with the globally shared key K_0, and its ID_s concatenated at the end of the encrypted part of the message. The message format of the request transmitted by sensor s is as follows

$$\{CGR\_REQ \| \eta_s\}_{K_0} \| ID_s.$$

Step 2. Every guard hearing the message broadcasted from s replies with a message containing $J(\eta_s)$, where $J(x)$ is a computationally efficient function, such as $J(x) = x - 1$, its coordinates, the next hash value of its chain that has not been published, and its ID_g. The message is encrypted using the pairwise key K_{s,g_i}, shared between the sensor s and each guard g_i. The message format broadcasted by each guard g_i hearing the sensor's request is as follows

$$\{(X_i, Y_i) \| J(\eta_s) \| H^{n-k}(PW_i)\}_{K_{s,s_i}} \| ID_{g_i}.$$

The node identifies the guard g'_i, whose reply arrives first as the closest guard to s.

Step 3. Using the communication range constraint property, node s identifies the valid set of guards GH_s as all the guards that are not more than $2R$ away[7] from g'_i and uses the fractional keys received from GH_s to establish pairwise keys and LBKs with its immediate neighbors. Figure 12 summarizes the steps of the *CGA* algorithm. Note that in order for a node s to identify its closest guard, we assume that no packet loss occurs during the execution of the CGA.

To execute CGA, a node must be able to communicate bi-directionally with at least one guard. The probability $P_{s\to g}$ of a node having a bi-directional link with at least one guard can be computed as shown in Fig. 13(b). From $P_{s\to g}$, we can compute the probability P_{bd} that all nodes can bi-directionally communicate with at least one guard. In Fig. 13(b), we show P_{bd} and the conditions on ρ_g, D_g, and r, so that every node has a bi-directional link with at least one guard with probability very close to unity and, hence, resolve any ambiguity in determining the valid set of guards heard. Derivation of the relations is provided at the Appendix A.

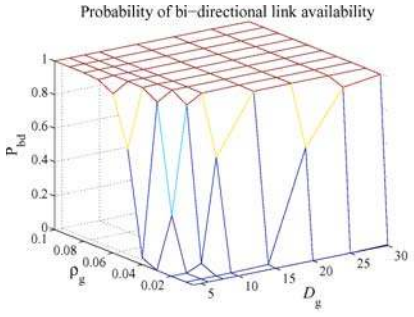
An additional implementation issue with the CGA algorithm involves collisions of multiple *CGA_REQ* messages at the guards and collisions of multiple replies at the nodes. Known techniques for multiple access of the same medium, such as CSMA protocols [2] and/or CDMA mode of communication [40] can be employed to enable the use of the same medium by multiple users. To mitigate the effect of collisions at the guards, nodes may randomize the time of broadcasting the *CGA_REQ* messages. Note that just a few nodes that are under attack need to execute the CGA algorithm, unless the adversary performs a large scale wormhole attack by deploying multiple wormhole links to attack many nodes at once.

For the case of collisions of replies originating from guards occurring at the node side, note that although a node may hear several guards, it can only bi-directionally communicate with a small fraction of the guards it hears, since regular nodes have a much smaller communication range than guards. In fact, in our deployment, bi-directional communication with only one guard is sufficient to resolve the ambiguity between the valid set of guards and the replayed one. Hence, not many guards (if more than one) will reply to the node's request. Moreover, in order to provide a valid response from the replayed set of one-hop guards, an adversary needs to (a) record the *CGA_REQ* transmitted by the node, (b) tunnel it via the wormhole link at the origin point of the attack, (c) replay it at the origin point of the attack, (d) record the guards reply, (e) tunnel the reply via the wormhole link to the destination point of the attack, and (f) replay the guards' reply at the destination point. However, any replies from the replayed guards will arrive at the node much later than the reply originating from the one-hop guards.[8] Hence, the replies provided by the attacker will not collide with the one provided by the closest guard.

In the case where no additional mechanism exists to resolve collisions, the node can engage in a challenge-response

[7] In the case where the guards are equipped with directional antennas, node s identifies the valid set of guards GH_s as all the guards whose sectors overlap with the sector of the closest guard g'_i.

[8] Note that we have assumed that the adversary does not jam the communication medium.



(a)

$$P_{s\to g} = 1 - e^{-\rho_g \pi r^2 (D_g)^{\frac{2}{\gamma}}}$$

$$P_{bd} = \left(1 - e^{-\rho_g \pi r^2 (D_g)^{\frac{2}{\gamma}}}\right)^{|S|}$$

$$r(D_g)^{\frac{1}{\gamma}} \geq \frac{\sqrt{-\ln\left(1 - \sqrt[|S|]{P_{bd}}\right)}}{\pi \rho_g}$$

$$\rho_g \geq \frac{\sqrt{-\ln\left(1 - \sqrt[|S|]{P_{bd}}\right)}}{\pi r^2 (D_g)^{\frac{2}{\gamma}}}$$

(b)

Fig. 13 (a) Probability that all nodes have a bi-directional link with, at least one guard for $r = 4.D_g$ denotes the guard antenna directivity, (b) $P_{s\to g}$, and P_{bd}, necessary ρ_g, D_G, r, so that every node has a bi-directional link with probability P_{bd}

protocol with each guard within the set GH'_s, such as the one in [21]. In order to compare the distances between different guards, the node needs to be equipped with an accurate timer, so that it can measure the round-trip-time (RTT) in the challenge-response exchange. Using the RTT from the challenge-response for different guards, the node can identify the closest guard and, hence, the valid set of guards. In our present scheme, nodes are not required to be equipped with such accurate timers (that was the reason why the CGA was proposed as opposed to a method that uses timers). However, if nodes can be equipped with timers, the node can also reject any reply that has an RTT longer than $2\frac{r(D_g)^{\frac{1}{\gamma}}}{c} + \delta$, where $r(D_g)^{\frac{1}{\gamma}}$ denotes the node-to-guard communication range, c denotes the speed of light, and δ denotes an upper bound on the guard processing delay. Hence, the node can verify that any reply with a RTT smaller than $2\frac{r(D_g)^{\frac{1}{\gamma}}}{c}$ comes from a guard within its range and can reject those replies taking more than $2\frac{r(D_g)^{\frac{1}{\gamma}}}{c} + \delta$.

5.4. Locating the origin point of the attack

While we have shown that we can detect and prevent the wormhole attack with a probability very close to unity by choosing appropriate parameters, we now show that our *CGA* algorithm can identify the region of the origin point of the attack. Assume that a node s is under a wormhole attack and hears a set of guards GH'_s. By executing the *CGA*, node s identifies the valid set of guards GH_s as well as the replayed set of guards $GH^r_s = GH'_s - GH_s$. Using the coordinates of the guards in GH^r_s, node s can identify the region where the attacker recorded the replayed information (i.e., the region of the origin point of the attack). Since the attacker has to be within distance R from every guard that it records, the origin point has to be within the intersection of the communication areas of all the guards in GH^r_s. The node under attack identifies the region where the attacker should be located by computing the overlapping region of the disks that define the communication areas of each guard being replayed.

The CGA algorithm only provides the capability for a node to identify the origin point of a wormhole attack. In order to provide an intrusion detection system (IDS) for the wormhole attack, additional relevant problems, such as identification of the destination point of the attack, event reporting, and verification of the validity of the reports, need to be addressed. We do not address these problems in this paper.

6. Performance evaluation

In this section, we provide simulation studies that evaluate to what extent our method prevents the wormhole attack. For varying network parameters, we evaluate the percentage of one-hop neighbors that are able to establish a pairwise key and, hence, a local broadcast key, as a function of the threshold *th*. We also evaluate the percentage of non-immediate neighbors that have more than *th* fractional keys in common, as a function of *th*. Finally, we show that in the case where it is possible to establish a wormhole link, that link is no longer than two hops, and based on our simulation results, we provide the rationale to determine the appropriate threshold value to establish LBK for each network setup.

6.1. Simulation setup

We generated random network topologies confined in a square area of size $\mathcal{A} = 10{,}000m^2$. For each network topology we randomly placed 5,000 nodes within $\mathcal{A}$, equivalent to a node density of $\rho_s = 0.5$ nodes/m^2 We then randomly placed the guards with density ρ_g, varying from 0.005 to 0.05 guards/m^2. To ensure statistical validity, we repeated each experiment for 1,000 networks and averaged the results.

Since the level of protection against wormholes depends upon the guard density ρ_g, we want to maintain a constant density across the whole network deployment area. However, if we deploy guards in the same area as the nodes of the network, nodes located at the border of the deployment area will experience a smaller guard density than nodes in the center of the area. To eliminate the border effects, we need to over-deploy guards at the borders of the borders of the deployment area or deploy guards at a slightly larger area than the area of the nodes.

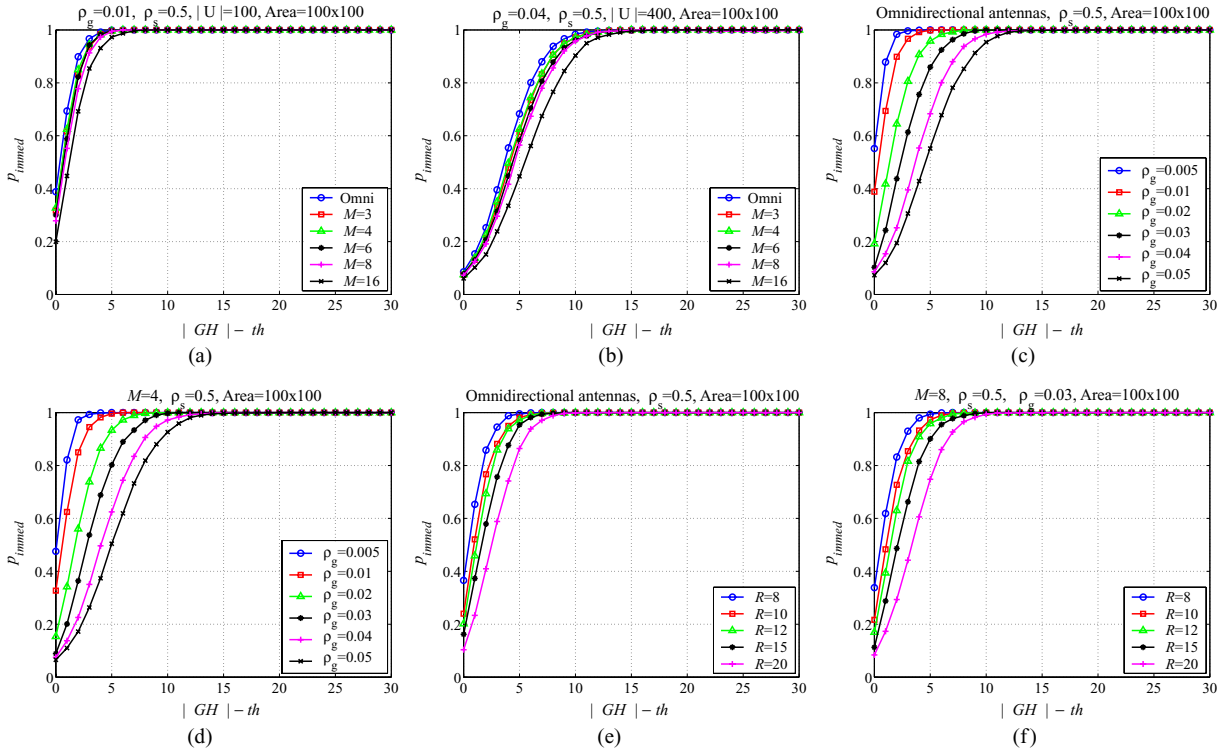


Fig. 14 Percentage of immediate neighbors that share more than th fractional keys for $\rho_s = 0.5$ nodes/m^2, $\mathcal{A} = 10{,}000\ m^2$ when, (a) different antennas are used at the guards and $\rho_g = 0.01$ guards/m^2, (b) different antennas are used at the guards and $\rho_g = 0.04$ guards/m^2, (c) omnidirectional antennas are used at the guards and ρ_g varies, (d) 4-sector directional antennas are used at the guards and ρ_g varies, (e) omnidirectional antennas are used at the guards, $\rho_g = 0.03$ guards/m^2 and R varies and, (f) 8-sector antennas are used at the guards, $\rho_g = 0.03$ guards/m^2, and R varies

To illustrate how deploying guards at a larger area can address the border effects issue, assume that nodes are to be deployed in a square of size $\mathcal{A} = A \times A$. In order to provide the same level of security at the borders as in the inside of the deployment area, we randomly deploy guards within a square of size $(A + R) \times (A + R)$, where R is the guard-to-node communication range. The number of guards that need to be over-deployed in order to eliminate the border effects is equal to $G_{over} = \rho_g(R^2 + 2AR)$. In our performance evaluation, we simulated the constant deployment density by deploying guards in the area $(A + R) \times (A + R)$ and nodes in the area $A \times A$.

In addition, as described in Section 4.3.3, we allowed each node s to locally compute the threshold based on the number of guards $|GH_s|$ that it hears. Hence, depending on $|GH_s|$, each node selects a different threshold value equal to $th = |GH_s| - c$, where c is some constant value. Our simulation graphs provide a mechanism to choose the appropriate value for the constant c, in order to maximize the probability of key establishment with one-hop neighbors, while keeping the probability of sharing more than the threshold keys with non-immediate neighbors below a desired value. In order to refer all results to a common axis, we use $|GH_a| - th$ instead of th.

6.2. Key establishment with one-hop neighbors

In our first experiment, we evaluated the percentage of one-hop (immediate) neighbors p_{immed} that each node is able to establish a pairwise key with, as a function of the threshold th, the guard density ρ_g and the number of antenna sectors M used by the guards. In Fig. 14(a), we present p_{immed} vs. $|GH_s| - th$, for a guard density $\rho_g = 0.01$ guards/m^2 and for different antennas sectors. We observe that for a threshold value $th \leq |GH_s| - 5$, the nodes establish a pairwise key with almost all their neighbors when omnidirectional or seciored antennas with M-3,4,6,8 sectors are used ($p_{immed} > 0.99$). For $M = 16$ we achieve[9] a $p_{immed} > 0.99$ for threshold values smaller than $th \leq |GH_s| - 7$.

[9] In today's technology, it may seem excessive to assume that guard nodes have 16 antennas each. However, as the frequency used for communication increases, the size of the antennas will decrease and, hence, in the near future it will be feasible to install more directional antennas in a single guard. Furthermore, the use of multiple-array patched antennas (antennas integrated on a chip) has enabled the implementation of directional antennas of very small factor. The goal of simulating such a high number of antennas at the guards is to explore the tradeoff between hardware complexity and level of security.

Note that the use of directional antennas does not significantly affect the threshold value for which nodes are able to establish pairwise keys with their immediate neighbors. This fact is an indication that immediate neighbors hear the same antenna sectors and, hence, acquire the same fractional keys. However, when directional antennas are used, less neighbors more than one-hop away will share more than *th* fractional keys as we will show in our second experiment.

In Fig. 14(b), we present p_{immed} vs. $|GH_s| - th$ for a higher guard density $\rho_g = 0.04$ guards/m^2. We observe that for $\rho_g = 0.04$ guards/m^2 we need a threshold value $th \leq |GH_s| - 13$ to allow all one-hop neighbors to establish pairwise keys. Since for $\rho_g = 0.04$ guards/m^2 each node hears almost four times more guards than for $\rho_g = 0.01$ guards/m^2, more guards are likely to be heard only to a fraction of the local neighborhood rather than the whole. Hence, we need a threshold value significantly lower than GH_s to allow all immediate neighbors to share a sufficient number of fractional keys for establishing a pairwise key. To further reinforce this fact, in Figs. 14(c) and (d) we present p_{immed} vs. $|GH_s| - th$, for varying guard densities ρ_g, and for omnidirectional and 4-sector directional antennas, respectively. We observe that from $\rho_g = 0.005$ guards/m^2 to $\rho_g = 0.05$ guards/m^2 we need to increase the $|GH_s| - th$ by 10 in order to achieve the same p_{immed}.

In Figs. 14(e) and (f), we present p_{immed} vs. $|GH_s| - th$ for varying guard-to-node communication ranges R, for omnidirectional and eight-sector directional antennas, respectively. We observe that as the communication range R increases we need a higher difference $|GH_s| - th$ in order to achieve the same p_{immed}. This is due to the fact that as R increases, each node is able to hear more guards (same effect as increasing the guard density ρ_g). Hence, out of the bigger set of possible guards heard, more guards are heard only to a fraction of the local neighborhood, and a lower threshold value relative to $|GH_s|$ is needed to allow all immediate neighbors to share more than *th* fractional keys.

6.3. Isolation of non-immediate neighbors

In order to prevent wormhole attacks, we must ensure that non-immediate neighbors remain isolated by not being able to establish a pairwise key. In our second experiment, we evaluated the percentage of non-immediate neighbors p_{non-im} that share more than *th* fractional keys as a function of *th*, for different guard densities ρ_g and number of antenna sectors M. For each node, we took into account in the percentage calculation only those neighbors that heard at least one common guard with the node under consideration.

In Fig. 15(a), we show p_{non-im} vs. $|GH_s| - th$ in a logarithmic scale for a guard density of $\rho_g = 0.01$ guards/m^2. From Fig. 15(a), we observe that the use of directional antennas can drop the p_{non-im} up to half compared to the omnidirectional antennas case, at the expense of hardware complexity at the guards. For example, for a threshold value $th = |GH_s| - 3$, $p_{non-im} = 0.0358, 0.0280, 0.0252, 0.0236, 0.0197, 0.0118$ for $M = 1, 3, 4, 6, 8, 16$ antenna sectors, respectively. In Fig. 15(b), we present p_{non-im} vs. $|GH_s| - th$ for a guard density $\rho_g = 0.04$ guards/m^2. We observe that for a higher guard density we are able to further limit the number of non-immediate neighbors that share more than *th* fractional keys. For example, when $th = |GH_s| - 10$, $p_{non-im} = 0.0117, 0.091, 0.089, 0.0079, 0.0068, 0.004$ for $M = 1, 3, 4, 6, 8, 16$ antenna sectors, respectively.

In Figs. 15(c) and (d) we present p_{non-im} vs. $|GH_s| - th$ for varying guard densities and show how we achieve higher isolation of non-immediate neighbors with the increase of ρ_g. In Fig. 15(e), we present p_{non-im} for different guard-to-node communication ranges R and show how we achieve higher isolation of non- immediate neighbors with the increase of R. As expected, a higher guard density ρ_g and a higher R achieve better non-immediate neighbor isolation for all values of the threshold *th*, since for both cases the set of guards heard at each node becomes bigger and more guards are only heard to a fraction of the non-immediate neighbors.

6.4. Length of a potential wormhole link

Our simulation results confirmed that by choosing appropriate network parameters, namely guard-to-node communication range R, guard density ρ_g, and number of directional antennas M, we can eliminate wormhole links with a very high probability. An adversary would have to gain a global view of the network topology by knowing all the locations of the nodes and the guards in order to identify, if any, a potential origin and destination point to launch its attack. In this section, we show that even in the case where that adversary does identify two points to launch his attack, the length of the wormhole link established is not longer than two hops. In fact, any non-immediate neighbors that share more than *th* fractional keys are located just outside the perimeter that defines their node-to-node communication range r.

In Fig. 15(f), we show the average distance normalized over r, between non-immediate neighbors that have in common more than *th* fractional keys. We observe that for threshold values lower than $th \leq |GH_s| - 10$, all non-immediate neighbors that share sufficient fractional keys are no more than two hops away, regardless of the number of directional antennas used at the guards. As the threshold increases towards its maximum value $|GH_s|$, the length of any potential wormhole link becomes smaller. For example, by examining Figs. 15(b) and (f), for 16-sector directional antennas and $th = |GH_s| - 5$, an attacker has a $p_{non-im} = 0.0004$ probability to establish a wormhole link between two non-immediate neighbors and that the link is $1.05r$ long.

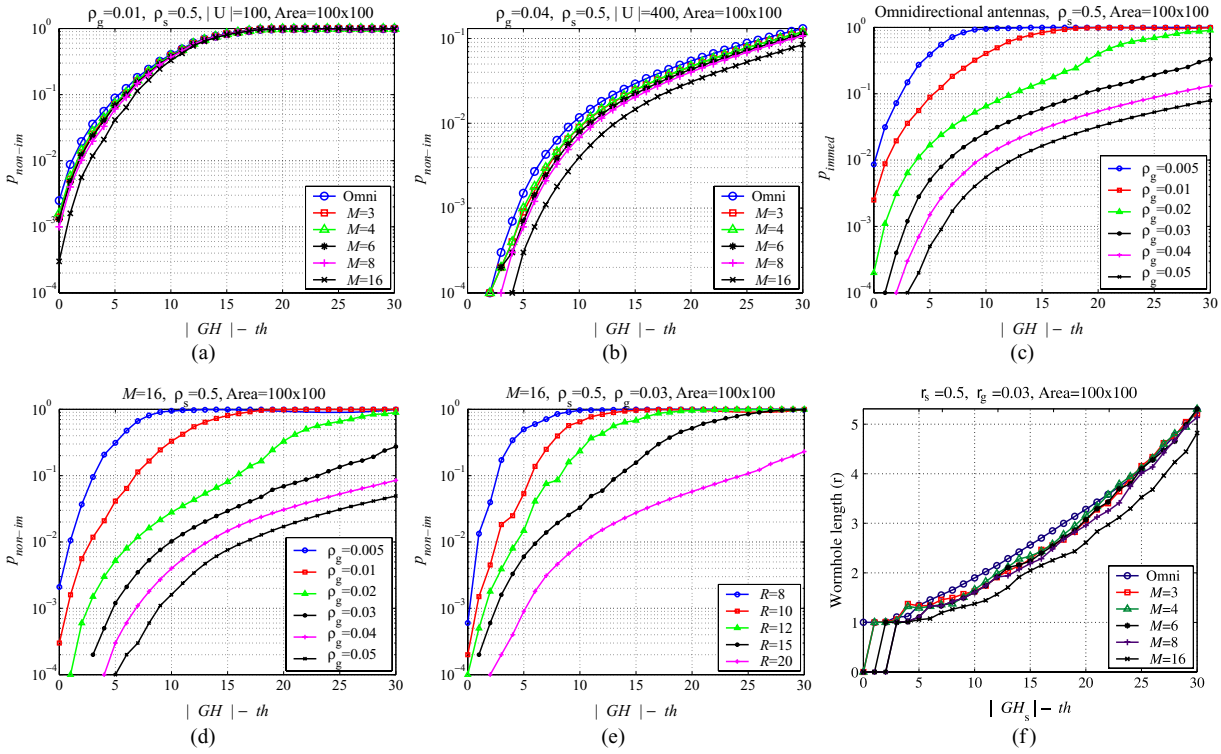


Fig. 15 Percentage of non-immediate neighbors that share more than *th* fractional keys for $\rho_s = 0.5$ nodes/m^2, $\mathcal{A} = 10{,}000m^2$ when (a) different antennas are used at the guards and $\rho_g = 0.01$ guards/m^2, (b) different antennas are used at the guards and $\rho_g = 0.04$ guards/m^2, (c) omnidirectional antennas are used at the guards and ρ_g varies, (d) 16-sector directional antennas are used at the guards and ρ_g varies, (e) 16-sector directional antennas are used at the guards, $\rho_g = 0.03$ guards/m^2, and R varies and, (f) average distance in number of hops between non-immediate neighbors that share more than *th* fractional keys

The worst case result of our approach allows the establishment of two-hop wormhole links with a very small probability. Those wormhole links can be a disruption for the nodes around the destination point. However, the impact of such wormholes is localized in the two-hop neighborhood around the destination point of the wormhole attack and does not affect the whole network. To illustrate this, consider a wormhole attack against a distance vector-based routing protocol as shown in Fig. 1(a) of Section 2. If a wormhole link is established between nodes s_3 and s_4, no traffic will be affected except for the messages directed from s_3 to s_4. On the other hand, if a wormhole link is established between nodes s_6 and s_9, all traffic that is passing through the vertex cut between s_6 and s_9 will be controlled by the attacker. While in our simple example the minimum cut between nodes s_1– s_7 and s_9–s_{13} consists of only one edge, in real network deployment scenarios the minimum cut is expected to have a much bigger size, due to the high network density and size.[10]

Another possible effect of a short wormhole is to disrupt the communication of certain *key nodes* of the network. As previously noted in the paper, a two-hop wormhole can force a single node to route through the wormhole link and give the attacker the advantage to control the traffic flow from/to that node. Our scheme does not prevent this type of attack. However, we anticipate that the operation of ad hoc networks that are envisioned to operate in a decentralized manner will not be dependent upon the existence of a single or a small number of "key nodes" that can be easily targeted by an attacker. Instead, the network operation will depend on the cooperation principle of an abundance of densely deployed devices with similar capabilities. If the network operation relies on the existence of few key nodes, the adversary can significantly disrupt the network by launching a variety of attacks, such as DoS attacks, since a key node is a single point of failure.

Finally, as an example, short wormholes are not a major network disruption in majority-based event-driven applications such as the one described in the Fig. 2 of Section 2. Revisiting the example of temperature monitoring, a clusterhead triggers an alarm if the majority of one-hop neighbors reports a temperature measurement greater than a threshold. In the case of a short wormhole, one can anticipate that

[10] Having a minimum cut of very few edges leaves the network vulnerable to many types of attacks such as DoS attacks, and node capture attacks, since it allows the adversary to concentrate its attack on a very small part of the network.

nodes located within a two-hop range from the clusterhead will not have significantly different temperature readings compared to the nodes within the one-hop range. Furthermore, the number of nodes located within the ring between the circles of radius r and $1.05r$ centered at the clusterhead is significantly smaller compared to the number of nodes located within the disk of radius r centered at the clusterhead ($[\rho_s \pi (1.05r^2 - r^2)] = 0.0625 \rho_s \pi r^2$) and, hence, even if the measurements of the two-hop nodes are greater than the threshold, they cannot overcome the majority of the measurements originating from nodes within the communication range r. As an example, if $r = 10m$ and $\rho_s = 0.05$ nodes/m^2, then there are 15.7 nodes on average within one hop from the clusterhead, while only 1.6 nodes on average exist between r and $1.05r$ from the clusterhead.

6.5. Determining the threshold value

For different system parameters, combining the plots for immediate and non-immediate neighbors, we can determine what is the appropriate threshold value to achieve both isolation of non-immediate neighbors, and allow one-hop neighbors to establish pairwise keys. For example, when $\rho_g = 0.01$ guards/m^2, from Figs. 14(a) and 15(a), a threshold of $th = |GH_s| - 4$ isolates 97.91% of the non-immediate neighbors, while allowing 93.13% of one-hop neighbors to establish pairwise keys, when $M = 16$. From Figs. 14(b) and 15(b), a threshold of $th = |GH_s| - 14$ isolates 99.996% of the non-immediate neighbors, while allowing 98.64% of the immediate neighbors to establish pairwise keys for $M = 16$.

Depending on the hardware complexity constraints at the guards (transmission power and number of directional antennas) and the security requirements, we can select the appropriate threshold value th to achieve the maximum connectivity to immediate neighbors. For example, if due to hardware complexity constraints only omnidirectional antennas can be used and the required non-immediate neighbor isolation is above 99%, one can achieve a $p_{immed} = 0.64$ for $\rho_g = 0.01$ when $th = |GH_s| - 2$ (see Figs. 14(a) and 15(a)). By increasing the guard density to $\rho_g = 0.04$ guards/m^2 for the same constraints, we can achieve a $P_{immed} = 0.90$ (see Figs. 14(b) and 15(b)). Hence, for any hardware constraint and security requirement, we can select the threshold value th and the network parameters, ρ_g, R, so that we maximize p_{immed}, while keeping p_{non-im} below a specific value.

6.6. Re-evaluating the system behavior under irregular radio pattern

In our simulation study up to Section 6.5 we have considered an idealized model for the communication range of both the guards and the nodes of the network. Every guard has the same communication range R and every node has the same communication range r. In this section, we study how the security parameters, namely the probability of establishing a pairwise key with a one-hop neighbor p_{immed}, the probability of sharing more than th fractional keys with a non-immediate neighbor p_{non-im}, and the length of a potential wormhole link vary, when the communication range R varies at each direction.

To simulate the variation of the communication range of each guard, we considered three different experiments. In the first experiment, each guard is equipped with an omnidirectional antenna, and for each possible direction it has a communication range R' that is randomly selected between the values of $[(1 - f)R, (1 + f)R]$, where f denotes the fraction of variation of the communication range.[11] We assigned to f the values $f : \{0, 0.1, 0.2, 0.3, 0.4, 0.5\}$. During this experiment, nodes could directly communicate with guards outside the nominal communication range R, on average, every guard heard the same number of guards $|GH_s|$ as in the case where the communication range R did not vary. Hence, the probability of establishing a pairwise key with a one-hop neighbor p_{immed}, the probability of sharing more than th fractional keys with a non-immediate neighbor p_{non-im}, and the length of a potential wormhole link did not show any variation.

In the second experiment, we biased the communication range of each guard to have smaller values than the nominal communication range R. Specifically, we assigned to each guard a communication range value randomly selected between the values of $[(1 - f)R, R]$. Hence, each node would hear, on average, a smaller number of guards compared to the case where the guard communication range was equal to R for all guards. In Fig. 16(a), we show the p_{immed} vs. the $|GH_s| - th$ for varying values of f. We observe that the probability of establishing a pairwise key with the one-hop neighbor does not vary significantly with the variation of R. This is due to the fact that the threshold is locally decided at each node and, hence, the parameter that affects the P_{immed} is the threshold relative to $|GH_s|$ and not the absolute value of GH_s. Furthermore, as we observe in Fig. 14(e), varying the value of R does not have a significant impact on p_{immed}.

In Fig. 16(b), we show the probability for two non-immediate neighbors to share more fractional keys than the threshold, vs. $|GH_s| - th$ for varying values of f. We observe that as f increases, the curves for the P_{non-im} are shifted to the left of the graph. This is essentially the same result as if we were decreasing the density of the guards (i.e., each node would hear a smaller number of guards (see Fig. 15(c))). In Fig. 16(c), we show the average distance normalized over r between non-immediate neighbors that have in common more than th fractional keys. We observe that for threshold values lower than $th \leq |GH_s| - 10$, all non-immediate

[11] A similar radio model was used for the evaluating the performance of the localization scheme in [17].

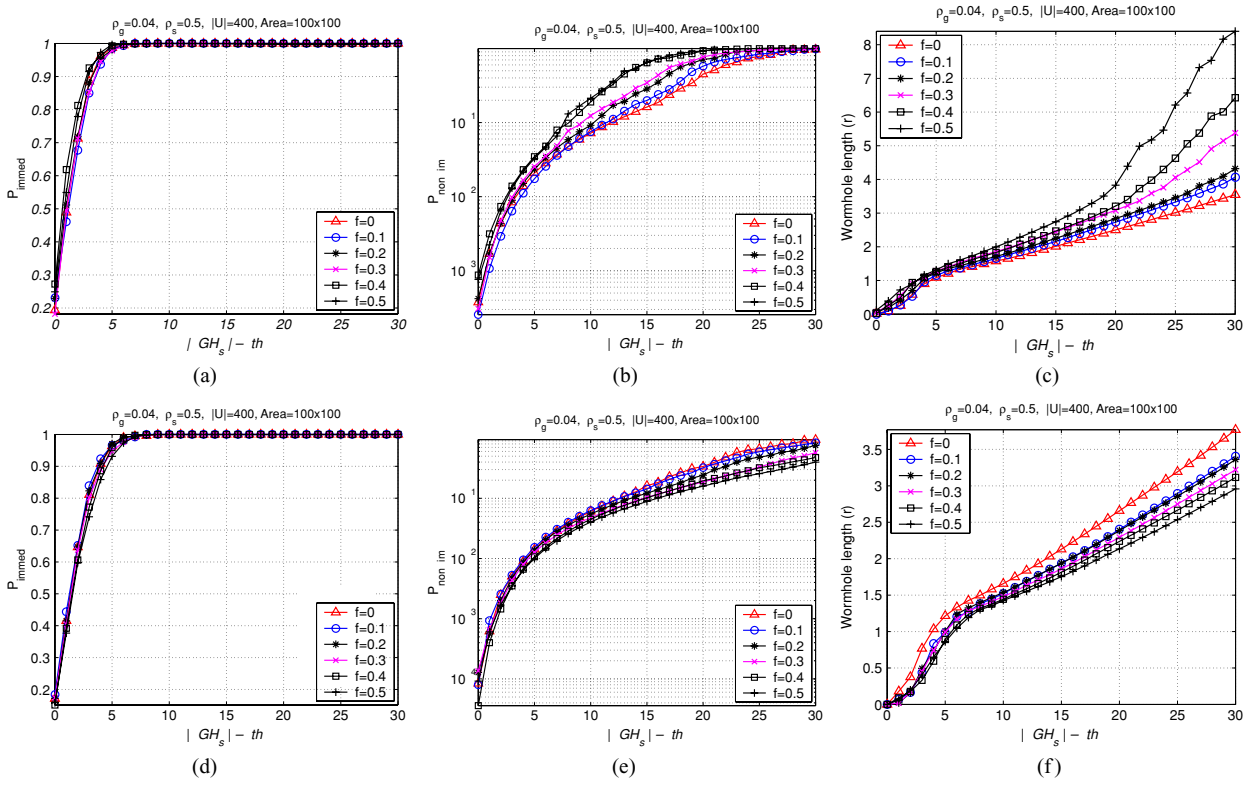


Fig. 16 Network parameter values: $\rho_s = 0.5$ nodes/m^2, $\rho_g = 0.04$ guards/m^2, $\mathcal{A} = 10{,}000m^2$. (a) Percentage of immediate neighbors that share more than *th* fractional keys when $R' \in [(1-f)R, R]$. (b) Percentage of non-immediate neighbors that share more than *th* fractional keys when $R' \in [(1-f)R, R]$. (c) Average distance in number of hops between non-immediate neighbors that share more than *th* fractional keys when $R' \in [(1-f)R, R]$. (d) Percentage of immediate neighbors that share more than *th* fractional keys when $R' \in [R, (1+f)R]$. (e) Percentage of non-immediate neighbors that share more than *th* fractional keys when $R' \in [R, (1+f)R]$. (f) Average distance in number of hops between non-immediate neighbors that share more than *th* fractional keys when $R' \in [R, (1+f)R]$

neighbors that share sufficient fractional keys are no more than two hops away, for any value of the fraction *f*. We also note that when the communication range of the guards is smaller than the nominal range *R*, the average wormhole length increases (the curves of the wormhole length are shifted to the left). This is due to the fact that as the fraction *f* increases, each node hears, on average, a smaller number of guards. Hence, it is more probable that two nodes not within communication range have in common a smaller number of fractional keys.

In the third experiment, we biased the communication range of each guard to have higher values than the nominal communication range *R*. Specifically, we assigned to each guard a communication range value randomly selected between the values of $[R, (1+f)R]$. Hence, each node would hear, on average, a higher number of guards compared to the case where the guard communication range was equal to *R* for all guards. In Fig. 16(d), we show the P_{immed} vs. the $|GH_s| - th$ for varying values of *f*. Again, the probability of establishing a pairwise key with the one-hop neighbor does not vary significantly with the variation of *R*. This result is consistent with the graph of Fig. 14(e), where the variation of *R* does not have a significant impact on P_{immed}.

In Fig. 16(e), we show the probability for two non-immediate neighbors to share more fractional keys than the threshold vs. $|GH_s| - th$ for varying values of *f*. We observe that as *f* increases, the curves for the P_{non-im} are shifted to the right of the graph. This is essentially the same result as if we were increasing the density of the guards, (i.e., each node would hear a higher number of guards (see Fig. 15(c))). In Fig. 16(f), we show the average distance normalized over *r* between non-immediate neighbors that have in common more than *th* fractional keys. We observe that for threshold values lower than $th \leq |GH_s| - 10$, all non-immediate neighbors that share sufficient fractional keys are no more than two hops away, for any value of the fraction *f*. We also note that when the communication range variation is biased towards a higher value than the nominal communication range *R*, the average wormhole length decreases (the curves of the wormhole length are shifted to the left). This is due to the fact that as the fraction *f* increases, each node

hears, on average, a higher number of guards. Hence, it is less probable that two nodes not within communication range have in common a higher number of fractional keys.

As a conclusion, based on our simulation results, we showed that our system can adapt to the variation of the communication range at the guards, since the threshold value is decided based on the number of guards heard at each node $|GH_s|$. While the variation of the communication range R affects the absolute value of GH_s, each node locally adapts its threshold to account for the variation.

7. Related work

7.1. Previously proposed mechanisms for preventing the wormhole attack

The wormhole attack in wireless ad-hoc networks was first introduced in [20,34]. In [20], Hu et al. propose two solutions for the wormhole attack. The first is based upon the notion of geographical leashes. Each node includes in every packet its location l_i and a timestamp indicating the time t_s the packet is sent. Since nodes are loosely synchronized, when a node with location l_j receives a packet at time t_p, it verifies the packet could have traveled the distance $\|l_i - l_j\| + \delta$ in a time $t_p - t_s + \Delta$, where δ is the location error and Δ is the synchronization error.

The second solution in [20] is based on temporal leashes. To implement a temporal leash, the sender includes in every packet a timestamp t_s indicating the time t_s the packet is sent and an expiration time t_e. A node that receives a packet at time t_r verifies that $t_r < t_e$ before it accepts the packet. Temporal packet leashes require tight synchronization between all nodes of the network. To illustrate the importance of the synchronization error if the sender's time is Δ time units ahead of the receiver's time, a packet can travel a distance up to $\Delta * c$ ($c = 3 \times 10^8$m/sec) longer than the distance imposed by the expiration time t_e. Similarly if the sender's time is Δ units behind the receiver's time, the receiver has to lie within a distance $\Delta * c$ closer to the sender, compared to the distance imposed by t_e. Hence, the synchronization error should be in the order of nanoseconds for the synchronization error to be negligible.

In [21], Hu et al. provide a bounding distance protocol based on [5] that utilizes a three-way handshake scheme to ensure that the communicating parties are within some distance. The sender sends a challenge to a receiver, who replies immediately with a response. The sender acknowledges the response by another response to complete the three-way handshake. Both parties verify that they lie within some distance by multiplying the round-trip time of flight with the speed of light. Though this protocol does not require the two nodes to be synchronized in order for the protocol to be executed, each node needs to have immediate access to the radio transmitter in order to bypass any queuing and processing delays. In addition, nodes should be equipped with highly accurate clocks with nanosecond precision to avoid distance enlargement.

In [46], Zhu et al. propose a cryptographic solution as a defense mechanism against the wormhole. Based on preloaded keys, nodes are able to derive a pairwise key with any other node without the need for any information exchange. Following a neighbor discovery phase, nodes unicast to every neighbor a cluster key encrypted with the previously derived pairwise key. While the network is secured against the wormhole attack once pairwise keys have been established, the authors of [46] point out that the network is still vulnerable to wormholes during the neighbor discovery phase. If an attacker tunnels and replays the *HELLO* messages between two nodes that are not one hop neighbors, the two nodes will assume that they are one-hop away and establish a cluster key.

A centralized solution for detecting wormhole links, based on multidimensional scaling (MDS), is presented by Wang and Bhargava [47]. Using received signal strength measurements, every node estimates its distance to all its neighbors and reports its distance estimates to a powerful base station. The base station applies MDS to generate a visualization of the network topology. In addition, a smoothing surface operation mitigates the effects of the error in the distance estimation. In a wormhole-free network, the reconstructed topology will correspond to a flat surface. However, in the presence of wormholes, the surface is bent in a circular pattern in order for the two nodes communicating via the wormhole to appear connected. The main limitation of this method is that it requires a relatively dense and uniformly distributed network to detect the wormhole links. Such a visualization cannot be applied to networks with irregular shapes, such as a string topology (nodes connected in one line) or networks with string parts. In addition, based on the simulation results in [47], while the method detects long wormholes (several hops long), smaller wormholes (two to three hops long) can stay undetected with a significantly high probability.

In [19], Hu and Evans utilize directional antennas to prevent wormhole links. Unlike our method, every node of the network is equipped with directional antennas and all antennas should have the same orientation. Different directions called zones are sequentially numbered and every node includes the transmitting zone at each message. A receiver hearing information at a zone A verifies that the sender transmitted the message at the correct zone B, where A, B are opposite zones. Based on information provided by neighbors that assist the wormhole detection by acting as verifiers, every node discovers its neighbors. As pointed out by the authors of [19], a valid verifier must exist in order for the wormhole to be detected, since not all neighbors can act as verifiers. Finally, as noted by the authors of [19], this method

can only prevent single wormholes and does not secure the network against multiple wormhole links [19].

7.2. Interpretation of related work based on our framework

In this section, we show that previously proposed defense mechanisms against the wormhole attack satisfy the graph theoretic model we presented in Section 2.

Time-based methods: In time-based methods [20], every transmitted message has a limited lifetime, less or equal to the communication range r of the nodes divided by the speed light. Hence, messages cannot travel distances longer than the communication range, and links are only established between direct neighbors. For any two synchronized neighbors i, j, node i accepts a message transmitted at time T_s from node j if it is received at a time $T_r < T_s + \frac{r}{c}$, where c is the speed of light. Hence, $e_{i,j} = 1$ if and only if $\|i - j\| \leq r$, a condition that satisfies the geometric graph model in (1). Note that as a requirement, time-based methods have to use the fastest available medium (RF or optical transmission) in order to prevent the wormhole attack.

In an alternative time-based method [5,6,21], nodes measure the time of flight of a challenge-response message before communicating with another node. By limiting the time of flight to twice the communication range over the speed of light, nodes ensure that they establish a link only with their direct neighbors. Hence, time of flight methods also satisfy the geometric graph model in (1).

Location-based methods: In location-based methods [20], every message contains the coordinates of its origin. Hence, any receiving node can infer its distance from the origin of the message and compare it to the communication range r. If $\|i - j\| \leq r$, the message is accepted, otherwise the message is rejected. Hence, a link between two nodes i, j can be established $e_{i,j} = 1$ if and only if $\|i - j\| \leq r$, a condition that satisfies the geometric graph model.

Wormhole visualization: In the wormhole visualization method [47], the base station executing the Multidimensional Scaling (MDS) algorithm constructs the logical graph $\tilde{G}$ of the network based on the distance estimations of each node of the network. By visualizing wormholes as links that will cause the flat network area to curve in a circular way and eliminating surface anomalies, the base station applies a transformation to $\tilde{G}$ that reconstructs the corresponding geometric graph G.

8. Discussion

In our wormhole attack model in Section 2.1, we have assumed that the adversary mounting the attack does not compromise the integrity and authenticity of the communication. Hence, the success of the attack is independent of the cryptographic methods used to secure the communication. The strength of the wormhole attack lies in the fact that the adversary does not need to compromise any cryptographic quantities or network nodes in order to perform the attack in a timely manner. The lack of any compromised entities makes the wormhole attack "invisible" to the upper layers and, hence, the attack is very difficult to detect [20]. Furthermore, the attacker does not need to allocate any computational resources to compromise the communication, thus making the wormhole attack very easy to implement.

Our most compelling argument for assuming no key or host compromise in a wormhole attack scenario is that, if the adversary were to be able to compromise cryptographic keys, there would be no need to record messages at one part of the network, tunnel them via a low-latency link, and replay them to some other part of the network. Instead, the adversary could use the compromised keys to fabricate any message and inject it into the network as legitimate. Using compromised keys to fabricate and inject bogus messages into the network, known as the Sybil attack [13,33], is overall a different problem that is not addressed in this paper.

Since the wormhole attacker does not need to compromise the network communications, we have used a globally shared symmetric key for the protection of the beacon broadcasts from the guards in order to achieve energy-efficient communications (utilize the broadcast advantage of the wireless medium in omnidirectional transmissions). We are indeed aware that a compromise of a single node exposes the globally shared key and allows access to the contents of the guards broadcasts. However, alternative methods for concealing and authenticating the broadcasts of the guards come at the expense of energy-efficiency. Asymmetric key cryptography is known to be computationally expensive for the energy-constrained devices [8]. On the other hand, using pairwise keys shared between the guards and the nodes would provide a higher level of security under key compromise, since only the communication of the node holding the pairwise key is exposed. However, the use of pairwise keys requires the fractional keys to be unicasted from each guard to each node within the communication range, thus making the use of the wireless medium highly inefficient in energy resources.

Furthermore, under key and/or node compromise the wormhole problem essentially becomes a node impersonation (Sybil attack) problem and, hence, cannot be prevented by any of the methods that address the wormhole attack. To illustrate this, consider the case where two nodes not within range have been compromised and that an attacker has deployed a wormhole link between the two nodes.[12] In such

[12] A similar scenario can be considered if the cryptographic keys held by the nodes are compromised and the attacker impersonates the two nodes without using the actual nodes for the attack implementation.

a case, the attacker can implement the wormhole attack via the compromised nodes by recording the information at the origin point, decrypting it and modifying necessary quantities to make the message look legitimate, re-encrypting the message, and tunneling it to the destination point. To prevent this type of attack, additional verifiable information needs to be available, such as verifiable geographical positions for each node or protection against impersonation attacks [33]. In this paper, we have not assumed that such information is available.

Similarly, other schemes that have been proposed for preventing the wormhole attack [19,20,46,47] cannot eliminate wormholes under key/node compromise. We now show for each of the methods in [19,20,46,47] which step is vulnerable to wormholes under key/node compromise.

In [46], different cluster keys are used to encrypt the communication within different one-hop neighborhoods. If cluster keys are compromised, an adversary can record messages at one neighborhood A, decrypt them with the compromised cluster key of neighborhood A, tunnel the messages via the wormhole link to a neighborhood B that is not within the communication range of neighborhood A, re-encrypt the messages with the compromised key of neighborhood B, and replay the messages in neighborhood B. Cluster keys can also be compromised if the adversary compromises the pairwise keys that are used by the nodes to distribute the cluster keys during the initialization phase. For the method in [46], compromise of two nodes that are not within communication range or two pairwise keys is sufficient to create a wormhole.

In [20], the authors use temporal packet leashes to prevent a message from traveling distances longer than a pre-defined distance. Each packet contains an expiration time t_e whose integrity is verified via the use of a keyed message authentication code, such as a key hash function (HMAC). When a node receives a packet, first it verifies that the HMAC for the expiration time is correct (i.e., the expiration time has not been altered while the packet is in transit). If the integrity verification is correct, the receiving node verifies that the packet has not traveled longer than the distance indicated by t_e (the nodes in the network are tightly synchronized). If an adversary were to compromise the keys of a node, it could alter the expiration time to any desired value and properly adjust the keyed message authentication code so that the message can travel any desired length. Thus, the compromise of a single node allows the creation of a wormhole of arbitrary length.

In the wormhole visualization method [47], detection of a wormhole is based on the reconstruction of the network topology via multi-dimensional scaling (MDS) and visualization of wormholes as loops in the network plane. In order to visualize the network topology, every sensor of the network has to report the distance from its one-hop neighbors to a base station. The distance report is protected by a group key known to every sensor. If the group key gets compromised, the adversary can alter the distance reports from the legitimate sensors and manufacture false reports, allowing the creation of wormhole links undetectable by the visualization method. Moreover, it would be very difficult for the visualization method to capture short wormholes in the case where the attacker manipulates the distance reports of the nodes. In the directional antenna method presented in [19], nodes rely upon reports from neighbor nodes to verify the validity of the neighbor discovery protocol. Hence, compromised neighbors can mislead nodes into accepting wormhole links [19] as valid ones.

Though we have shown that the adversary can mount a wormhole attack under node/key compromise, as in the seminal paper in [20], we argue that the strength of the wormhole attack lies in the fact that the adversary does not allocate computational resources to compromise nodes/keys and that it remains "invisible" to upper layers of the network (the attack is implementable with minimal resources). Furthermore, under the node/key compromise assumption, relatively more powerful attacks, such as the Sybil attack [13,33], can be mounted, and there is no need for the adversary to record and replay messages (it can forge messages instead of recording them). Nevertheless, the wormhole attack can still cause significant disruption to vital network operations, such as routing, even if the network communications are not compromised, and, hence, needs to be addressed.

9. Conclusion

We presented a graph theoretic framework for characterizing the wormhole attack in wireless ad hoc networks. We showed that any candidate prevention mechanism should construct a communication graph that is a connected subgraph of the geometric graph of the network. We then proposed a cryptography-based solution to the wormhole attack that makes use of local broadcast keys. We provided a distributed mechanism for establishing local broadcast keys in randomly deployed networks and provided an analytical evaluation of the probability of wormhole detection based on spatial statistics theory. We analytically related network parameters such as deployment density and communication range with the probability of detecting and eliminating wormholes, thus providing a design choice for preventing wormholes with any desired probability. Finally, we also illustrated the validity of our results with extensive simulations.

Appendix A: Choosing the system parameters

The random deployment of the network nodes and guards can be modeled after a *Spatial Homogeneous Poisson Point*

Process [11]. The random placement of a set U of guards with a density $\rho_g = \frac{|U|}{\mathcal{A}}$ ($|\cdot|$ denotes the cardinality of a set) is equivalent to a sequence of events following a homogeneous Poisson point process of rate ρ_g. Given that $|U|$ events occur in area $\mathcal{A}$, these events are uniformly distributed within that area. The random deployment of a set S of nodes with a density $\rho_g = \frac{|S|}{\mathcal{A}}$, is equivalent to a random sampling of the deployment area with rate ρ_s [11]. According to nearest-neighbor theory for the spatially random distribution [11], the distance x of a randomly placed sample point (network node) to the nearest event (guard) has a probability density function (pdf):

$$f(x) = 2x\rho_g\pi e^{-\rho_g\pi x^2}, \tag{28}$$

where $f(x)$ denotes the pdf of the distance x from a randomly placed node to the closest guard, given that guards are also randomly placed.

Avoiding isolated nodes: Given the guard-to-node communication range R and (28), we can compute the probability that a node can hear its nearest guard. Let GH_s denote the set of guards heard by node s, i.e. being within range R from s.

$$\begin{aligned} P(|GH_s| > 0) = P(x \le R) &= \int_0^R 2x\rho_g\pi e^{-\rho_g\pi^2} \\ &= 1 - e^{-\pi\rho_g R^2}. \end{aligned} \tag{29}$$

Extending (29) to all $|S|$ nodes of the network we can ensure that *every* node hears at least one guard. Since nodes are randomly deployed, the number of guards heard by each node is independent of the number of guards heard by another node. Hence, the probability that no network node is isolated is given by the binomial distribution:

$$\begin{aligned} &P(GH_s| > 0, \forall s \in S) \\ &\quad = \binom{|S|}{|S|} P(|GH_s| > 0)^{|S|} P(GH_s| = 0)^0 \\ &\quad = \left(1 - e^{-\rho_g\pi R^2}\right)^{|S|}. \end{aligned} \tag{30}$$

Using (30), we can determine the desired guard density ρ_g, or guard-to-node communication range R, so that each node hears at least one guard with a probability p.

$$\rho_g \ge \frac{-\ln\left(1 - p^{\frac{1}{|S|}}\right)}{\pi R^2}, \quad R \ge \sqrt{\frac{-\ln\left(1-p\right)^{\frac{1}{|S|}}}{\pi\rho_g}} \tag{31}$$

Hearing more than one guard: Since guards are randomly deployed, the probability for a guard to be in an area of size A_g is $p_g = \frac{A_g}{\mathcal{A}}$. In addition, the random guard deployment implies statistical independence between guards being located in a network region A_g, $P(g_i \in A_g | g_j \in A_g) = P(g_i \in A_g)$. Hence, the probability that *exactly* k guards are in A_g is given by the binomial distribution.

$$P(k \in A_g) = \binom{|U|}{k} p_g^k (1 - p_g)^{|U|-k}. \tag{32}$$

For $|U| \gg 1$ and $\mathcal{A} \gg A_g$, we can approximate the binomial distribution with a Poisson distribution:

$$P(k \in A_g) = \frac{\frac{A_g}{\mathcal{A}}|U|}{k!} e^{-\frac{A_g}{\mathcal{A}}|U|} = \frac{\rho_g A_g}{k!} e^{-\rho_g A_g}. \tag{33}$$

By letting $A_g = \pi R^2$, we can compute the probability of having exactly k guards within the communication area (inside a circle of radius R, centered at the node) of any node.

$$P(|GH_s| = k) = \frac{(\rho_g\pi R^2)^k}{k!} e^{-\rho_g\pi R^2}. \tag{34}$$

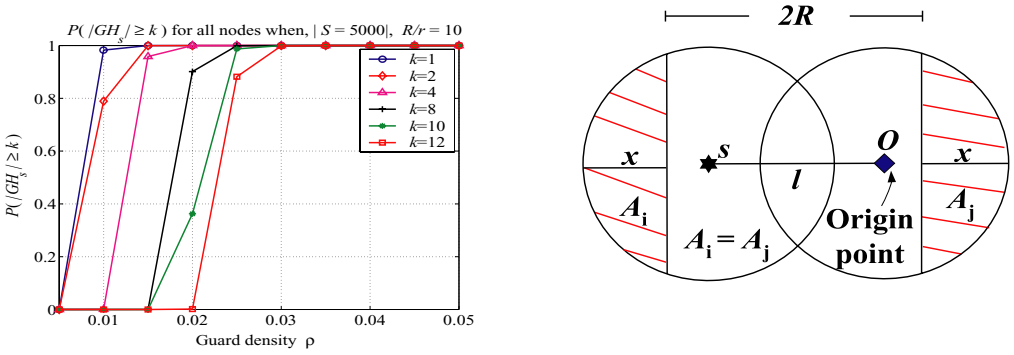
Using (34), we compute the probability that *every* node hears *at least* k guards. The random node deployment implies statistical independence in the number of guards heard by each node and hence:

$$\begin{aligned} P(|GH_s| \ge k, \forall s \in S) &= P(|GH_s| \ge k)^{|S|} \\ &= (1 - P(|GH_s| < k))^{|S|} \\ &= \left(1 - \sum_{i=0}^{k-1} \frac{(\rho_g\pi R^2)^i}{i!} e^{-\rho_g\pi R^2}\right)^{|S|}. \end{aligned} \tag{35}$$

From (35), we can graphically determine the guard density needed, so that each node hears at least k guards with a very high probability p. In Fig. 17(a), we plot the probability in (35), for different guard densities ρ_g and different values of k.

Expected number of guards heard: The expected number of guards that each node hears can be calculated by taking the mean value of (34) assuming an infinite plane.

$$\begin{aligned} E(|GH_s|) &= \sum_{k=0}^{\infty} k \frac{(\rho_g\pi R^2)^k}{k!} e^{-\rho_g\pi R^2} \\ &= \rho_g\pi R^2 e^{-\rho_g\pi R^2} = \sum_{k=0}^{\infty} \frac{(\rho_g\pi R^2)^{k-1}}{(k-1)!} = \rho_g\pi R^2. \end{aligned} \tag{36}$$



(a) (b)

Fig. 17 Probability that each node hears at least k guards, for different guard densities ρ_g and different values of k, when $\frac{R}{r} = 10$

Appendix B: Maximizing the lower bound on $P(CR)$

The lower bound on detection probability based on the communication range constraint property is given by:

$$P(CR) \geq \left(1 - e^{-\rho_g A_i}\right)\left(1 - e^{-\rho_g A_j}\right). \tag{37}$$

We want to compute the values of A_i^*, A_j^*, that maximize the right side of (37). From Fig. 17(b),

$$A_i(d) = 2\int_{R-d}^{R} \sqrt{R^2 - z^2}dz,$$
$$A_j(d) = 2\int_{R+d-l}^{R} \sqrt{R^2 - z^2}dz. \tag{38}$$

where $l = \|s - O\|$. Since, both A_i, A_j are expressed as function of d, the lower bound $LB(d)$ on $P(CR)$ can be expressed as:

$$LB(d) = \left(1 - e^{-\rho_g A_i(d)}\right)\left(1 - e^{-\rho_g A_j(d)}\right). \tag{39}$$

To maximize $LB(d)$ we differentiate over d and set the derivative equal to zero:

$$\begin{aligned} LB'(d) &= \rho_g A_i'(d)e^{-\rho_g A_i(d)} + \rho_g A_j'(d)e^{-\rho_g A_j(d)} \\ &\quad -\rho_g(A_i'(d) + A_j'(d))e^{-\rho_g(A_i(d)+A_j(d))} \\ &= \rho_g\left[A_i'(d)\left(e^{-\rho_g A_i(d)} - e^{-\rho_g(A_i(d)+A_j(d))}\right)\right. \\ &\quad \left. +A_j'(d)\left(e^{-\rho_g A_j(d)} - e^{-\rho_g(A_i(d)+A_j(d))}\right)\right] = 0. \end{aligned} \tag{40}$$

A trivial solution to $LB'(d) = 0$ is $A_i(d) = 0$, or $A_j(d) = 0$, but both yield a minimum ($LB(d) = 0$), rather than a maximum. However if we set $A_i(d) = A_j(d)$, from (38), $R + d - l = R - d \Rightarrow d = \frac{l}{2}$. In addition, differentiating (38), and evaluating at $d = \frac{l}{2}$ yields $A_i'(\frac{l}{2}) = -A_j'(\frac{l}{2})$. Hence, for $A_i(d) = A_j(d)$, $LB'(d) = 0$, and the maximum value on the lower bound $LB(d)$ is achieved. The values of A_i, A_j that maximize $LB(d)$ are,

$$\begin{aligned} A^*(d) &= A_i^*(d) = A_j^*(d) \\ &= 2\int_{R-d}^{R} \sqrt{R^2 - z^2}dz \\ &= d\sqrt{R^2 - d^2} - R^2 \tan^{-1}\left(\frac{d\sqrt{R^2 - d^2}}{d^2 - R^2}\right), d = \frac{l}{2}, \end{aligned} \tag{41}$$

and the lower bound can now be expressed as:

$$LB(d) = \left(1 - e^{-\rho_g A^*(d)}\right)^2. \tag{42}$$

References

1. C. Balanis, *Antenna Theory* (John Wiley & Sons, 1982).
2. D. Bertsekas and R. Gallager, *Data Networks* (2nd ed.) (Prentice Hall, NJ, 1992).
3. C. Bettstetter, On the minimum node degree and connectivity of a wireless multihop network, in: *Proceedings of the Third ACM International Symposium on Mobile Ad Hoc Networking & Computing*, (Oct. 2002) pp. 80–91.
4. R.V. Boppana and S. Konduru, An adaptive distance vector routing algorithm for mobile Ad Hoc Networks, in: *Proceedings of INFOCOM* (April 2001) pp. 1753–1762.
5. S. Brands and D. Chaum, Distance-bounding protocols, in: *Proceedings of Cryptography* (Aug. 1994) Vol. 839 of Lecture Notes in Computer Science, pp. 344–359.
6. S. Čapkun, L. Buttyan, J. Hubaux, SECTOR: Secure tracking of node encounters in multi-hop wireless networks, in: *Proceedings of Security of Ad Hoc and Sensor Networks* (Oct. 2003) pp. 21–32.
7. S. Čapkun, and J.P. Hubaux, Secure positioning of wireless devices with application to sensor networks, in: *IEEE INFOCOM 2005* (March 2005).

8. D. Carman, P. Kruus and B. Matt, Constraints and Approaches for Distributed Sensor Network Security, NAI Labs Technical Report No. 00-010 (2002).
9. H. Chan, A. Perrig and D. Song, Random key pre-distribution schemes for sensor networks, in: *Proceedings of the IEEE Symposium on Research in Security and Privacy* (May 2003) pp. 197–213.
10. D. Coppersmith and M. Jakobsson, Almost optimal hash sequence traversal, in: *Proceedings of the Financial Cryptography* (March 2002) Lecture Notes in Computer Science, IFCA, Springer-Verlag, Berlin, Germany.
11. N. Cressie, *Statistics for Spatial Data* (John Wiley & Sons, NY 1993).
12. W. Diffie and M.E. Hellman, New directions in cryptography, IEEE Transactions on Information Theory 22 (Nov. 1976) 644–654.
13. J. Douceur, The Sybil attack, in: *Proceedings of IPTPS 2002* (March 2002).
14. L. Eschenauer and V. D. Gligor, A key-management scheme for distributed sensor networks, in: *Proceedings of the Ninth ACM Conference on Computer and Communications Security* (Nov. 2002) pp. 41–47.
15. A. Farago, Scalable analysis and design of ad hoc networks via random graph theory, in: *Proceedings of the Sixth International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications* (2002) pp. 43–50.
16. J. Gao, L. Guibas, J. Hershberger, L. Zhang, and A. Zhu, Geometric spanner for routing in mobile networks, in: *Proceedings of the Second ACM International Symposium on Mobile Ad Hoc Networking & Computing* (Oct. 2002) pp. 45–55.
17. T. He, C. Huang, B. Blum, J. Stankovic and T. Abdelzaher, Range-free localization schemes in large scale sensor networks, in: *Proceedings of the Fourth ACM International Symposium on Mobile Ad Hoc Networking & Computing* (Oct. 2004) pp. 81–95.
18. B. Hofmann-Wellenhof, H. Lichtenegger and J. Collins, *Global Positioning System: Theory and Practice* (4th ed.) (Springer-Verlag, Vienna, 1997).
19. L. Hu and D. Evans, Using directional antennas to prevent wormhole attacks, in: *Proceedings of NDSS* (Feb. 2004).
20. Y. Hu, A. Perrig and D. Johnson, Packet leashes: A defense against wormhole attacks in wireless ad hoc networks, in: *Proceedings of INFOCOM* (April 2003) vol. 2, 1976–1986.
21. Y. Hu, D. Johnson and A. Perrig, Rushing attacks and defense in wireless ad hoc network routing protocols, in: *Proceedings of the ACM Workshop on Wireless Security* (Sept. 2003) pp. 30–40.
22. D.B. Johnson, D.A. Maltz and J. Broch, The dynamic source routing protocol for multihop wireless ad hoc networks, in: *Ad Hoc Networking*, (Addison-Wesley, 2001), ch. 5, pp. 139–172.
23. F. Kuhn, R. Wattenhofer, Y. Zhang and A. Zollinger, Geometric routing: of theory and practice, in: *Proceedings of the 22nd ACM Symposium on the Principles of Distributed Computing (PODC)* (2003).
24. F. Kuhn, R. Wattenhofer and A. Zollinger, Asymptotically optimal geometric mobile ad-hoc routing, in: *Proceedings of the Sixth International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIAL-M)*, pp. 24–33.
25. C. Karlof and D. Wagner, Secure routing in wireless sensor networks: Attacks and countermeasures, in: *Proceedings of WSNA* (May 2003) pp. 113–127.
26. L. Lamport, Password authentication with insecure communication, Communications of the ACM, 24, (Nov. 1981), 770–772.
27. L. Lazos and R. Poovendran, SeRLoc: Secure range-independent localization for wireless sensor networks, in: *Proceedings ACM WiSe* (Oct. 2004).
28. L. Lazos, S. Čapkun and R. Poovendran, ROPE: Robust position estimation in wireless sensor networks, in: T*he Fourth International Conference on Information Processing in Sensor Networks (IPSN 2005)* (April 2005).
29. Z. Li, W. Trappe, Y. Zhang and B. Nath, Robust statistical methods for securing wireless localization in sensor networks, in: T*he Fourth International Conference on Information Processing in Sensor Networks (IPSN 2005)* (April 2005).
30. D. Liu and P. Ning, Attack-resistant location estimation in sensor networks, in: T*he Fourth International Conference on Information Processing in Sensor Networks (IPSN 2005)* (April 2005).
31. MICA Wireless Measurement System, available at: *http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA.pdf*.
32. S. Murthy and J.J. Garcia-Luna-Aceves, An efficient routing protocol for wireless networks, ACM Mobile Networks and App. J., Special Issue on Routing in Mobile Communication Networks (Oct. 1996) pp. 183–97.
33. J. Newsome, E. Shi, D. Song and A. Perrig, The Sybil attack in sensor networks: Analysis and defenses, in: *Proceedings of IPSN 2004*, Berkeley, CA (April 2004).
34. P. Papadimitratos and Z.J. Haas, Secure routing for mobile ad hoc networks, in: *Proceedings of CNDS 2002* (Jan. 2002).
35. M. Penrose, *Random Geometric Graphs* (Oxford University Press, NY, 2003).
36. C.E. Perkins and P. Bhagwat, Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers, in: *Proceedings of the SIGCOMM* (Aug. 1994) pp. 234–244.
37. C.E. Perkins and E.M. Royer, Ad-hoc on-demand distance vector routing, in: *Proceedings of WMCSA* (Feb. 1999) pp. 90–100.
38. A. Perrig, R. Canetti, J. Tygar and D. Song, The TESLA broadcast authentication protocol, RSA Cryptobytes, 5(2), Summer 2002.
39. A. Perrig, R. Szewczyk, V. Wen, D. Culler and J. Tygar, SPINS: Security protocols for sensor networks, in: *Proceedings of MOBICOM 2001*, Rome, Italy (July 2001).
40. J. Proakis, *Digital Communications* (4th ed.) (McGraw Hill Inc., 2001).
41. R.L. Rivest, The RC5 encryption algorithm, in: *Proceedings of the First Workshop on Fast Software Encryption* (Dec. 1994) pp. 86–96.
42. R.L. Rivest, A. Shamir and L.M. Adleman, A method for obtaining digital signatures and public-key cryptosystems, Communications of the ACM 21 (Jan. 1978) 120–126.
43. A. Savvides, C. Han and M. Srivastava, Dynamic fine-grained localization in ad-hoc networks of sensors, in: *Proceedings of MOBICOM* (July 2001) pp. 166–179.
44. Y. Shang, W. Ruml, Y. Zhang and M. Fromherz, Localization from mere connectivity, in: *Proceedings of MOBIHOC* (June 2003) pp. 201–212.
45. D. Stinson, *Cryptography: Theory and Practice* (2nd ed.) (CRC Press, 2002).
46. S. Zhu, S. Setia and S. Jahodia, LEAP: Efficient security mechanisms for large-scale distributed sensor networks, in: *Proceedings of CCS* (Oct. 2003) pp. 62–72.
47. W. Wang and B. Barhgava, Visualization of wormholes in sensor networks, in: *Proceedings of WISE 2004* (Oct. 2004).
48. R. Wattenhofer, L. Li, P. Bahl, and Y.-M. Wang, Distributed topology control for power efficient operation in multihop wireless ad hoc networks, in: *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)* (2001) pp. 1388–1397.

Radha Poovendran received the Ph.D. degree in electrical engineering from the University of Maryland, College Park, in 1999. He has been an Assistant Professor in the Electrical Engineering Department, University of Washington, Seattle, since September 2000. His research interests are in the areas of applied cryptography for multiuser environment, wireless networking, and applications of information theory to security. Dr. Poovendran is a recipient of the Faculty Early Career Award from the National Science Foundation (2001), Young Investigator Award from the Army Research Office (2002), Young Investigator Award from the Office of Naval Research (2004), and the 2005 Presidential Early Career Award for Scientists and Engineers, for his research contributions in the areas of wired and wireless multiuser security.

Loukas Lazos received the B.S. and M.S. degrees from the Electrical Engineering Department, National Technical University of Athens, Athens, Greece, in 2000 and 2002, respectively. He is currently working towards the Ph.D. degree in the Electrical Engineering Department, University of Washington, Seattle. His current research interests focus on cross-layer designs for energy-efficient key management protocols for wireless ad-hoc networks, as well as secure localization systems for sensor networks.