

Secure Neighbor Discovery in Wireless Networks: Is It Possible?

Marcin Poturalski, Panos Papadimitratos, Jean-Pierre Hubaux

ABSTRACT

Wireless communication enables a broad spectrum of applications, ranging from commodity to tactical systems. *Neighbor discovery* (ND), that is, determining which devices are within direct radio communication, is a building block of network protocols and applications, and its vulnerability can severely compromise their functionalities. A number of proposals to *secure* neighbor discovery have been published, but none have analyzed the problem formally. In this paper, we provide such an analysis: we build a formal model capturing salient characteristics of wireless systems, most notably obstacles and interference, and we contribute a specification of the ND problem. Then, we derive an *impossibility result* for a general class of protocols that we term “message and time protocols,” to which many of the schemes in the literature belong. This implies that those schemes solve problems related, but *not equivalent*, to neighbor discovery. We also identify conditions under which the impossibility result is lifted. Moreover, we explore a second class of protocols we term “message, time and location protocols,” and prove they can secure ND.

1. INTRODUCTION

Wireless networking is a key enabler for mobile communication systems, ranging from cellular infrastructure-based data networks and wireless local area networks (WLANs) to disaster-relief, tactical, and sensor networks, and short-range wire replacement and radio frequency identification (RFID) technologies. In all such systems, any two wireless devices communicate directly when in range, without the assistance of other devices. The ability to determine if direct, one-hop, communication takes place is fundamental. For example, a WLAN access point (AP) assigns a new IP address to a mobile station only when it is within the AP’s coverage area. Or, a mobile node does not initiate a route discovery across a mobile ad hoc network (MANET) if a sought destination is already in its neighbor table. Or, an RFID tag will be read only if the signal transmitted by the tag can be received directly by the reader. These examples illustrate

that, depending on whether another system entity denoted as *node* in the rest of the paper, is a *neighbor* or not, actions are taken (e.g., by the AP or the router) or implications are derived (e.g., the RFID tag and reader are physically close). In other words, *discovering a neighbor*, or knowing that a node *is* a neighbor, is a common building block and enabler of diverse system functionality.

The naive solution for *neighbor discovery* (ND) is to have nodes broadcast periodically their identity; reception at node *A* of such a beacon from node *B* suffices for *A* to add *B* to its neighbor table. In many systems, the mere reception of a signal implies its sender is a neighbor. Unfortunately, this is trivial for an adversary to abuse, even though entity authentication may appear as a solution: an adversary *C* beaconing C_1, C_2, \dots, C_k identities cannot mislead a correct, protocol-abiding node *A* it has *k* fictitious neighbors. However, authentication does *not* imply the node is a neighbor; it only establishes which node created but not which sent a message across the wireless medium. To illustrate this, consider *A* and *B* unable to communicate directly, and *C* within range of both *A* and *B*. Node *C* receives and *repeats* *B*’s beacon, for example, digitally signed and time-stamped, with no modification. Then, *A* receives the beacon and identifies *B* as a neighbor, even though this is not so, exactly because *A* cannot distinguish whether the message (beacon) was sent directly by *B* or it was *relayed* by another node. In fact, the adversary does not need to receive an entire message before it replays it, but can relay messages extremely efficiently: Reid *et al* [19] substantiate this is a *realistic threat*, reporting relaying delays as small as 40ns.

Preventing such *relaying* attacks, one form of which is often termed as a *wormhole*, is critical. Manipulation of ND can be a simple yet powerful attack to mount, without the need to break a cryptosystem or exploit security protocol flaws. Letting legitimate nodes erroneously believe that they are neighbors allows, for example, the adversary to fully control communication across these artificial links. The threat lies in that the attacker can deny communication exactly when a message critical for the system operation is transmitted. From a different point of view, “worm-holing” an RFID signal while the tag (and its owner) are not physically near the RFID reader can gain unauthorized access to the premises of the tag owner.

A number of schemes were designed to thwart wormholes, essentially safeguarding neighbor discovery. *Distance bound-*

ing [2] has been the basic approach: the distance of two nodes is estimated by measuring the signal *time of flight* from and to those nodes. If the estimate is below a threshold corresponding to the nodes' communication range, the node is accepted as neighbor. Other related distributed protocols, running at the nodes themselves, are discussed in detail in Sec. 7.¹ Existing solutions were not formally analyzed, and our findings here suggest that none of the existing schemes addresses the problem of secure neighbor discovery to its full extent.

Arguably, distance bounding may provide the desired level of security for some applications. For example, if an RFID reader can conclude that a tag is within a range of 10cm, it is safe to have the building door opened. However, in general, proximity is *not* sufficient for two nodes to be neighbors. Obstacles or interference can prevent two nearby nodes from communicating directly. This allows the attacker to abuse a ND mechanism oblivious to such obstructions, and mislead two nodes that they are neighbors. This aspect of ND has been largely overlooked by schemes proposed to this date.

In this paper, we address the above problems, contributing a formal model of wireless ad hoc networks, a specification of secure neighbor discovery, and a unifying analysis of two general classes of protocols. To the best of our knowledge, this is the first such specification, formal wireless network model with the necessary detail for the problem at hand, and formal investigation of ND. We denote the first class as the *message and time* (MT) protocols, for which nodes exchange messages and are able to accurately measure time. For this class, which includes a large fraction of existing schemes, we show the following *impossibility result*: no MT protocol can solve the (secure) ND problem if adversarial nodes are able to relay messages with a delay below a certain threshold (Sec. 3). On the contrary, if the minimum relaying delay is above that same threshold, we show it is possible to achieve secure ND (Sec. 4). Then, in Sec. 5, we consider the second class of protocols we term *message, time, and location* (MTL) protocols: nodes are, in addition to an MT protocol capabilities, aware of their location. We show that MTL protocols secure ND even if adversarial nodes can relay messages with almost no delay. Due to space limitations, we present the impossibility result proof and a sketch of the MT positive result, but omit all other proofs. We also discuss implications of our results and model assumptions (Sec. 6), along with future and related work. Practical considerations on protocol design are covered in App. A.

2. SYSTEM MODEL

We are interested in modeling a wireless network, with its basic entities, *nodes*, being processes running on computational platforms equipped with transceivers communicating over a wireless channel. We assume here that nodes have synchronized clocks and are static (not mobile). Nodes either follow the implemented system functionality, in which case we denote them as *correct*, or they are under the control of an adversary; then, we denote them as *adversarial* nodes.

We model communication at the physical layer, rather than

¹We focus on distributed solutions and do not consider here centralized proposals utilizing statistical [4] or visual information [24].

at higher layers (data link, network, or application), in order to capture the inherent characteristics of neighbor discovery in wireless networks. For simplicity, correct nodes are assumed to use a single wireless channel and omnidirectional antennas, but we do not require them to have equal transmission power and receiver sensitivity. On the other hand, adversarial nodes have enhanced capabilities, are assumed to use directional antennas, and are able to communicate not only across the wireless channel used by correct nodes, but also across a dedicated *adversary channel* imperceptible to correct nodes.

Our system model comprises: (i) a setting \mathcal{S} , which describes the type (correct or adversarial) and location of nodes, and how the wireless channel state changes over time; (ii) a protocol model \mathcal{P} , which determines the behavior of correct nodes; (iii) an adversary model \mathcal{A} , which determines the capabilities of adversarial nodes.

We make the assumption that if we look at the system at any point in time, one or more phenomena occur. We are interested in phenomena relevant to the wireless communication and the system at hand, and, consequently, to our analysis. We denote these phenomena, associated with nodes, as *events* (Def. 2). Then, we model the system evolution over time using the notion of *trace*, i.e., a set of events (Def. 3). More precisely, we use *feasible* traces, which satisfy constraints specified by \mathcal{S} (proper correspondence between wireless sending and receiving of messages), \mathcal{P} (correct nodes follows the protocol), and \mathcal{A} (adversarial node behave according to their capabilities).

Naturally, we provide the specification of secure neighbor discovery exclusively with respect to feasible traces. The specification consists of two properties requiring that (i) if a node concludes that some other correct node is a neighbor, then it is indeed a neighbor (in every feasible trace), and (ii) if two correct nodes are neighbors, it should be possible for them to conclude they are neighbors (in some setting and feasible trace). We call this *two-party* neighbor discovery, with only two nodes participating in an ND protocol run. We discuss later an alternative *multi-party* ND, which relies on the participation of additional correct nodes to conclude successfully on whether two nodes are neighbors or not.

2.1 Public parameters

Public parameters are constants used by the protocols, known to the protocol designer and the adversary, depending on the technology and wireless channel in use. The protocol designer has limited control on the selection of these parameter values. Our model includes the following *public parameters*:

- \mathbb{V} , the set of unique *node identifiers*, which for simplicity we will consider with the nodes themselves,
- $\mathbf{R} \in \mathbb{R}_{>0}$, the *neighbor discovery* (ND) *range*,
- $\mathbf{v} \in \mathbb{R}_{>0}$, the *signal propagation speed* across the wireless channel,
- $\mathbf{v}_{\text{adv}} \geq \mathbf{v}$, the *information propagation speed* over the *adversary channel*,
- \mathbb{M} , the set of *messages*,

- $|\cdot| : \mathbb{M} \rightarrow \mathbb{R}_{>0}$, the *message duration* function, returning the transmission delay² for any message $m \in \mathbb{M}$.

\mathbf{R} is often denoted as the *nominal range* for wireless communication, however, we use \mathbf{R} more freely as the communication range for which ND inferences are drawn, or in other words, as part of the ND specification (Sec. 2.7). Parameter \mathbf{v} defines how fast information can propagate across the wireless channel; once a communication technology is selected, this cannot be controlled by the system designer. Parameter \mathbf{v}_{adv} is under the control of the adversary: he can elect the technology and thus how fast information can propagate between adversarial nodes across the adversary channel. The message space is system specific and under the control of the system designer, while the message duration function depends on the used technology and the achievable transmission rates, e.g., in bits per second.

2.2 Settings

A setting describes the type and location of nodes, and how the state of the wireless channel changes over time.

DEFINITION 1. A setting \mathcal{S} is a tuple $\langle V, \text{loc}, \text{type}, \text{link} \rangle$, where:

- $V \subset \mathbb{V}$ is a finite set of nodes. A pair $(A, B) \in V^2$ is called a link.
- $\text{loc} : V \rightarrow \mathbb{R}^2$ is called a location function³. As we assume nodes are not mobile, this function does not depend on time. We define $\text{dist} : V \times V \rightarrow \mathbb{R}_{\geq 0}$ as $\text{dist}(A, B) = d_2(\text{loc}(A), \text{loc}(B))$, where d_2 is the Euclidean distance in \mathbb{R}^2 . We require the loc function to be injective, so that no two nodes share the same location. Thus, $\text{dist}(A, B) > 0$ for $A \neq B$.
- $\text{type} : V \rightarrow \{\text{correct}, \text{adversarial}\}$ is the type function; it defines which nodes are correct and which are adversarial. This function does not depend on time, as we assume that the adversary does not corrupt new nodes during the system execution. We denote $V_{\text{cor}} = \text{type}^{-1}(\{\text{correct}\})$ and $V_{\text{adv}} = \text{type}^{-1}(\{\text{adversarial}\})$.
- $\text{link} : V^2 \times \mathbb{R}_{\geq 0} \rightarrow \{\text{up}, \text{down}\}$ is the link state function. Accordingly to this function we say that at a given time $t \geq 0$, a link $(A, B) \in V^2$ is up (denoted $t :: A \rightarrow B$) or down (denoted $t :: A \nrightarrow B$). We use abbreviations $t :: A \leftrightarrow B \stackrel{\text{def}}{=} t :: A \rightarrow B \wedge t :: B \rightarrow A$ and $t :: A \nleftrightarrow B \stackrel{\text{def}}{=} t :: A \nrightarrow B \wedge t :: B \nrightarrow A$. We extend the “ $t :: A \rightarrow B$ ” notation from single time points to sets (note that we only use intervals) as follows: $T :: A \rightarrow B \stackrel{\text{def}}{=} \forall t \in T, t :: A \rightarrow B$. We assume the convention $\mathbb{R}_{\geq 0} :: A \nrightarrow A$.

We denote the set of all settings by Σ .

²Note that transmission delay does *not* include the propagation delay.

³All the results of this paper can be immediately transcribed to \mathbb{R}^3 . The \mathbb{R}^2 space is used only for presentation simplicity.

2.3 Traces

We use the notion of *trace* to model an execution of the system; a trace is composed of *events*. We model events related to the wireless communication and the detection of a neighbor. The former, denoted as **Bcast**, **Dcast** and **Receive**, model broadcast (or omnidirectional) transmission, directional transmission, and reception, respectively. The latter, denoted as **Neighbor**, means that a node accepted another node as a neighbor. Each event is primarily associated with (essentially, takes place at) a node we denote as the *active* node. For some events, a secondary association with another node can exist. In particular:

DEFINITION 2. An event is one of the following terms:

- $A; t :: \text{Bcast}(m)$
- $A; t :: \text{Dcast}(\alpha, \beta, m)$
- $A; t :: \text{Receive}(B, m)$
- $A; t :: \text{Neighbor}(C, t')$

where: $A \in \mathbb{V}$ is the active node, $m \in \mathbb{M}$ is a message, $t \in \mathbb{R}_{\geq 0}$ is the start time, $\alpha \in [0, 2\pi)$ is the sending direction, $\beta \in (0, 2\pi]$ is the sending angle, $B \in \mathbb{V}$ is the sender node, $C \in \mathbb{V}$ is a declared neighbor, $t' \in \mathbb{R}_{\geq 0}$ is the time at which C is a neighbor according to A 's declaration.

For an event e , we write $\text{start}(e)$ for its start time and $\text{end}(e)$ for its end time. For events including a message m , $\text{end}(e) = \text{start}(e) + |m|$, while for the **Neighbor** event $\text{end}(e) = \text{start}(e)$.

Dcast, representing a message sent with a directional antenna at direction α over an angle β is illustrated in Fig. 1. **Receive** represents message reception triggered (caused) by any incoming message, and thus a previous **Bcast** and **Dcast** event (which are self-triggered at the sending node). **Neighbor** can be thought of as an internal outcome (of a neighbor discovery protocol, to be defined later). Then, traces comprising the above events are defined.

DEFINITION 3. A trace θ is a set of events that satisfies what we will call the finite cut condition: for any finite $t \geq 0$, the subset $\{e \in \theta \mid \text{start}(e) < t\}$ is finite.

We denote the set of all traces by Θ .

The finite cut condition ensures that during a finite amount of time only a finite number of events occurs, which should be expected as we allow only a finite number of nodes in any setting.

2.4 Setting-Feasible Traces

Feasibility with respect to a setting \mathcal{S} is a set of conditions ensuring a proper causal and time relation between send and receive events.

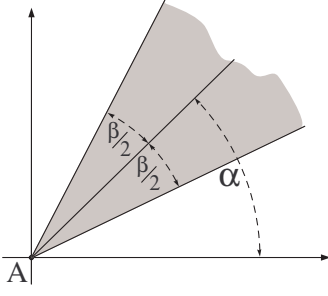


Figure 1: Range of the Dcast primitive. $\text{inrange}(A, \alpha, \beta, B)$ is true iff B is located in the gray region.

DEFINITION 4. A trace $\theta \in \Theta$ is feasible with respect to a setting $\mathcal{S} = \langle V, \text{loc}, \text{type}, \text{link} \rangle$, if the following conditions are satisfied:

1. $\forall A; t :: \text{Receive}(B, m) \in \theta,$
 $(A, B \in V) \wedge ([t, t + |m|] :: B \rightarrow A) \wedge$
 $(B; t - t_{AB} :: \text{Bcast}(m) \in \theta \vee (\text{inrange}(B, \alpha, \beta, A) \wedge$
 $B; t - t_{AB} :: \text{Dcast}(\alpha, \beta, m) \in \theta))$
2. $\forall A; t :: \text{Bcast}(m) \in \theta, (A \in V) \wedge$
 $(\forall B \in V, [t + t_{AB}, t + t_{AB} + |m|] :: A \rightarrow B \implies$
 $B; t + t_{AB} :: \text{Receive}(A, m) \in \theta)$
3. $\forall A; t :: \text{Dcast}(\alpha, \beta, m) \in \theta, (A \in V_{\text{adv}}) \wedge$
 $(\forall B \in V, (\text{inrange}(A, \alpha, \beta, B) \wedge$
 $[t + t_{AB}, t + t_{AB} + |m|] :: A \rightarrow B) \implies$
 $B; t + t_{AB} :: \text{Receive}(A, m) \in \theta)$

Where \vee denotes logical exclusive or, $t_{AB} = \frac{\text{dist}(A, B)}{v}$ is the time of flight, and $\text{inrange}(A, \alpha, \beta, B)$ is defined in Fig. 1.

We denote the set of all traces feasible with respect to a setting \mathcal{S} by $\Theta_{\mathcal{S}}$.

Cond. 1 of Def. 4 ensures that every message that is received was previously sent. Cond. 2 ensures that a broadcasted message is received by all nodes enabled to do so by the link relation.⁴ Cond. 3 ensures that a Dcast-ed message is received only by the nodes in the area as per the Dcast transmission (see Fig. 1), and only if the link is up. In other words, communication is causal (a receive is always preceded by a sent), and reliable as long as the link is up. Unreliability, expected and common in wireless communications, is modeled by links state being *down*. Furthermore, the three conditions in Def. 4 introduce a strict time relation between events, reflecting the signal propagation across the channel with a constant speed v .

2.5 Protocol-Feasible Traces

A trace is essentially a *global view* of the system execution. To describe what a node observes during a system execution, we use the notion of *local view*, a significant part of which is a *local trace* composed of *local events*. We define these next.

⁴Note that time is “measured” at the receiver, not the sender.

DEFINITION 5. A local event is one of the terms:
 $t :: \text{Bcast}(m), t :: \text{Receive}(m), t :: \text{Neighbor}(B, t')$, where $B \in V, m \in M, t, t' \in \mathbb{R}_{\geq 0}$. For a local event e , $\text{start}(e)$, $\text{end}(e)$ are defined as in Def. 2.

DEFINITION 6. A local trace is a set of local events that satisfies the finite cut condition. Given a node identifier $A \in V$, time $t \geq 0$ and trace $\theta \in \Theta$, we calculate the local trace of node A at time t derived from trace θ , denoted $\theta|_{A, t}$, as follows:

$$\theta|_{A, t} = \{t_1 :: \text{Bcast}(m) \mid t_1 < t \wedge A; t_1 :: \text{Bcast}(m) \in \theta\} \cup \quad (1)$$

$$\{t_1 :: \text{Receive}(m) \mid t_1 + |m| < t \wedge \exists B \in V, A; t_1 :: \text{Receive}(B, m) \in \theta\} \cup \quad (2)$$

$$\{t_1 :: \text{Neighbor}(B, t') \mid t_1 < t \wedge A; t_1 :: \text{Neighbor}(B, t') \in \theta\} \quad (3)$$

We call $\theta|_{A, \infty}$ a complete local trace of A in θ and denote it shortly $\theta|_A$.

Note that the **Receive** local event, contrary to its global counterpart, does not include the information about the sender of the message. This is of central importance, capturing the earlier mentioned challenge in securing ND in wireless networks: a receiver can identify the creator of a message but not how the message was received.

We identify two variants of the local view notion: an *MT-local view*, as the basis for defining the class of message and time protocols, and an *MTL-local view*, used to define the class of message, time and location protocols.

DEFINITION 7. Given a trace θ , an MT-local view of node A at time t derived from θ is a tuple $\langle A, t, \theta|_{A, t} \rangle$; we denote it $\theta|_{A, t}$.

DEFINITION 8. Given a trace θ and a setting \mathcal{S} , an MTL-local view of node A at time t derived from θ is a tuple $\langle A, t, \text{loc}(A), \theta|_{A, t} \rangle$; we denote it $\theta|_{A, t}$ as well.

Note that \mathcal{S} is part of Def. 8 since the location of node A is defined only within a specific setting. With the notion of local view(s) in hand, we can proceed with the definition of a protocol model. This definition captures the property of protocols essential to our investigation: the fact that protocol behavior depends *exclusively* on the local view of the node executing the protocol.

DEFINITION 9. An MT(MTL)-protocol model \mathcal{P} is a function which given a MT(MTL)-local view $\theta|_{A, t}$, determines a finite, non-empty set of actions; an action is one of the terms: $\epsilon, \text{Bcast}(m)$ or $\text{Neighbor}(A, t)$, where $m \in M, A \in V, t \in \mathbb{R}_{\geq 0}$.

The interpretation of **Bcast** and **Neighbor** actions is natural. The ϵ action means that the node does not execute

an event, with the exception of possible **Receive** event(s). Note that modeling the protocol output (i.e., the protocol model codomain) as a family of *sets of actions* allows for non-deterministic protocols.

DEFINITION 10. A trace $\theta \in \Theta_S$ is feasible with respect to a MT or MTL protocol model \mathcal{P} , if the following conditions are satisfied:

1. $\forall A \in V_{\text{cor}}, \forall A; t :: \text{Bcast}(m) \in \theta, \text{Bcast}(m) \in \mathcal{P}(\theta|_{A,t})$
2. $\forall A \in V_{\text{cor}}, \forall A; t :: \text{Neighbor}(B, t') \in \theta, \text{Neighbor}(B, t') \in \mathcal{P}(\theta|_{A,t})$
3. $\forall A \in V_{\text{cor}}, \forall t \in X_A, \epsilon \in \mathcal{P}(\theta|_{A,t})$, where
 $X_A = \mathbb{R}_{\geq 0} \setminus \text{start}(\theta|_A \cap E)$,
 $E = \{t :: \text{Bcast}(m) \mid m \in \mathbb{M}, t \in \mathbb{R}_{\geq 0}\} \cup \{t :: \text{Neighbor}(B, t') \mid B \in \mathbb{V}, t, t' \in \mathbb{R}_{\geq 0}\}$

We denote the set of all traces feasible with respect to a setting S and MT(MTL)-protocol model \mathcal{P} by $\Theta_{S,\mathcal{P}}$.

Cond. 1 and 2 of Def. 10 ensure that **Bcast** of **Neighbor** actions taken by a node are allowed by the protocol. Cond. 3, with X_A the set of all points in time at which no event other than **Receive** happens at node A , ensures that if a node performs no action, that is also allowed by the protocol.

2.6 Adversary-Feasible Traces

We consider a relatively limited adversary, which is only capable of relaying messages. We denote this model as $\mathcal{A}_{\Delta_{\text{relay}}}$, with the $\Delta_{\text{relay}} > 0$ parameter the minimum relaying delay introduced by an adversarial node; this delay is due to processing exclusively, it does not include any propagation time.

DEFINITION 11. A trace $\theta \in \Theta_{S,\mathcal{P}}$ is feasible with respect to an adversary model $\mathcal{A}_{\Delta_{\text{relay}}}$, if:

1. $\forall A; t :: \text{Bcast}(m) \in \theta, A \notin V_{\text{adv}}$
2. $\forall A; t :: \text{Dcast}(\alpha, \beta, m) \in \theta, \exists B \in V_{\text{adv}}, \exists \delta \geq \Delta_{\text{relay}} + \frac{\text{dist}(A,B)}{v_{\text{adv}}}, \exists C \in V, B; t - \delta :: \text{Receive}(C, m) \in \theta$

We denote the set of all traces feasible with respect to a setting S , MT-protocol model \mathcal{P} , and adversary model $\mathcal{A}_{\Delta_{\text{relay}}}$ by $\Theta_{S,\mathcal{P},\mathcal{A}_{\Delta_{\text{relay}}}}$.

Cond. 1 of Def. 11 is only to facilitate the presentation of proofs in subsequent sections, stating that adversarial nodes take only **Dcast** actions. Cond. 2 states that every message sent by an adversarial node is necessarily a replay of a message m that either this or another adversarial node received. In addition, the delay between receiving m and re-sending it, or more precisely the difference between the start times of the corresponding events, needs to be at least Δ_{relay} , plus the propagation delay across the adversary channel in case another adversarial node received the relayed message.

From $\mathcal{A}_{\Delta_{\text{relay}}}$, we derive two weaker adversary models, $\mathcal{A}'_{\Delta_{\text{relay}}}$ and $\mathcal{A}''_{\Delta_{\text{relay}}}$, defined next. Model $\mathcal{A}'_{\Delta_{\text{relay}}}$ restricts adversarial nodes to broadcasts, while $\mathcal{A}''_{\Delta_{\text{relay}}}$ precludes adversarial nodes from utilizing an adversary channel. As it will become clear in Sec. 3, all these adversary models are valuable for the impossibility result, and their weakness *strengthens* the impossibility result.

DEFINITION 12. A trace $\theta \in \Theta_{S,\mathcal{P}}$ is feasible with respect to an adversary model $\mathcal{A}'_{\Delta_{\text{relay}}}$, if:

1. $\forall A; t :: \text{Dcast}(\alpha, \beta, m) \in \theta, A \notin V_{\text{adv}}$
2. $\forall A; t :: \text{Bcast}(m) \in \theta \in \theta, \exists B \in V_{\text{adv}}, \exists \delta \geq \Delta_{\text{relay}} + \frac{\text{dist}(A,B)}{v_{\text{adv}}}, \exists C \in V, B; t - \delta :: \text{Receive}(C, m) \in \theta$

DEFINITION 13. A trace $\theta \in \Theta_{S,\mathcal{P}}$ is feasible with respect to an adversary model $\mathcal{A}''_{\Delta_{\text{relay}}}$, if:

1. $\forall A; t :: \text{Bcast}(m) \in \theta, A \notin V_{\text{adv}}$
2. $\forall A; t :: \text{Dcast}(\alpha, \beta, m) \in \theta, \exists \delta \geq \Delta_{\text{relay}}, \exists C \in V, A; t - \delta :: \text{Receive}(C, m) \in \theta$

2.7 Neighbor Discovery Specification

The ability to communicate directly, without the intervention or 'assistance' of relays, is expressed in our model by a link being up, thus the following definition:

DEFINITION 14. Node A is a neighbor of node B in setting S at time t , if $t :: A \rightarrow B$. If $t :: A \leftrightarrow B$ we will say that nodes A and B are neighbors at time t .

For simplicity of presentation, we use the " $t :: A \rightarrow B$ " notation to denote neighbor relation as well as the link relation. Having defined the neighbor relation, we are ready to present the formal specification of secure neighbor discovery.

DEFINITION 15. A protocol model \mathcal{P} satisfies (solves) two-party neighbor discovery for an adversary model \mathcal{A} , if the following properties are both satisfied:

- ND1 $\forall S \in \Sigma, \forall \theta \in \Theta_{S,\mathcal{P},\mathcal{A}}, \forall A, B \in V_{\text{cor}}, A; t :: \text{Neighbor}(B, t') \in \theta \implies t' :: B \rightarrow A$
- ND2 $\forall d \in (0, \mathbf{R}], \forall A, B \in \mathbb{V}, A \neq B, \exists S \in \Sigma, V = V_{\text{cor}} = \{A, B\} \wedge \text{dist}(A, B) = d \wedge \mathbb{R}_{\geq 0} :: A \leftrightarrow B \wedge \exists \theta \in \Theta_{S,\mathcal{P},\mathcal{A}}, A; t :: \text{Neighbor}(B, t') \in \theta$

Intuitively, property ND1 requires that if a node accepts some other correct node B as neighbor at time t' , then B is actually a neighbor at that time. Property ND2 complements ND1, assuring that the protocol is *usable*: it requires that for every distance d in the desired ND range \mathbf{R} , there should be at least some setting, in which the protocol is able to conclude that a node is a neighbor (in some, not all executions);

this setting should contain exactly two nodes at distance d , being neighbors, and both *correct*. The “two nodes setting” requirement clarifies why we call this *two-party* ND. The ND2 property is the least one can require from a usable two-party ND protocol, which makes the impossibility result in Sec. 3 more meaningful: having this with respect to a weak property implies impossibility for any stronger property.

3. IMPOSSIBILITY RESULT FOR MT PROTOCOLS

We show in this section that no message and time (MT) protocol can solve the neighbor discovery problem as specified by Def. 15. We base the proof on the intuitive fact that it is impossible for a correct node to distinguish between different settings based on an MT local view. Lemma 1 captures exactly this, with the impossibility result in Thm. 1 stemming from showing two settings which are indistinguishable by a correct node, one in which two nodes are neighbors and one where they are not. We elaborate on the assumptions and implications of this result in Sec. 6.

LEMMA 1. *Let \mathcal{P} be a MT-protocol model, \mathcal{S} and \mathcal{S}' be settings such that $V_{\text{cor}} = V'_{\text{cor}}$, and $\theta \in \Theta_{\mathcal{S}, \mathcal{P}}$ and $\theta' \in \Theta_{\mathcal{S}'}$ be traces such that local traces $\theta|_A = \theta'|_A$ for all $A \in V_{\text{cor}}$. Then θ' is feasible with respect to MT-protocol model \mathcal{P} .*

THEOREM 1. *If $\Delta_{\text{relay}} < \frac{R}{v}$ then there exists no MT-protocol model which satisfies neighbor discovery for the adversary model $\mathcal{A}_{\Delta_{\text{relay}}}''$.*

PROOF. To prove that under the assumptions of the theorem no MT-protocol model can satisfy both ND1 and ND2, we show that any MT-protocol model that satisfies ND2 cannot satisfy ND1.

Take any MT-protocol model \mathcal{P} satisfying ND2. Pick some distance in the ND range. Property ND2 guarantees the existence of a setting such as the one shown in figure 2(a) (we denote it \mathcal{S}^a), and a trace $\theta \in \Theta_{\mathcal{S}^a, \mathcal{P}, \mathcal{A}_{\Delta_{\text{relay}}}}$ such that $A; t :: \text{Neighbor}(B, t') \in \theta$. As θ is feasible with respect to setting \mathcal{S}^a , this trace has to be of the form:

$$\begin{aligned} \theta = & \{A; t_i :: \text{Bcast}(m_i) \mid i \in I_A\} \cup \\ & \{B; t_i + \Delta :: \text{Receive}(A, m_i) \mid i \in I_A\} \cup \\ & \{B; t_i :: \text{Bcast}(m_i) \mid i \in I_B\} \cup \\ & \{A; t_i + \Delta :: \text{Receive}(B, m_i) \mid i \in I_B\} \cup \\ & \{A; t_i :: \text{Neighbor}(B, t'_i) \mid i \in J_A\} \cup \\ & \{B; t_i :: \text{Neighbor}(A, t'_i) \mid i \in J_B\} \end{aligned}$$

where $\Delta = \frac{\text{dist}^a(A, B)}{v}$, $t_i, t'_i \in \mathbb{R}_{\geq 0}$ and I_A, I_B, J_A, J_B are pairwise disjoint index sets with $J_A \neq \emptyset$ (all the other index sets can be empty).

In setting \mathcal{S}^b , shown in figure 2(b), we have $\mathbb{R}_{\geq 0} :: B \leftrightarrow A$. Consider the following trace, which is essentially the same

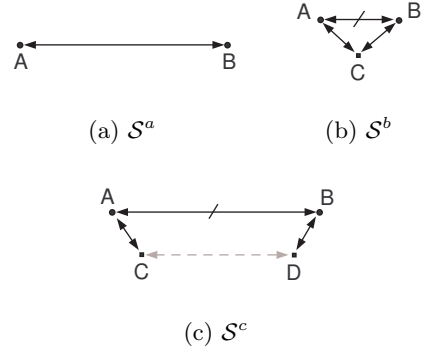


Figure 2: Settings used in the impossibility result proof. Settings $\mathcal{S}^a = \langle \{A, B\}, \text{loc}^a, \text{type}^a, \text{link}^a \rangle$, $\mathcal{S}^b = \langle \{A, B, C\}, \text{loc}^b, \text{type}^b, \text{link}^b \rangle$ and $\mathcal{S}^c = \langle \{A, B, C, D\}, \text{loc}^c, \text{type}^c, \text{link}^c \rangle$. In all settings, nodes A and B are correct, nodes C and D are adversarial. The location functions are such that $\text{dist}^b(A, C) + \text{dist}^b(B, C) + v\Delta_{\text{relay}} \leq \text{dist}^a(A, B) \leq R$ and $\text{dist}^c(A, C) + \text{dist}^c(D, B) + \frac{v}{v_{\text{adv}}} \text{dist}^c(C, D) + v\Delta_{\text{relay}} \leq \text{dist}^a(A, B)$. The state of links does not change over time and is shown in the figure. The dashed arrow in figure (c) denotes the adversarial channel.

as θ , but for node C relaying all the communication between nodes A and B :

$$\begin{aligned} \theta' = & \{A; t_i :: \text{Bcast}(m_i) \mid i \in I_A\} \cup \\ & \{C; t_i + \delta_1 :: \text{Receive}(A, m_i) \mid i \in I_A\} \cup \\ & \{C; t_i + \delta_2 :: \text{Dcast}(0, \pi, m_i) \mid i \in I_A\} \cup \\ & \{B; t_i + \Delta :: \text{Receive}(C, m_i) \mid i \in I_A\} \cup \\ & \{B; t_i :: \text{Bcast}(m_i) \mid i \in I_B\} \cup \\ & \{C; t_i + \delta_3 :: \text{Receive}(B, m_i) \mid i \in I_B\} \cup \\ & \{C; t_i + \delta_4 :: \text{Dcast}(-\pi, \pi, m_i) \mid i \in I_B\} \cup \\ & \{A; t_i + \Delta :: \text{Receive}(C, m_i) \mid i \in I_B\} \cup \\ & \{A; t_i :: \text{Neighbor}(B, t'_i) \mid i \in J_A\} \cup \\ & \{B; t_i :: \text{Neighbor}(A, t'_i) \mid i \in J_B\} \end{aligned}$$

where $\delta_1 = \frac{\text{dist}^b(A, C)}{v}$, $\delta_2 = \Delta - \frac{\text{dist}^b(C, B)}{v}$, $\delta_3 = \frac{\text{dist}^b(B, C)}{v}$ and $\delta_4 = \Delta - \frac{\text{dist}^b(C, A)}{v}$.

It is simple to check that this trace is feasible with respect to setting \mathcal{S}^b . It is also feasible with respect to MT-protocol model \mathcal{P} : as $\theta|_{A, t} = \theta'|_{A, t}$ and $\theta|_{B, t} = \theta'|_{B, t}$, this follows from Lem. 1. Finally, θ' is feasible with respect to the adversary model $\mathcal{A}_{\Delta_{\text{relay}}}$, because $\delta_2 - \delta_1 = \delta_4 - \delta_3 \geq \Delta_{\text{relay}}$. Therefore θ' belongs to $\Theta_{\mathcal{S}^b, \mathcal{P}, \mathcal{A}_{\Delta_{\text{relay}}}}$ and together with \mathcal{S}^b forms the counterexample that we were looking for: MT-protocol model \mathcal{P} does not satisfy ND1. As \mathcal{P} was chosen arbitrarily, this concludes the proof. \square

We can use the same technique (using settings \mathcal{S}^a and \mathcal{S}^c , illustrated in Fig.2) to prove a corresponding theorem for the adversary model $\mathcal{A}_{\Delta_{\text{relay}}}^T$:

THEOREM 2. *If $\Delta_{\text{relay}} < \frac{R}{v}$ then there exists no MT-protocol model \mathcal{P} which satisfies neighbor discovery for the adversary model $\mathcal{A}_{\Delta_{\text{relay}}}^T$.*

4. MT PROTOCOL SOLVING ND

Thm. 1 considers adversarial nodes that relay messages with a delay smaller than $\frac{R}{v}$. In this section we demonstrate a specific MT-protocol (we denote it as \mathcal{P}^T) which satisfies ND (Def. 15) if the minimum relaying delay incurred by adversarial nodes is greater than $\frac{R}{v}$. The proof of \mathcal{P}^T correctness is sketched in App. B.

Protocol. Informally, the \mathcal{P}^T protocol requires nodes to transmit authenticated messages containing a time-stamp set at the time of sending. Upon receipt of such a message, a receiver checks its “freshness” by verifying that the message time-stamp is within a threshold of the receiver’s current time. If so, it accepts the message creator as a neighbor. Note that this protocol is essentially the *temporal packet leash* proposed by Hu, Perrig and Johnson in [13].

Message space. We specify the message space relevant to this particular MT protocol to be:

$$\{\text{auth}_A(t)\}_{A \in V, t \in \mathbb{R}_{\geq 0}} \subseteq \mathbb{M}$$

with $\text{auth}_A(x)$ denoting that the content of each message x is authenticated by node A . We do not dwell on which cryptographic primitive (e.g., digital signature or message authentication code) is used to that end. We call the message $\text{auth}_A(t)$ a *beacon message*, and t the *beacon-time*.

Feasibility. Below we define feasibility w.r.t. protocol \mathcal{P}^T described informally above⁵.

DEFINITION 16. *A trace $\theta \in \Theta_S$ is feasible with respect to \mathcal{P}^T , if the following conditions are satisfied:*

1. $\forall A \in V_{\text{cor}}, \forall A; t_1 :: \text{Bcast}(\text{auth}_B(t)) \in \theta, (B = A) \wedge (t = t_1)$
2. $\forall A \in V_{\text{cor}}, \forall A; t_0 :: \text{Neighbor}(B, t_1) \in \theta, \exists C \in V, (A; t_1 :: \text{Receive}(C, \text{auth}_B(t)) \in \theta) \wedge (t_1 - t \leq \frac{R}{v}) \wedge (t_0 > \text{end}(A; t_1 :: \text{Receive}(C, \text{auth}_B(t))))$

Cond. 1 ensures that a correct node only broadcasts beacon messages that are authenticated by itself and have beacon-time set to the start of the beacon sending time. Recall that correct nodes have synchronized clocks, otherwise they

⁵For clarity and brevity, we define this “from scratch,” rather than specifying an MT protocol model according to Def. 9 and relying on Def. 10 for feasibility.

cannot be considered correct. Cond. 2 ensures that a correct A accepts B as a neighbor only after it receives and deems fresh a beacon generated by B .

Adversary Model. Towards proving that \mathcal{P}^T solves the ND problem, we need to develop a stronger than $\mathcal{A}_{\Delta_{\text{relay}}}^T$ adversary model. This is necessary as proving that a protocol is secure against a weak adversary would be of little value. The new adversary model, $\mathcal{A}_{\Delta_{\text{relay}}}^T$, allows not only message relay but also generation and transmission of any message, as long as the employed cryptosystem is not broken (this approach is compliant with the classical Dolev-Yao model [6]).

DEFINITION 17. *A trace $\theta \in \Theta_{S, \mathcal{P}^T}$ is feasible with respect to an adversary model $\mathcal{A}_{\Delta_{\text{relay}}}^T$, if:*

1. $\forall A; t :: \text{Bcast}(m) \in \theta, A \notin V_{\text{adv}}$
2. $\forall A \in V_{\text{adv}}, \forall A; t_1 :: \text{Dcast}(\alpha, \beta, \text{auth}_B(t)) \in \theta, (B \in V_{\text{adv}}) \vee (\exists C \in V_{\text{adv}}, \exists \delta \geq \Delta_{\text{relay}} + \frac{\text{dist}(C, A)}{v_{\text{adv}}}, \exists D \in V, C; t_1 - \delta :: \text{Receive}(D, \text{auth}_B(t)) \in \theta)$

Cond. 1 simplifies presentation mandating that adversarial nodes do not use the **Bcast** primitive. Nonetheless, this is not a limitation because **Bcast**(m) is equivalent to a **Dcast**($0, 2\pi, m$), by which we mean that it triggers exactly the same **Receive**(m) events. Cond. 2 ensures that an adversarial node is allowed to send any message as long as it is authenticated by an adversarial node (itself or other). This implies that adversarial nodes can share cryptographic keys or any material used for authentication. Furthermore, Cond. 2 reflects that the adversary cannot forge authenticated messages: it ensures a message sent by an adversarial node, and authenticated by a correct node must be a relayed one. In other words, some (possibly other) adversarial node must have received this message earlier, at least Δ_{relay} plus the propagation time between the two nodes (over the adversarial channel).

THEOREM 3. *If $\Delta_{\text{relay}} \geq \frac{R}{v}$ then \mathcal{P}^T satisfies neighbor discovery for the adversary model $\mathcal{A}_{\Delta_{\text{relay}}}^T$.*

5. MTL PROTOCOL SOLVING ND

Next, we consider message, time and location (MTL) protocols, allowing nodes to be more powerful than those in the MT protocol class. Our result reflects this, showing that if $v = v_{\text{adv}}$, an MTL-protocol, we denote as \mathcal{P}^{GT} , solves ND regardless of how small Δ_{relay} is. The reason why the impossibility theorem fails can be traced back to Lem. 1: even given identical local traces, correct nodes can resort to location information to distinguish setting \mathcal{S}^a from \mathcal{S}^b . The proof is similar to the MT case, sketched in App. B.

Protocol. Informally, the \mathcal{P}^{GT} protocol requires that nodes send authenticated messages containing a time-stamp set at the time of sending and their own location. Upon receipt of such a message m sent from a node B , the receiver A calculates two estimates of the A, B distance. The first estimate

is based on the difference of its own clock at reception time (the start of reception) and m 's time-stamp. The second one is calculated with the help of the location in m and A 's location. If the two distance estimates are equal, and m is authenticated, A accepts B as a neighbor. Note that this protocol is a variation of the *geographical packet leash* [13].

Message space. We specify the message space as follows:

$$\{\text{auth}_A(t, l)\}_{A \in \mathbb{V}, t \in \mathbb{R}_{\geq 0}, l \in \mathbb{R}^2} \subseteq \mathbb{M}$$

We call the message $\text{auth}_A(t, l)$ a *beacon message*, t the *beacon-time* of the message, and l the *beacon-location* of the message.

Feasibility. With the following we define feasibility w.r.t. to \mathcal{P}^{GT} .

DEFINITION 18. A trace $\theta \in \Theta_S$ is feasible with respect to \mathcal{P}^{GT} , if the following conditions are satisfied:

1. $\forall A \in V_{\text{cor}}, \forall A; t_1 :: \text{Bcast}(\text{auth}_B(t, l)) \in \theta,$
 $B = A \wedge t = t_1 \wedge l = \text{loc}(A)$
2. $\forall A \in V_{\text{cor}}, \forall A; t_0 :: \text{Neighbor}(B, t_1) \in \theta, \exists C \in V,$
 $A; t_1 :: \text{Receive}(C, \text{auth}_B(t, l)) \in \theta \wedge t_1 - t = \frac{d_2(\text{loc}(A), l)}{\mathbf{v}} \wedge$
 $t_0 > \text{end}(A; t_1 :: \text{Receive}(C, \text{auth}_B(t, l)))$

Adversary model. The adversary model, $\mathcal{A}_{\Delta_{\text{relay}}}^{\text{GT}}$, is almost identical to $\mathcal{A}_{\Delta_{\text{relay}}}^{\text{T}}$ but for the format of beacon messages.

DEFINITION 19. A trace $\theta \in \Theta_{S, \mathcal{P}^{\text{GT}}}$ is feasible with respect to the adversary model $\mathcal{A}_{\Delta_{\text{relay}}}^{\text{GT}}$, if:

1. $\forall A; t :: \text{Bcast}(m) \in \theta, A \notin V_{\text{adv}}$
2. $\forall A \in V_{\text{adv}}, \forall A; t_1 :: \text{Dcast}(\alpha, \beta, \text{auth}_B(t, l)) \in \theta,$
 $(B \in V_{\text{adv}}) \vee (\exists C \in V_{\text{adv}}, \exists \delta \geq \Delta_{\text{relay}} + \frac{\text{dist}(C, A)}{\mathbf{v}_{\text{adv}}},$
 $\exists D \in V, C; t_1 - \delta :: \text{Receive}(D, \text{auth}_B(t, l)) \in \theta)$

THEOREM 4. If $\mathbf{v} = \mathbf{v}_{\text{adv}}$ and $\Delta_{\text{relay}} > 0$ then \mathcal{P}^{GT} satisfies neighbor discovery for the adversary model $\mathcal{A}_{\Delta_{\text{relay}}}^{\text{GT}}$.

6. DISCUSSION

6.1 Implications

The impossibility theorems and the related $\Delta_{\text{relay}} < \frac{\mathbf{R}}{\mathbf{v}}$ constraint imply that, for a given \mathbf{R} and \mathbf{v} , any MT protocol for ND can be successfully attacked by an adversary capable of relaying with delay smaller than $\frac{\mathbf{R}}{\mathbf{v}}$, independently of any considerations on the node implementation: accuracy of clocks, processing power of nodes, etc. Nonetheless, in general the system designer can select a ND range value \mathbf{R} , trading off practicality for security. A low \mathbf{R} forces the

adversary to relay messages faster to succeed, but it also precludes the discovery of nodes that are directly reachable but farther than \mathbf{R} . Inversely, a large \mathbf{R} , for example, matching the node wireless capabilities results in more effective yet less secure ND. In fact, as $\mathbf{R} \rightarrow \infty$, it becomes clear that one cannot expect to have an MT protocol securing ND for arbitrarily distant nodes. On the contrary, the magnitude of \mathbf{R} can be essentially ignored for the analysis and design of MTL protocols.

We can compare our findings with the feasible relaying delay reported by [19], that is 40ns . If we consider even the relatively short-range 802.11 radios, communicating typically at 300m, we can see the impossibility result coming into the picture: $40\text{ns} < \frac{300\text{m}}{c} \approx 1\mu\text{s}$.

Simple quantitative results. Thm. 1 and Thm. 2 show that it is impossible to secure ND even if the adversary cannot utilize an adversarial channel for the communication of the nodes it controls (but in that case it uses directional antennas). However, quantitatively, the relative magnitude of \mathbf{v}_{adv} and \mathbf{v} , the signal propagation velocity across the system wireless channel and the adversary channel respectively, determines the impact of the adversary.

To illustrate this, we consider first an $\mathcal{A}_{\Delta_{\text{relay}}}^{\text{adv}}$ adversary and setting \mathcal{S}^b in Fig. 2, with A, B correct and C adversarial nodes, for which it holds $\text{dist}^b(A, C) + \text{dist}^b(B, C) + \mathbf{v}\Delta_{\text{relay}} \leq \mathbf{R}$. This equation yields, when combined with the triangle inequality $\text{dist}^b(A, B) \leq \text{dist}^b(A, C) + \text{dist}^b(B, C)$, that $\text{dist}^b(A, B) \leq \mathbf{R} - \mathbf{v}\Delta_{\text{relay}}$. Note that the relative locations and thus the distance of A and B are not controlled by the adversary. This implies that adversary can violate ND1 only if the distance between A and B is smaller than $\mathbf{R} - \mathbf{v}\Delta_{\text{relay}}$ and C is conveniently located.

On the other hand, for $\mathcal{A}_{\Delta_{\text{relay}}}^{\text{adv}}$ and setting \mathcal{S}^c in Fig. 2, $\text{dist}^c(A, C) + \text{dist}^c(D, B) + \frac{\mathbf{v}}{\mathbf{v}_{\text{adv}}} \text{dist}^c(C, D) + \mathbf{v}\Delta_{\text{relay}} \leq \mathbf{R}$. Utilizing this and the triangular inequality twice, that is, $\text{dist}^c(A, B) \leq \text{dist}^c(A, C) + \text{dist}^c(C, D) + \text{dist}^c(D, B)$, we get $\text{dist}^c(A, B) \leq \frac{\mathbf{v}_{\text{adv}}}{\mathbf{v}}(\mathbf{R} - \mathbf{v}\Delta_{\text{relay}})$. If this inequality holds for the distance of the correct nodes A, B , the adversary can succeed with the use of an adversarial channel and two nodes C, D . It is interesting that the bound on $\text{dist}^c(A, B)$ is multiplied by a factor of $\frac{\mathbf{v}_{\text{adv}}}{\mathbf{v}}$. In other words, if $\mathbf{v} \ll \mathbf{v}_{\text{adv}}$, as it holds, for example, for ultrasound and radio frequency velocities, the use of the adversarial channel magnifies the impact on ND: the adversary can mislead nodes at remote locations (and thus unable to communicate directly) that they are neighbors. Thus, whenever possible, the system designer should aim at having $\mathbf{v} = c$, which she can expect to be the choice of the adversary. This is further strengthened by the fact that the \mathcal{P}^{GT} can be proven correct only if $\mathbf{v} = \mathbf{v}_{\text{adv}}$.

Adversary models. Formally, the relative strength of adversary models is defined in Def. 20 below.

DEFINITION 20. Adversary model \mathcal{A}_1 is (non-strictly) weaker than adversary model \mathcal{A}_2 ($\mathcal{A}_1 \leq \mathcal{A}_2$), if for every setting \mathcal{S}

and every protocol model \mathcal{P} it holds that $\Theta_{\mathcal{S}, \mathcal{P}, \mathcal{A}_1} \subseteq \Theta_{\mathcal{S}, \mathcal{P}, \mathcal{A}_2}$.

We provide without proof, as it is trivial, the relation between the considered adversary models presented below:

$$\begin{aligned} \mathcal{A}'_{\Delta_{\text{relay}}} &\preceq \mathcal{A}_{\Delta_{\text{relay}}} \leq \mathcal{A}^{\top}_{\Delta_{\text{relay}}} \\ \mathcal{A}''_{\Delta_{\text{relay}}} &\leq \mathcal{A}_{\Delta_{\text{relay}}} \end{aligned}$$

We use a different notation, $\mathcal{A}'_{\Delta_{\text{relay}}} \preceq \mathcal{A}_{\Delta_{\text{relay}}}$, as the “ \leq ” relation does not hold: in one case the adversarial nodes can only use **Bcast** and in the other only **Dcast**. However, **Bcast**(m) is equivalent to a **Dcast**($0, 2\pi, m$). Accordingly, we can define a renaming function ρ , and show that the \leq relation holds up to renaming: $\rho(\Theta_{\mathcal{S}, \mathcal{P}, \mathcal{A}'_{\Delta_{\text{relay}}}}) \subseteq \Theta_{\mathcal{S}, \mathcal{P}, \mathcal{A}_{\Delta_{\text{relay}}}}$.

The relation among adversary models is interesting because one can intuitively expect that if a protocol \mathcal{P} can solve ND for \mathcal{A}_1 , it can also solve ND for a weaker adversary model \mathcal{A}_2 . One can prove this under the assumption that the adversary model allows the adversarial nodes to remain silent⁶, which is the case for all the adversary models that we consider. Thus, our impossibility result, proven for the minimal elements, and the proof of correctness of protocol \mathcal{P}^{\top} for the maximal element, hold for all adversary models considered in this paper. This clarifies that Δ_{relay} is the most significant factor affecting security of ND, as opposed to the ability to use directional antennas, the adversary channel, or generate arbitrary messages (in a Dolev-Yao fashion).

6.2 Modeling assumptions

Our assumptions on wireless communication, protocols, and adversarial behavior, and our ND specification aim at a simple model. Nonetheless, these assumptions do not impair the generality and meaningfulness of our results. The discussion below establishes this mostly with respect to the impossibility result, as it is easy to see that most of these simplifying assumptions do not affect the ND protocols we model and prove correct.

Protocol model. Recall that our definition of a protocol model only requires that the behavior of the protocol is determined by the local view. This is much broader than the typical approach, in which a protocol is modeled by a Turing machine. But since our definition is an over-approximation, our impossibility result remains valid for more realistic protocol models.

Settings and traces. We emphasize that the quite general forms of settings (correct nodes being able to communicate at arbitrary distances), and Medium Access Control modeling (Def. 4 not prohibiting a correct node from sending and receiving an arbitrary number of messages at the same time) is *not* essential to the impossibility result. It is possible to add additional constraints to make the model more realistic, but this would impair generality and clarity.

⁶In general, one might imagine an adversarial model for which this would not hold, but such a model would be of no practical importance.

Events. In Sec. 2, we model correct nodes equipped with omnidirectional antennas. We can extend our model so that correct nodes use directional antennas, but from the structure of the impossibility result proof it should be clear that this would not lift the impossibility. However, mounting a successful relay attack would require adversarial node(s) to be located on or close to the line connecting A and B .

In Sec. 2, we model success and failure (in fact, complete unawareness of failure) in receiving a message, but not the ability of receiver to detect a transmission (wireless medium activity) without successfully decoding the message. An extension of our model to include this is straightforward and would not affect the impossibility result. Intuitively, if nodes were able to solve the ND problem if they cannot decode all the messages they receive, then they would also be able to solve ND when all messages are received correctly.

We emphasize that the above argument relies on the assumption that nodes cannot control how messages they send are received by other nodes, that is, they do not have control over their wireless transmission power. However, if nodes had this ability, the very notion of neighborhood would change, and our model would need to change as well. We will investigate this in future work.

ND specification. In light of the impossibility result, one could consider an alternative, less restrictive neighbor discovery specification, notably, the already mentioned *multi-party* ND that requires the participation of more than two nodes to securely conclude on a neighbor relation. This is an interesting direction resonating with emergent properties of ad-hoc networks [9]. Technically, this ND specification would differ in the ND2 property, where the requirement that the protocol needs to work for some *two-node* setting would be changed to an *arbitrary* setting. As discuss in Sec. 7, there exist protocols in the literature related to our notion of multi-party ND, but they work under weaker adversary models. Whether some other MT protocol can solve multi-party ND in our model, is an open question we plan to investigate in future work.

7. RELATED WORK

The primary wormhole prevention mechanism is based on *distance bounding*, which was first proposed by Brands and Chaum in [2] to thwart a relay attack between two correct nodes, also termed as *mafia fraud*. Essentially, distance bounding estimates the distance between two nodes, with the guarantee it is not smaller from their real distance. Subsequent proposals contributed in aspects such as mutual authentication [21], efficiency [10], and resistance to execution of the protocol with a colluding group of adversarial nodes [3, 19]. In the latter, the attack termed as *terrorist fraud* is thwarted under the assumption that adversarial nodes do not expose their private cryptographic material; if not, one adversarial node can undetectably impersonate another one and successful stage a terrorist fraud. *Authenticated ranging*, proposed by Čapkun and Hubaux in [22], lifts the technically non-trivial requirement of rapid response (present in all the above protocols), at the expense of not being resilient to a *distance fraud*, when the protocol is executed with a *single, non-colluding* adversarial node [3].

This group of protocols, in which temporal packet leases [13] and TrueLink [8] (both not resistant to the distance fraud) can be included, was the main inspiration for our investigation that led to a very general impossibility result. This clearly shows these protocols solve a slightly different problem than neighbor discovery: they can be used to detect and prevent long-range relay attacks, a defense sufficient for applications such as RFID building access control, but clearly *not* all relay attacks.

Another group of ND mechanisms is based on location, with geographical packet leases [13] the primary representative. The impossibility result does not apply here, since MT protocols are not location-aware. Indeed, we prove that \mathcal{P}^{GT} , an MTL protocol, *can* solve ND. We emphasize that \mathcal{P}^{GT} is different from geographical packet leases, because it requires clock synchronization as tight as that for temporal packet leases. Essentially, \mathcal{P}^{GT} is a combination of temporal and geographical leases. Upon careful inspection of the literature, there exist prior passages seemingly cluing or relating to this idea: the introduction of [12] or the discussion of combining a so-called node-centric localization scheme with distance bounding techniques [23]. Nonetheless, to the best of our knowledge, we are the first to explicitly point out the advantages, over other approaches for secure ND, of combining location information with tight temporal bounds. We note that the authors of [13] mention the obstacle problem, but only in the case of geographical packet leases. Unfortunately, the solution that they propose – having a radio propagation model at every node – is not applicable in most scenarios.

The approach of Poovendran and Lazos [17] can be seen as an extension of a location based scheme: a few trusted nodes (*guards*) are aware of their location, transmit it periodically in beacons, and all other nodes determine their neighbors based on whether they received sufficiently many common beacons. This scheme is a multi-party ND protocol and thus our impossibility result does not apply. Unfortunately, [17], from the perspective of our approach, has some serious drawbacks. Most notably, it relies on the “no obstructions” assumption – nodes which are close but cannot communicate can be tricked into establishing a neighbor relation. In addition, adversarial nodes are rather limited in their behavior: one can see an attack against this scheme, in particular Claim 2, when adversarial nodes are allowed to selectively relay beacon messages.

A scheme using directional antennas was proposed by Hu and Evans in [12], with the interesting property it can be used as a two-party ND protocol, or as a multi-party ND protocol with additional nodes serving as *verifiers* of neighbor relations. In the two-party operation the scheme has security weaknesses that the multi-party version is called upon to remedy. In the latter case, our impossibility result does not apply directly. Nonetheless, significant security problems remain, with the scheme oblivious to obstacles and the adversary model limited. As the authors point out, a successful attack can be mounted if more than two adversarial nodes collaborate. Recall that in our proofs we allow for arbitrary node collaboration (or collusion).

A different approach to secure neighbor discovery could ex-

plot radio frequency fingerprinting (RFF) [5]: devices from same production line are not identical, but rather signals each one emits may have unique identifiable features. If those can be identified upon reception of a message, it becomes impossible for an adversarial node to relay any message undetected. If such a scheme were in place, our impossibility result would not apply. The reason is that impossibility hinges on the very fact that a correct node cannot identify how a message was received. This essentially allows the adversary to relay wireless transmissions (messages). However, it is questionable if RFF can be used to secure ND. Investigations with different types of devices, e.g., [18] or [20], show classification success rate around 90% in laboratory conditions. At the same time, findings such as “...radios were found to have fingerprints that were virtually indistinguishable from each other, making the identification process more difficult, if not impossible...” [7] clue on unresolved limitations.

A large body of work on formal reasoning on cryptographic protocols exists, yet the classical cryptographic protocols live in the Internet: thus these methods are agnostic to the characteristics of the communication medium, especially a wireless one. On the other hand, there has been a rising interest in formalizing analysis of security protocols in wireless networks. The problem of distance bounding has been treated formally in [14], while other works were concerned with routing [15, 1, 16, 25] or local area networking [11]. These works are concerned with different problems and their approaches are not amenable to reason about secure neighbor discovery.

8. CONCLUSIONS

We investigate the problem of secure neighbor discovery (ND) in wireless networks. We build a formal framework, and provide a specification of neighbor discovery, or, more precisely, its most basic variant, two-party ND. We consider two general classes of protocols: message and time (MT) protocols and message, time and location (MTL) protocols. For the MT class, we identify a fundamental limitation governed by a threshold value depending on the ND range: we prove that no MT protocol can solve the ND problem if and only if adversarial nodes can relay messages faster than this threshold. This result is a useful measure of the ND security achieved by MT protocols and leads us to investigate other classes of protocols.

In particular, we prove that no such limitation exists for the class of MTL protocols: they can solve the ND problem for any adversary, as long as the time and location measurements are accurate enough. The protocols we provide and show solving the ND problem are very simple if not the simplest possible that allow positive results. In future work, we will focus on a larger spectrum of protocols, most notably multi-party neighbor discovery, as well as model additional aspects, such as the ability of nodes of controlling their transmission power.

9. REFERENCES

- [1] Gergely Ács, Levente Buttyán, and István Vajda. Provably secure on-demand source routing in mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, 5(11):1533–1546, 2006.

- [2] Stefan Brands and David Chaum. Distance-bounding protocols. In *EUROCRYPT '93: Workshop on the theory and application of cryptographic techniques on Advances in cryptology*, pages 344–359, Secaucus, NJ, USA, 1994.
- [3] Laurent Bussard. *Trust establishment protocols for communicating devices*. PhD thesis, Thesis, October 2004.
- [4] Levente Buttyán, László Dóra, and István Vajda. Statistical wormhole detection in sensor networks. In Refik Molva, Gene Tsudik, and Dirk Westhoff, editors, *ESAS*, volume 3813 of *Lecture Notes in Computer Science*, pages 128–141, 2005.
- [5] Howard C. Choe, Clark E. Poole, Andrea M. Yu, and Harold H. Szu. Novel identification of intercepted signals from unknown radio transmitters. In H. H. Szu, editor, *Proc. SPIE Vol. 2491, p. 504-517, Wavelet Applications II, Harold H. Szu; Ed.*, volume 2491 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, pages 504–517, April 1995.
- [6] Danny Dolev and Andrew Chi-Chih Yao. On the security of public key protocols (extended abstract). In *FOCS*, pages 350–357, 1981.
- [7] K. J. Ellis and Nur Serinken. Characteristics of radio transmitter fingerprints, 2001. *Radio Science*, Volume 36, Issue 4, p. 585-598.
- [8] Jakob Eriksson, Srikanth V. Krishnamurthy, and Michalis Faloutsos. Truelink: A practical countermeasure to the wormhole attack in wireless networks. In *ICNP '06. Proceedings of the 2006 14th IEEE International Conference on Network Protocols*, pages 75–84, 2006.
- [9] Virgil D. Gligor. Security of emergent properties in ad-hoc networks (transcript of discussion). In Bruce Christianson, Bruno Crispo, James A. Malcolm, and Michael Roe, editors, *Security Protocols Workshop*, volume 3957 of *Lecture Notes in Computer Science*, pages 256–266, 2004.
- [10] Gerhard Hancke and Markus Kuhn. An RFID distance bounding protocol. In *Conference on Security and Privacy for Emerging Areas in Communication Networks – SecureComm 2005*, Athens, Greece, September 2005. IEEE.
- [11] Changhua He, Mukund Sundararajan, Anupam Datta, Ante Derek, and John C. Mitchell. A modular correctness proof of IEEE 802.11i and TLS. In *CCS '05: Proceedings of the 12th ACM conference on Computer and communications security*, pages 2–15, New York, NY, USA, 2005.
- [12] Lingxuan Hu and David Evans. Using directional antennas to prevent wormhole attacks. In *Symposium on Network and Distributed Systems Security (NDSS)*, 2004.
- [13] Yih-Chun Hu, Adrian Perrig, and David B. Johnson. Packet leashes: A defense against wormhole attacks in wireless networks. In *IEEE Conference on Computer Communications INFOCOM*, 2003.
- [14] Catherine Meadows, Radha Poovendran, Dusko Pavlovic, LiWu Chang, and Paul Syverson. Distance bounding protocols: Authentication logic analysis and collusion attacks. In *Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks*. Springer-Verlag, Series: Advances in Information Security, Vol. 30 Poovendran, Radha; Wang, Cliff; Roy, Sumit (Eds.), 2007.
- [15] Sebastian Nanz and Chris Hankin. A framework for security analysis of mobile wireless networks. *Theoretical Computer Science*, 367(1):203–227, 2006.
- [16] Panagiotis Papadimitratos, Zygmunt Haas, and Jean-Pierre Hubaux. How to Specify and How to Prove Correctness of Secure Routing Protocols for MANET. In *IEEE-CS Third International Conference on BroadBand Communications, Networks, and Systems*, 2006.
- [17] Radha Poovendran and Loukas Lazos. A graph theoretic framework for preventing the wormhole attack in wireless ad hoc networks. *Wireless Networks*, 13(1):27–59, 2007.
- [18] Kasper Bonne Rasmussen and Srdjan Čapkun. Implications of radio fingerprinting on the security of sensor networks. Technical Report 536, ETH Zürich, 2006.
- [19] Jason Reid, Juan M. Gonzalez Nieto, Tee Tang, and Bouchra Senadji. Detecting relay attacks with timing-based protocols. In *ASIACCS '07: Proceedings of the 2nd ACM symposium on Information, computer and communications security*, pages 204–213, New York, NY, USA, 2007.
- [20] Önder H. Tekbaş, Nur Serinken, and Oktay Üreten. An experimental performance evaluation of a novel radio-transmitter identification system under diverse environmental conditions. *IEEE Canadian Journal of Electrical and Computer Engineering*, 29, July 2004.
- [21] Srdjan Čapkun, Levente Buttyán, and Jean-Pierre Hubaux. SECTOR: secure tracking of node encounters in multi-hop wireless networks. In *SASN '03: Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks*, pages 21–32, New York, NY, USA, 2003.
- [22] Srdjan Čapkun and Jean-Pierre Hubaux. Secure Positioning in Wireless Networks. *IEEE Journal on Selected Areas in Communications (JSAC)*, 24(2):221–232, 2006.
- [23] Srdjan Čapkun, Mario Čagalj, and Mani Srivastava. Securing localization with hidden and mobile base stations. In *IEEE Conference on Computer Communications (INFOCOM)*, April 2006.
- [24] Weichao Wang and Bharat Bhargava. Visualization of wormholes in sensor networks. In *WiSe '04: Proceedings of the 2004 ACM workshop on Wireless security*, pages 51–60, New York, NY, USA, 2004.
- [25] Shahan Yang and John S. Baras. Modeling vulnerabilities of ad hoc routing protocols. In *SASN '03: Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks*, pages 12–20, New York, NY, USA, 2003.

APPENDIX

A. PROTOCOL DESIGN

In this section, we discuss some of the more important aspects for actual deployment of secure neighbor discovery protocols. First, we consider in this paper one side of ND: A discovers if B is a neighbor. However, with asymmetric

links, a dual problem exists: A discovers if it is a neighbor to B . The protocols we consider are not designed to solve this problem, but we note that challenge-response schemes, such as distance bounding protocols [2], can.

Moreover, we consider ND when both nodes running the ND protocol are correct. Removing this assumption implies that, for example, the \mathcal{P}^T protocol does not satisfy the ND specification: consider an adversarial node B that generates a message time-stamped in the future, passes this message to another adversarial node C , which in turn passes it to a correct node A that falsely accepts (a perhaps very remote) B as a neighbor. In Sec. 7 two protocols that solve this problem under a specific assumption are discussed.

As mobility was not included in our model, the protocols that we analyze can be considered secure as long as the node movement during the protocol execution is negligible. This is not a strong requirement, if we compare the typical speed at which nodes move (below the speed of sound in almost all cases) with the RF propagation speed. However, notably because some computational operations may be time-consuming, we plan to include mobility in our model in the future.

All the adversary models in this paper capture the technically feasible, yet non-trivial ability to send and receive messages at the same time. For a weaker security result, one could assume that an adversarial node must receive the whole message before it can relay it. For such an adversary, a protocol whose every messages duration is longer than $\frac{R}{v}$ would solve ND (by Thm. 3).

Similarly to the vision of the authors of [13], \mathcal{P}^T and \mathcal{P}^{GT} functionality could be integrated into every packet as a leash. Alternatively, ND beacons can be broadcasted periodically, with the neighbor relation interpolated in between received beacons. The former solution provides better security at expense of transmission overhead, while the latter might offer the adversary a window of opportunity to launch an attack if and only if the state of neighbor relation changes between two beacon broadcasts.

Definition 4 implies signal propagation over a straight line. In reality, this is not always the case, as two nodes could communicate even if there is no line-of-sight between them, and the signal is reflected several times on the way.⁷ For such additionally delayed links, \mathcal{P}^T and especially \mathcal{P}^{GT} could be unable to conclude successfully. Mitigating this effect relates to the discussion below on inaccuracies in time and location information these protocols need to cope with in practice.

Imperfect clocks and localization. We assumed to this point that correct nodes have accurate time and location information. However, inaccuracies are possible in reality: (i) time inaccuracies due to clock drifts, failure to synchronize clocks, as well as coarse-grained clocks, and (ii) location inaccuracies due to unavailability of infrastructure (e.g., Global Positioning System (GPS), or base stations) provid-

⁷We could include this phenomenon in our model, for example, by introducing an additional link-specific delay to the propagation time. This would not affect any of our results.

ing location information, malicious disruptions of infrastructure, and granularity and capabilities of self-localization sensors. As the \mathcal{P}^T and \mathcal{P}^{GT} protocols rely on distance estimates based on time and location measurements, their effectiveness can be affected by inaccuracies.

We model the impact of time inaccuracy by a parameter δ , such that *measured delay* = *real delay* + d , with $|d| \leq \delta$. Similarly, for location information, *measured distance* = *real distance* + sv , with $|s| \leq \tau$. We express the inaccuracy term sv as a function of delay (time), so that it is straightforward to consider the cumulative impact for the \mathcal{P}^{GT} protocol.

First, for \mathcal{P}^T , two correct neighbors at distance larger than $R - v\delta$ may fail to conclude they are neighbors, thus violating ND2. This can be addressed if $R' = R + v\delta$ is used in place of the ND range R . But then, if $\Delta_{\text{relay}} < \frac{R}{v} + \delta$, or $\Delta_{\text{relay}} < \frac{R'}{v}$, ND1 would be violated, that is, the adversary would mount a successful attack. In other words, time inaccuracies essentially decrease the ND security.

Naturally, the “idealized” version of \mathcal{P}^{GT} presented in Sec. 5 should be changed slightly when used in the real world: it should not check for *equality* of the time- and location-based estimates of distance, but rather *approximate equality*; otherwise ND2 will be violated. More precisely, these two estimates should be within $\delta + \tau$ of each other. But, again, ensuring practicality decreases security: if $\Delta_{\text{relay}} < 2(\delta + \tau)$, the adversary could violate ND1.

More generally, for MT protocols, no additional consideration with respect to the impossibility results is necessary, as $R \leq R'$. But for MTL protocols, the inaccuracies in time and location could be viewed as an impossibility factor: for given δ, τ , there is no protocol solving the ND problem if the adversary can relay with delay $\Delta_{\text{relay}} < 2(\delta + \tau)$. We emphasize however that the nature of this impossibility results differs, as it is not fundamental, as in the MT case, but can be mitigated by introducing more sophisticated technology and obtaining accurate time and location.

Finally, we note that accurate time and location information are not possible to achieve without specialized hardware. In addition, tight synchronization is nontrivial, but challenge-response protocols that do not need synchronized clocks can overcome this problem.

B. PROOF OF THEOREM 3

We sketch here the proof of Thm. 3. We note that the proof of Thm. 4 is very similar, and the proof Lem. 1 is straightforward; they are therefore omitted.

PROOF. (Sketch) To prove ND1: first, we use 16.2 to see that A received B ’s beacon m . If m was sent directly by a correct node, we conclude that link (B, A) is up (from 4.1, 16.1). Otherwise, a relay took place (17.2); but this is not possible because the Δ_{relay} delay induced by 17.2 contradicts 16.1 (applied for B) and 16.2 (for A) and the assumption $\Delta_{\text{relay}} > \frac{R}{v}$. To prove ND2: for every d , the setting can be S^a (Fig. 2(a)) and the trace can be: $\{B; 1:: \text{Bcast}(\text{auth}_B(1)), A; 1+\frac{d}{v}:: \text{Receive}(B, \text{auth}_B(1)), A; 2:: \text{Neighbor}(B, 1+\frac{d}{v})\}$ \square