# Alibi framework for identifying reactive jamming nodes in wireless LAN

Hoang Nguyen, Thadpong Pongthawornkamol and Klara Nahrstedt
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801
E-mail:{hnguyen5,tpongth2,klara}@illinois.edu

***Abstract*— Reactive jamming nodes are the nodes of the network that get compromised and become the source of jamming attacks. They assume to know any shared secrets and protocols used in the networks. Thus, they can jam very effectively and are very stealthy. We propose a novel framework for identifying the reactive jamming nodes in wireless LAN (WLAN). We rely on the half-duplex nature of nodes: they cannot transmit and receive at the same time. Thus, if a compromised node jams a packet, it cannot guess the content of the jammed packet. More importantly, if an honest node receives a jammed packet, it can prove that it cannot be the one jamming the packet by showing the content of the packet. Such proofs of jammed packets are called "alibis" - the key concept of our approach.**

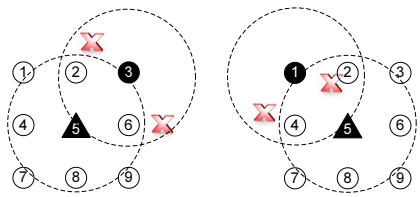
**In this paper, we present an alibi framework to deal with reactive jamming nodes in WLAN. We propose a concept of *alibi-safe* topologies on which our proposed identification algorithms are proved to correctly identify the attackers. We further propose a realistic protocol to implement the identification algorithm. The protocol includes a BBC-based timing channel for information exchange under the jamming situation and a similarity hashing technique to reduce the storage and network overhead. The framework is evaluated in a realistic TOSSIM simulation where the simulation characteristics and parameters are based on real traces on our small-scale MICAz test-bed. The results show that in reasonable dense networks, the alibi framework can accurately identify both non-colluding and colluding reactive jamming nodes.** [1]

## I. Introduction

Wireless communications are inherently vulnerable to jamming attacks due to the open and shared nature of wireless medium. In the jamming attack, an attacker injects a high level of noise into the wireless system which significantly reduces the signal to noise and interference ratio (SINR) and reducing probability of successful message receptions.

In this paper, we consider the problem of *identifying* compromised nodes who launch stealthy reactive jamming attacks in half-duplex single-channel wireless LAN. This is a very challenging problem because the attackers are assumed to know any shared secret and protocols in the network. They try to stay undetected as long as possible while maximizing the damage done to the network. To the best of our knowledge, none of the existing work can deal with this type of attackers in the context of WLAN (see Section VI) briefly due to two reasons. First, many approaches are only concerned about how to build jamming-resistant communications [1]–[5] without *identifying* the source of jamming. Jamming-resistant communications are necessary but not sufficient: as long as the jamming nodes are not identified, they always have effective jamming attacks on the network. Second, there are also several works on identifying mis-behaving nodes. However, because the attackers leave no identity information in the jammed packets (e.g., by corrupting the sender field), detection systems relying on identity clues to infer nodes causing the jammed packet do not work (e.g., [6], [7])

We propose an alibi framework to deal with this challenging problem. Alibi is "a form of defense whereby a defendant attempts to prove that he or she was elsewhere when the crime in question was committed". Because in a half-duplex wireless network where nodes cannot send and receive at the same time, attackers who jam will not be able to receive the content of jammed packets. Therefore, any nodes that can show proofs of corrupted packets are subject to getting an alibi in that time slot. In the long term, nodes who obtain least number of alibis are likely to get accused. Figure 1 illustrates how each honest node gets alibis on jamming events by node 5. In Figure 1(a), node 3's message is jammed. Node 2 and 6 receive a corrupted packet. If they both show the content corrupted packet, they can both claim an alibi. Similarly, in Figure 1(b), when node 1's message is jammed, node 2 and 4 will get an alibi by showing proofs of receiving an corrupted packet. In this manner, at some point each honest node in the network will get at least one alibi while the jammer (node 5) will have no alibis. Until then, the jammer can be identified.




(a) Node 3's message is jammed by node 5. Node 2, 6 get an alibi by showing the content of the corrupted packet.

(b) Node 1's message is jammed by node 5. Node 2, 4 gets an alibi by showing the content of the corrupted packet.

Fig. 1. Example of Alibi Scheme.

Even though the concept of alibi framework may appear

[1]This material is based upon work supported by the Department of Energy under Grant Award Number DE-OE0000097. This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

simple, there are numerous challenges to make it work. First, an attacker who jams in a time slot can always show a random corrupted packet content as a proof of reception. Second, collided packets are not different from jammed packets. That means, nodes may get "false" alibis resulted from collisions. Thus, there will be some noises in the detection process relying on number of alibis. Third, in an extreme situation where attackers decide to jam every single packet, the system cannot operate on the main channel anymore. Therefore, there has to be a jamming-resistant communication to exchange alibi information in the network. The alibi framework uses a BBC-based timing channel with zero shared-secret for nodes to communicate. Lastly, there might be multiple attackers in the system. A jamming action of one attacker may help other attackers to obtain alibis. The situation is even worse when they can collude with each other.

In our previous works [8], [9], we have considered a similar technique but in a different context: *multi-channel* wireless networks. Most importantly, in those works, we made an assumption that the corrupted packets received by nodes 2 and 6 in Figure 1(a) (or nodes 2 and 4 in Figure 1(b)) are *identical*. This assumption, however, does not always hold as shown in Section III. In this paper, we validate that assumption by a MICAz test-bed experiments and incorporate a much more realistic model of alibi into the framework. Our contributions can be summarized as follows. First, we carry out experiments on a MICAz test-bed to study the impact of reactive jamming attacks on the network performance and the hypothesis of packet content similarity under reactive jamming attacks (Section III). Second, based on the results from the experiments, we propose a concept of "alibi-safe" network topology where the alibi framework can identify attackers and corresponding alibi identification algorithms (Sections II and IV). We also prove the correctness of the alibi identification algorithms on alibi-safe topologies. To this end, we study the connection between physical network topologies and alibi-safe topologies (Section V-A). This connection is very important in helping network designer to deploy the wireless networks to defend against the reactive jamming attacks. Third, we design a practical alibi protocol including a BBC-based timing channel and a similarity hashing technique for more efficient proof exchange under the reactive jamming attacks. Lastly, we implement and evaluate the alibi framework on a large-scale networks in TOSSIM simulator (Section V). The implementation includes BBC-based timing channel, similarity hashing and the identification algorithms. We incorporate the small-scale MICAz experimental results including the packet error rate trace, the noise and the reception similarity trace into TOSSIM for accurate simulation results.

The rest of the paper is organized as follows. First, in Section II, we give the system models and assumptions. In Section III, we study the impact of reactive jamming attacks on the network performance and the reception similarity. Then, we show the details of the alibi framework in Section IV including the notations and definitions, algorithms, the framework and basic building blocks such as BBC-based timing channel and similarity hashing. We show the evaluation results of the alibi framework in Section V. We discuss the related work in Section VI. Finally, we conclude the paper in Section VII.

## II. System Model

**Network model**: We consider a single-channel WLAN of $n$ nodes. One node is the trusted base station. Denote $\mathcal{N}$ as the set of the nodes in the network (i.e,, $|\mathcal{N}| = n$). Each node in the network is equipped with a half-duplex radio, i.e. it cannot transmit and receive at the same time. Thus, there will be non-negligible delay to switch from transmit mode to receive mode and vice versa. We assume CRC-failed packets are still delivered to the upper layer. We assume each node uses CSMA/CA MAC. We also assume a central detection model, i.e. nodes will send information to the base station.

**Attack model**: We assume some nodes in the network are the reactive jamming attackers. Thus, the attackers also have same physical capabilities as other nodes. The attackers, however, have the complete control of the MAC, physical parameters of the radio network interface. The attackers are insider attackers. That means, they are assumed to know any security-related information of the node such as security keys. They also know any protocols used in the system. The goal of the attackers is to remain undetected while maximizing the number of jammed packets. The attackers use probabilistic reactive jamming strategy. That means whenever an attacker $J$ senses an on-going packet by detecting the presence of a preamble, it will transmit a jamming packet with probability $p_J$. $p_J$ is called the "reactive jamming probability" and is defined for *each sending packet*.

## III. Impact of reactive jamming attacks

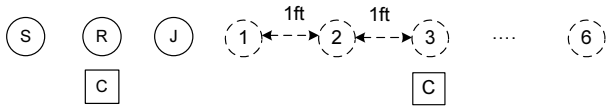
As shown in Figure 1, *we assume that received corrupted packets, caused by the same jamming event, have identical contents* (e.g., nodes 2 and 6 in Figure 1(a)). In this section, we verify this assumption by carrying out several experiments on a testbed of MICAz motes with CC2420 radio. Specifically, we seek answers for the following two questions: 1) *what is the impact of reactive jamming attacks on the network?* and 2) *what is the similarity model of corrupted packet contents under reactive jamming attacks?*.

**Impact of reactive jamming attacks on network performance**: In our experiments, a reactive jamming attack is performed on a set of 3 nodes as shown in Figure 2(a). Nodes are placed such that they can hear each other at the strongest power level (i.e., level 31, 0dBm). $S$ and $J$ are the sender and jammer, respectively. $R$ is the node receiving packets from $S, J$. $C$ acts as the experiment controller. To produce a reactive jamming attack, $C$ will broadcast a message. Upon receiving the broadcast message from $C$, $S$ starts sending a message with a random payload. $J$ also starts sending a message but with a delay $\delta > 0$ between $150\mu s$ and $200\mu s$.

In our experiments, we vary the major factors affecting the SINR: the distance between $S \rightarrow R, J \rightarrow R$ and the sending powers of $S, J$. We put 6 MICAz motes in the line as shown in Figure 2(b). In a reactive jamming attack scenario, we have one sender at mote $i$ ($i = 1..6$) with the sending power level $k$ ($k = 1..30$), one jammer at mote $j$ ($j \neq i, j = 1..6$) with the sending power level $l$ ($l = 1..30$) and 4 receivers at remaining nodes. We try all possible combinations of $(i, j)$ in 6 positions where $j > i$. For each $(i, j)$ pair, we measure the received signal strength indication (RSSI) from the sender (jammer) to each receiver $R$, denoted as $RSSI_{SR}(d_{SR}, P_S)$ ($RSSI_{JR}(d_{JR}, P_J)$), without any sending of the jammer

(sender). We also collect corrupted packets at each receiver for content similarity calculation.

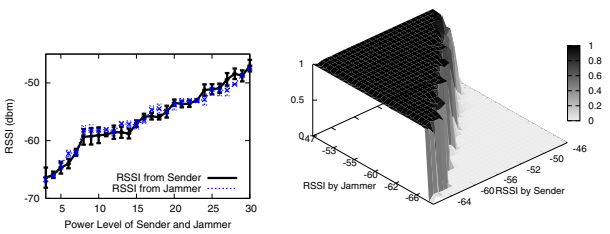
Figure 3(a) shows the RSSI at a receiver 1ft away from the sender and the receiver at different sending power levels, i.e., $RSSI_{SR}(1ft, P_S)$ and $RSSI_{JR}(1ft, P_J)$. The x-axis is the sending power level of the sender and the jammer $(P_S, P_J)$. The y-axis is the RSSI in dBm. Figure 3(b) shows the packet error rate under reactive jamming attacks. The x-axis, denoted by "RSSI by sender" (i.e., $RSSI_{SR}$), is the RSSI of the signal from the sender to the receiver. Note that the RSSI metric takes into account of both the sending power and the distance between the sender and the receiver. Similarly, the y-axis, denoted by "RSSI by jammer" (i.e., $RSSI_{JR}$), is the RSSI of the signal from the jammer to the receiver. A pair of RSSI from the sender and the jammer characterizes a receiver. The z-axis shows the packet error rate of each receiver.




(a) Reactive jamming attack configuration (b) Topology of 6 nodes

Fig. 2. Reactive jamming experiment settings

From the results, we have the following three observations. First, if $RSSI_{SR}(d_{SR}, P_S) >> RSSI_{JR}(d_{JR}, P_J)$, the packet error rate decreases sharply to 0. This region is referred to as the *"white" region*. Receivers in this region are called *white receivers*. Second, if $RSSI_{SR}(d_{SR}, P_S) << RSSI_{JR}(d_{JR}, P_J)$, the packet error rate increases sharply to 1. This region is referred to as the *"black" region*. Receivers in this region are called *black receivers*. Third and last, when $RSSI_{SR}(d_{SR}, P_S)$ and $RSSI_{JR}(d_{JR}, P_J)$ are close ($< 5dBm$ difference), the packet error rate is between 0 and 1. This region is referred to as the *"grey" region*. Receivers in this region are called *grey receivers*.




(a) Power level vs. RSSI (b) Packet Error Rates (z-axis)

Fig. 3. Results on reactive jamming experiments

**Reception similarity under reactive jamming attacks**: We want to see the similarity of received packet content of any pair of receivers under reactive jamming attacks. We treat a packet content as a binary string. The similarity of two packet contents is defined as the similarity of the two corresponding binary strings. The similarity of two binary strings $B_1, B_2$ of length $l$ is defined as $sim(B_1, B_2) = 1 - \frac{\mathcal{H}(B_1,B_2)}{l}$, where $\mathcal{H}(B_1, B_2)$ is the Hamming distance of $B_1$ and $B_2$ defined as the number of positions at which the corresponding bits of $B_1$ and $B_2$ are different. The similarity has a range of $[0, 1]$. We calculate the similarity of all received packet content for each pair of receivers. Thus, we have a table whose lines are tuples of $(RSSI_{SR_1}, RSSI_{JR_1}, RSSI_{SR_2}, RSSI_{JR_2}, sim)$. The pairs $(RSSI_{SR_1}, RSSI_{JR_1})$ and $(RSSI_{SR_2}, RSSI_{JR_2})$ characterize the first and the second receiver, respectively. $sim$ is the average of similarity of received packet content of $R_1$ and $R_2$. Note that in our experiments, the lowest value of the similarity is $0.5$, statistically. This is because the content of the sending packet is uniformly generated, i.e. each bit is uniformly generated between 0 and 1. After constructing the reception similarity table, we have the following observations.

- For a white receiver (i.e., successful reception), it will have very strong reception similarity (close to 1) with other white receivers. This is obvious because nodes in the white region have successful packet reception. It has a wide range of weak reception similarity with grey receivers (range of $[0.55, 0.95]$). This can be explained as the grey receivers still have some correct bits from the sender's content. However, a white receiver has very weak reception similarity with black receivers (range of less than $0.55$). Figure 4(a) shows the reception similarity of a typical white receiver $R$ ($RSSI_{SR} = -50dBm$, $RSSI_{JR} = -64dBm$) with all other receivers.
- Similarly, for a black receiver (i.e., unsuccessful reception), it has a very strong reception similarity with other black receivers, a wide range of weak reception similarity with grey receivers and very weak reception similarity with white receivers. Figure 4(b) shows the reception similarity of a typical black receiver $R$ ($RSSI_{SR} = -59dBm$, $RSSI_{JR} = -54dBm$) with all other receivers.
- For a grey receiver, it has a strong reception similarity (range of $[0.6, 0.8]$) with other grey receivers in the grey region. It has wide range of a weak reception similarity with black and white receivers. Figure 4(c) shows the reception similarity of a grey receiver $R$ ($RSSI_{SR} = -59dBm$, $RSSI_{JR} = -61dBm$) with all other receivers.

From this experimental study, we can conclude that 1) the corrupted packets by different receivers *are not identical* and 2) the similarity of received packet content under reactive jamming attacks has *the locality property* (i.e., receivers closer in the RSSI plane have stronger similarity of packet contents). Thus, in Section IV, we use a threshold $\alpha$ to define a reception similarity between nodes, rather than assuming identical reception among nodes like in the previous work [8] [9].

Note that we also tried the experiments on the butter-fly topology and obtained similar results. This is intuitive because what matters is the *received signal strength from the sender and the jammer to each receiver*.

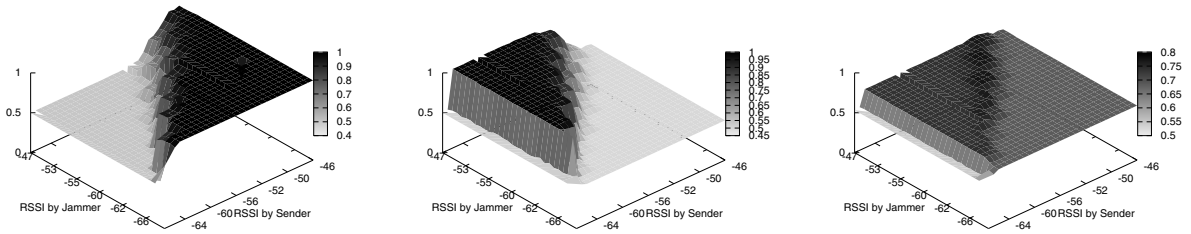
## IV. Alibi Framework

### A. Definitions and notations

Denote $\mathcal{S}(t)$ as the set of transmitters at time slot $t$. Denote $P_{\mathcal{S}\to r}(t)$ as the packet content received by the receiver $r$ under the concurrent sending of senders in $\mathcal{S}(t)$. Denote $PR_{\mathcal{S}\to r}(t)$ ($\mathcal{S} \subset \mathcal{N}\backslash r$) the *proof of reception* for a receiver $r$ at time slot $t$.

**Definition 1** (Alibi of reception (AR))**.** *An alibi of reception for two receivers $r$ and $q$ at time slot $t$ under the set of sender $\mathcal{S}$ is defined as $AR_{\mathcal{S}\to r,q}(t) = sim(PR_{\mathcal{S}\to r}(t), PR_{\mathcal{S}\to q}(t))$, where $sim$ is the similarity function defined in Section III.*

**Definition 2** ($\alpha$-alibi neighbors)**.** *Two nodes $r$ and $q$ are called $\alpha$-alibi neighbors ($0 \leq \alpha \leq 1$) under a set of senders $\mathcal{S}$*




(a) Reception similarity of a white receiver $R$ ($RSSI_{SR} = -52dBm$, $RSSI_{JR} = -64dBm$)

(b) Reception similarity of a black receiver $R$ ($RSSI_{SR} = -59dBm$, $RSSI_{JR} = -54dBm$)

(c) Reception similarity of a grey receiver $R_1$ ($RSSI_{SR_1} = -59dBm$,$RSSI_{JR_1} = -61dBm$)

Fig. 4. Experiment results on the reception similarity (z-axis: darker colors represents higher values)

*in the time slot set* $\mathcal{T} = t_1, t_2, .., t_{|\mathcal{T}|}$ *if* $E[AR_{\mathcal{S}\to r,q}(\mathcal{T})] = \frac{1}{|\mathcal{T}|}\sum_{\forall t\in\mathcal{T}} AR_{\mathcal{S}\to r,q}(t) \geq \alpha$.

**Definition 3** ($\beta$-jammer)**.** *A jammer is called $\beta$-jammer ($\beta \geq 0.5$) if it can have get $\beta$ reception with the packet from a sender $\in \mathcal{N}$ it jams.*

**Definition 4** (Alibi-safe topology with trusted senders)**.** *A wireless network topology is called a $(\alpha, \kappa, \Lambda)$-alibi-safe topology under a set of trusted senders $\Lambda \subset \mathcal{N}$ if for all pairs of sender $s \in \Lambda$ and jammer $j \in \mathcal{N}\backslash\Lambda$, every receiver $r \in \mathcal{N}\backslash\{j, \Lambda\}$ has at least $\kappa$ $\alpha$-alibi neighbors.*

### B. Alibi identification algorithms

*1) Identifying non-colluding attackers:* For a $\beta$-alibi-jammer, whenever it jams a packet, it only has $\beta$ similarity while the other nodes get at least $\alpha$ similarity (statistically). Let us first define an alibi score function $ascore$ for a node $r$ in a time slot set $\mathcal{T}$ as $ascore_r^\alpha(\mathcal{T}) = \sum_{t\in\mathcal{T}} \delta_r^\alpha(t)$, where $\delta_r^\alpha(t) = \begin{cases} 1 & \text{if } \max_{q\in\mathcal{N}\backslash r} AR_{S\to r,q}(t) \geq \alpha \\ 0 & \text{otherwise} \end{cases}$

$\delta_r^\alpha(t)$ indicates whether at time slot $t$, a node $r$ has a reception similarity greater than $\alpha$ with some other nodes. Thus, the necessary condition to identify a jammer $j$ is $ascore_r^\alpha(\mathcal{T}) > ascore_j^\alpha(\mathcal{T})$, $\forall r \in \mathcal{N}\backslash j$. In other words, a node is accused if it has lowest alibi score.

**Lemma 1** (Necessary condition for identifying non-colluding attackers)**.** *In $(\alpha, \kappa)$-alibi-safe topology ($\kappa \geq 1$) with $k_a$ $\beta$-jammers $\mathcal{J} = j_1, ..j_{k_a}$ with jamming rate $p_{j_1}, ...p_{j_{k_a}}$, the alibi scheme can identify any attacker $j \in \mathcal{J}$ if $p_s^{max} \leq 2 - \frac{1-p_j}{p^{agg}(\mathcal{J})}$, where $p^{agg}(\mathcal{J}) = 1 - \prod_{i=1}^{k_a}(1-p_{j_i})$ is the aggregated jamming rate of the jammer set $\mathcal{J}$ and $p_s^{max}$ is the maximum of sending probabilities of the honest nodes.*

*Proof*: See our technical report [10].

*2) Identifying colluding attackers:*

**Definition 5** (Colluding attackers)**.** *Colluding attackers are those who have pre-shared knowledge among themselves*[2].

*Identification algorithm:* The basic strategy to deal with colluding attackers is to exploit their only weakness: they cannot manipulate the reception similarity with other honest nodes under the jammed packets from the trusted senders $S \in \Lambda$. Thus, we define the alibi indicator function as

$$\tilde{\delta}_r^\alpha(t) = \begin{cases} 1 & \text{if } q_\kappa \in \Upsilon_r(\mathcal{N}), AR_{S\to r,q_\kappa}(t) \geq \alpha \\ 0 & \text{otherwise} \end{cases}$$

, where $\Upsilon_r(\mathcal{N})$ is the list of nodes in $\mathcal{N}$ decreasingly sorted by the reception similarity with $r$ at time $t$ and $Q_k$ is the $k$-th element in $\Upsilon_r(\mathcal{N})$. Once again, the alibi score is defined as $\tilde{ascore}_r^\alpha(\mathcal{T}) = \sum_{t\in\mathcal{T}} \tilde{\delta}_r^\alpha(t)$. Similar to the case of non-colluding attackers, an attacker $j$ is identified if $\tilde{ascore}_r^\alpha(\mathcal{T}) > \tilde{ascore}_j^\alpha(\mathcal{T})$, $\forall r \in \mathcal{N}\backslash j$.

**Lemma 2** (Necessary condition for identifying colluding attackers)**.** *In a $(\alpha, \kappa, \Lambda)$-alibi network topology ($\kappa > 1$) with $k_a$ $\beta$-alibi colluding jammers $\mathcal{J} = j_1, .., j_{k_a}$ with jamming rate $p_{j_1}, .., p_{j_{k_a}}$ and the total jamming target $p_T = \sum_{i\in\mathcal{J}} p_i$, the alibi scheme can identify at least one attacker if $p_s^{max} < \frac{p_T}{k_a}$.*

*Proof*: See our technical report [10].

### C. Alibi protocol

The identification algorithm in Section IV-B requires the proofs to be present at the detector. That means, every node in the network has to participate in the detection algorithm. Nodes that do not report proofs are at risk of being accused as attackers and received appropriate reaction from the system such as their removals.

*Jamming detection*: Because sending proofs incurs overhead to nodes, the base station only collects proofs when there is a jamming attack. To detect the presence of jamming attacks, we use a similar detection techniques proposed in [11]. For the uplink traffic (i.e., from nodes to the base station), a jamming attack is declared if the base station receives a significant number of corrupted packets with strong received signal strength. For the unicast downlink traffic, the base station declares the presence of jamming attacks after getting a significant number of sending packets without receiving acknowledgements.

- *Step 1:* Once the base station detects a jamming attack, it starts a *proof-collection* period of $T_{collect}$ in which it starts to broadcast "test" packets at random intervals. The content of the test packets are uniformly drawn from the message space, i.e. each bit in the content is uniformly drawn from $\{0, 1\}$. The uniform randomness ensures that the attacker can only have a blind guess on the content. Note that each honest node always maintains proofs in the latest window interval of $T_{collect}$. At the end of the proof-collection period, the base station broadcasts a *proof-exchange* message.
- *Step 2:* When nodes receive the *proof-exchange* message from the base station, they immediately start sending the

[2]Note that the definition above allows the attackers to collude through a pre-shared knowledge only. *We do not consider the case where attackers can share new knowledge during the network operations* (e.g., their proofs of reception).

proofs that they have just collected in the last $T_{collect}$ time interval to the base station.

- *Step 3:* After receiving proofs from nodes collected in the last $T_{collect}$ time interval, the base station starts the identification algorithm based on the set of time slots in which it sent "test" packets. It removes any identified attackers. After that if there is still jamming attack going on, it repeats Step 1.

All messages between nodes and the base station are transmitted using BBC-based timing channel to be much more jamming-resistant to reactive jamming attacks. BBC is a keyless jamming-resistant broadcast communication proposed in [4]. The basic idea is to have the sender create *"indelible" marks* in an additive-OR channel to convey a sending message. Thus, a jammer with limited power cannot erase the marks: it can only add extra marks. The alibi framework exploits this property to create a BBC-based timing channel which is robust against reactive jammers. More details can be found in our technical report [10].

To ensure that the proofs can be exchanged on the timing channel, the alibi framework compresses the proofs using a hashing technique called "similarity preserving hashing". Due to the space limit, we omit the dicussion of this technique here. More details can be found in our technical report [10].

## V. Evaluation

### A. Evaluation of alibi-safe topologies

There is a strong connection between alibi-safe topologies and physical topologies. Thus, it is important to determine whether a given physical topology is an alibi-safe topology. We perform the alibi-safe test for three types of physical topology in a $20m \times 20m$ square: star, grid and random topology where the set $\Lambda$ only has the trusted base station located at the center of the square. Figure 5 shows the results of the alibi-safe tests for the network size from 10 to 100. The x-axis is the $\alpha$ value and the y-axis is the *minimum* number of $\alpha$-neighbors for every node. That means, each point represents a possible $(\alpha, \kappa)$-alibi-safe topology. For star topology where nodes surround the base station, there is high chance for a node to have alibi neighbors as shown in Figure 5(a). For grid topology, it is less alibi-safe as shown in Figure 5(b) because nodes have different distance to the base station. The alibi-safe of random topologies really depends on network density as shown in Figure 5(c). As we can see, in terms of robustness to colluding attackers, star topologies are the strongest. In a star topology, each node has roughly $30\%$ of other nodes as its $0.55$-neighbors. For grid topologies and random topologies, these numbers are around $15\%$ and $7\%$, respectively.

### B. Evaluation of alibi on TOSSIM simulator

We use TOSSIM [12] as our network simulator for a large-scale evaluation of the alibi framework. We incorporate the noise trace and the reception similarity obtained from the experiments in Section III into TOSSIM. BBC-based timing channel is implemented at the MAC layer in TOSSIM and is transparent to the application layer. There are two alibi protocols: *the Omniscient protocol (OMS)* and *the Alibi-BBC-Simhash protocol (ABS)*. The Omniscient protocol is the optimal protocol because it assumes every proof is available immediately at the central detector right after its creation. The ABS protocol uses the BBC-based timing channel and simhash. We evaluate the two alibi protocols under three types of topologies: star, grid and random topology in a square of $20m \times 20m$ and two types of attackers: non-colluding and colluding attackers. We vary the simulation parameters as shown in Table I. We obtain the detection accuracy (i.e., the detection probability and the false alarm rate), the detection time, the network performance and the network overhead. However, due to the space limit, we only show the results for the star topology.

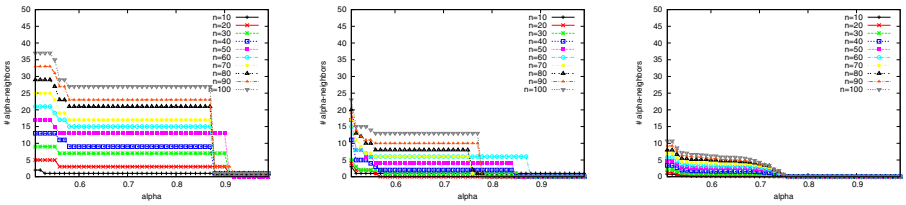
| Parameter | Values |
|---|---|
| Number of nodes | $n = [10 - 40]$ |
| Number of attackers | $[1 - 9]$ |
| Jamming rate | $[0.1 - 1.0]$ |
| Value for $\alpha$-neighbors | $\alpha = 0.55$ |
| $\gamma$ | 0.05 |
| Simulation time | 1000 seconds |
| BBC maximum number of concurrent transmissions | 50 |
| BBC message length in time | 4s |
| Number of BBC messages per proof-exchange period | 10 |
| Number of bits for simhash | 16 bits |

TABLE I
SIMULATION PARAMETERS

*Detection accuracy:* Figure 6 shows the detection accuracy of the ABS protocol under non-colluding jamming attacks and colluding jamming attacks. Specifically, Figures 6(a) shows the detection accuracy for non-colluding attackers using the same jamming rate of $0.2$. It is shown that as the number of attackers increases, the detection probability decreases. This is because when the network has more attackers, the chance for an attacker to get alibis by the jamming actions of the other attackers also increase. It is shown that as the network size increases from 10 to 40, the detection probability increases. This is because the larger network size increases the number of $\alpha$-neighbors and makes the attackers busier to jam. We also obtain the results for different $\gamma$ but do not plot them here. Essentially, larger value of $\gamma$ will lead to lower false alarms but also lower detection probability. Figures 6(b) shows the detection accuracy for colluding attackers who collude and coordinate to target at $100\%$ jamming rate. The trends are similar to the case of the non-colluding case. However, the detection probability is slightly worse compared to the non-colluding cases. This is due to the fact that 1) colluding jammers need to jam less than non-colluding jammers for the same aggregated jamming target and 2) each node needs more $\alpha$-neighbors than the non-colluding case. Figure 6(c) and 6(d) show the detection performance of ABS protocol against the OMS protocol for $n = 40$. ABS protocol has a gap of around $0.2 - 0.3$ to the OMS protocol. This is because the nodes cannot send all the proofs to the central detector due to the low throughput of the timing channel. We also have the results for packet error rate and detection time. Please refer to our technical report [10] for more details.

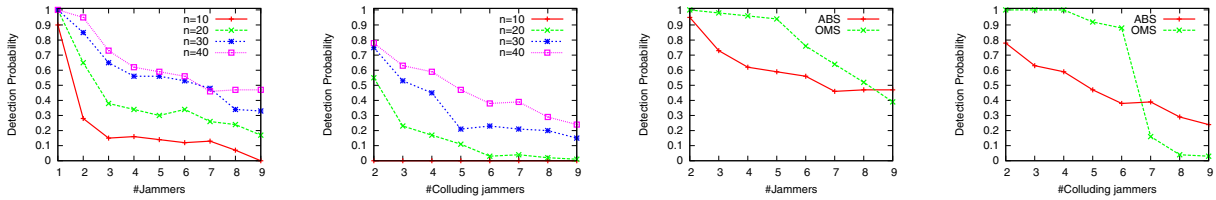
## VI. Related work

Spread spectrum techniques have been the most effective jamming defense mechanisms (e.g., DSSS, FHSS and CSS) [13]. However, spread spectrum does not work if the jammer knows the hopping-pattern of the FHSS and the pseudo-noise chip sequence of DSSS [14]. Therefore, there is a strong need to to use multiple shared secrets rather just one single shared secret. There have been efforts using a tree of shared secrets [15] or multiple shared-secrets [16] [17]. Another




(a) Star topology (b) Grid topology (c) Random topology

Fig. 5. Physical topologies vs. Alibi-safe topologies




(a) ABS (non-colluding. n=10-40) (b) ABS (colluding, n=10-40) (c) ABS vs. OMS (non-colluding, n = 40) (d) ABS vs. OMS (colluding, n = 40)

Fig. 6. Detection probability

approach is to remove the dependencies on the pre-shared secret of traditional spread spectrum technologies, referred to as zero shared-secret spread spectrum such as Uncoordinated Frequency Hopping (UFH) [2] [1] [5], Uncoordinated Direct Sequence Spread Spectrum (UDSSS) [3], RD-DSSS [18] and BBC [4]. The main drawback of this approach is the low communication throughput (compared to other classic spread spectrum techniques). Therefore, such zero shared-secret spread spectrum should only be used for delivering new shared secret to the network. Even though spread spectrum techniques have raised the bar for jamming defense, they are not sufficient to deal with the situation where attackers are compromised nodes in the network. In such a case, any attempts to deliver new shared secrets are useless because the attackers are still inside the network. Thus, it is necessary to first identify compromised nodes that launch jamming attacks to the network and then deliver the new shared secrets to un-compromised nodes only. Researchers have been looking into the problem of identifying mis-behaving/compromised nodes. In [6] [7], the authors propose the detection schemes to identify mis-behaving nodes that greedily consume the bandwidth by modifying its MAC parameters. However, these detection schemes will fail to detect stealthy reactive jamming attacks considered in this paper because they rely on the identity-related clues to infer the mis-behaving nodes. Alibi framework is a complement to the above approaches. It needs a jamming-resistant communication like BBC [4]. In terms of detection, it collects the proofs showing good behaviors of nodes instead of collecting proofs of bad behaviors of nodes [6] [7].

## VII. Conclusions

We have presented a design and implementation of the alibi framework to deal with reactive jamming nodes. The framework relies on the novel concept of "alibi" in which the detector collects good proofs of nodes to infer the compromised nodes. We have shown a necessary condition, the alibi-safe topology condition, to check whether a given network topology is safe under reactive jamming nodes if the alibi framework is used. We have also evaluated the framework via analysis, simulation and test-bed experiments. The results show that the alibi framework can deal with both non-colluding and colluding attackers in reasonably dense networks.

## References


[1] M. Strasser, C. Pöpper, S. Čapkun, and M. Čagalj, "Jamming-resistant key establishment using uncoordinated frequency hopping," in *IEEE Symposium on Security and Privacy (Oakland)*, 2008.

[2] M. Strasser, C. Pöpper, and S. Čapkun, "Efficient uncoordinated fhss anti-jamming communication," in *ACM MobiHoc*, 2009.

[3] C. Pöpper, M. Strasser, and S. Čapkun, "Jamming-resistant broadcast communication without shared key," in *USENIX security Symposium*, 2009.

[4] L. C. B. III, W. L. Bahn, , and M. D. Collins, "Jam-resistant communication without shared secrets through the use of concurrent codes," U.S. Air Force Academy, Tech. Rep., 2007.

[5] T. Jin, G. Noubir, and B. Thapa, "Zero pre-shared secret key establishment in the presence of jammers," in *ACM Mobihoc*, 2009.

[6] M. Raya, J.-P. Hubaux, and I. Aad, "Domino: A system to detect greedy behavior in IEEE 802.11 hotspots," in *ACM MobiSys*.

[7] P. Kyasanur and N. H. Vaidya, "Detection and handling of MAC layer misbehavior in wireless networks," in *IEEE DSN*, 2003.

[8] H. Nguyen, T. Pongthawornkamol, and K. Nahrstedt, "Identifying insider-based jammers in single-hop wireless networks," in *IEEE MILCOM*, 2009.

[9] ——, "Identifying insider-based jammers in multi-channel wireless networks," in *IEEE Globecom*, 2010.

[10] ——, "Alibi framework for identifying reactive jamming nodes in wireless lan," University of Illinois at Urbana-Champaign, Tech. Rep., August 2010. [Online]. Available: http://www.ideals.illinois.edu/handle/2142/16661

[11] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *ACM MobiHoc*, 2005.

[12] P. Levis, N. Lee, M. Welsh, and D. Culler, "TOSSIM: accurate and scalable simulation of entire tinyos applications," in *ACM Sensys*, 2003, pp. 126–137.

[13] R. Negi and A. Perrig, "Jamming analysis of mac protocols," Carnegie Mellon Technical Memo, Tech. Rep., 2003.

[14] D. J. Thuente and M. Acharya, "Intelligent jamming in wireless networks with applications to 802.11b and other networks," in *IEEE MILCOM*, 2006.

[15] J. Chiang and Y.-C. Hu, "Dynamic jamming mitigation for wireless broadcast networks," in *IEEE INFOCOM*, 2008.

[16] A. Chan, X. Liu, G. Noubir, and B. Thapa, "Control channel jamming: Resilience and identification of traitors," in *IEEE ISIT*, 2007.

[17] L. Lazos, S. Liu, , and M. Krunz, "Mitigating control-channel jamming attacks in multi-channel ad hoc networks," in *ACM WiSec*, 2009.

[18] Y. Liu, P. Ning, H. Dai, and A. Liu, "Randomized differential DSSS: Jamming-resistant wireless broadcast communication," in *IEEE INFOCOM*, 2010.