Alibi: A novel approach for detecting insider-based jamming attacks in wireless networks

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}@uiuc.edu

***Abstract*— We consider the problem of detecting the insider-based attacks in the form of jammers in a single-hop wireless LAN environment, where jammers have the inside knowledge of frequency hopping patterns and any protocols used in the wireless network. We propose a novel jammer model in which the jammers are modeled by the number of channels that they can jam simultaneously. We further propose the novel concept of an atomic jammer which is the basic component necessary to deal with stronger jammers. To deal with atomic jammers, we propose a class of novel protocols called alibi protocols. The basic idea of the alibi protocols is to exploit one major limitation of the atomic jammers: they cannot jam two channels at the same time. Therefore, honest nodes in the network can occasionally switch to another channel, called the alibi channel, to transmit proofs for their honesty witnessed by some other honest nodes. We specify a necessary condition and desired properties such as detection time, false alarms and miss detections of this class of protocols. We prove that with high probability the detection time of these protocols is $O(n\ln(n))$. We also propose some more practical alibi-based protocols such as 1-propagation and 1-gossiping and prove their desired properties. We further extend our work to the lossy channel model. The simulation results in ns2 confirm our analysis. The overall results of these protocols show a promising research direction to deal with insider-based jamming attacks.** [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 (SNIR) and reducing probability of successful message receptions.

The jamming attack is serious in several ways. First, jamming attack is a type of Denial-of-Service attacks (DoS). Jammed communication channels are useless most of the time. Second, it is relatively easy to perform a jamming attack. The attacker only needs a transmitter (i.e. jamming device) powerful enough to transmit a signal to disrupt the targeted wireless communication because the wireless medium is open and shared in nature. For example, an inexpensive device able to transmit signal on 2.4Ghz is enough to jam a 802.11b network [1], [2]. Third, it is hard to detect the jamming attack (i.e. the existence of the attack) and identify/locate the attacker. The main reason is due to the ambiguity between unintentional interference and intentional jamming attacks [3]. Lastly, even if the jamming attack and the attacker are detected, it is very challenging to automatically recover from the jamming attacks [4][5][6]. The network needs an out-of-band means to defense the attacks (e.g. having a person remove the jamming device or having the network do a spatial retreat [4]).

There has been plethora body of research work on jamming attacks and defenses. Jamming attacks can be classified as proactive or reactive. In the proactive jamming strategy, the attacker jams the channel without caring about the ongoing communication. A typical example of this type is the continuous jamming [7][2]. This strategy is the simplest way to perform a jamming attack. However, it is not energy-efficient due to the continuous jamming activity. This also makes the attacker easy to detect. Reactive jamming strategy [8][9][10][11][12] [13][14][15][2][16] [17], in contrast, avoids these drawbacks by intelligently listening and jamming the channel. In this strategy, the attacker only keeps listening and jams "important" packets such as control packets [14][15]. Corrupted control packets can drastically reduce the effective throughput of the communication channel [14][15]. Reactive jamming attack is more complicated than a proactive jamming attack. It is harder to detect and does not necessarily use less energy because the transmitter has to listen to the communication channel. The danger of reactive jamming lies in the"effectiveness" measured by the ratio between the effort spent to corrupt packets and the damage caused to the communication channel.

Due to the danger of various jamming attacks, jamming defenses have gained much attention from researchers. One of the most effective jamming mitigation is the spread spectrum techniques. By hopping the carrier frequency (frequency-hopping spread spectrum - FHSS) or spreading its signal in time (direct-sequence spread spectrum - DSSS), the network can force the jammer to expend several-fold more power than if spread spectrum were not used [18][17]. However, spread spectrum does not work if the jammer knows the hopping-pattern (HP) of the FHSS or the pseudo-noise chip (PN) sequence of DSSS. Once the attacker knows such knowledge, he can jam the channel very effective. For example, in 802.11

[1]This material is based upon work supported by the National Science Foundation under Grant CNS-0524695 and Vietnam Education Foundation. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of those agencies.

DSSS the PN is a common knowledge and the attacker can easily obtain it [19]. By just using the COTS 802.11 cards, the attacker can easily modify the firmware to have an effective 802.11 jammer [2]. That said, the "outsider" attack (i.e. no knowledge of the HP or PN) can be defended effectively with spread spectrum technology while "insider" attack is still a problem.

Indeed, dealing with insider-based attacks, where the "shared secret" such as shared HP or PN is compromised, is a challenging problem. This problem exists not only in the spread spectrum technology but also in other wireless technologies such as Ultra-wide band (UWB) (pulse-pattern as the shared secret)[18][20]. Unfortunately, there have been few research results on this topic. These research results share the view of considering shared secret as a type of "shared key" among all nodes. From this point of view, dealing with compromised shared key is similar to the key management in the traditional security literature. Specifically, hierarchical key management and asymmetrical key scheme have been explored in [5] and [20]. In [5], the authors extend the idea from the well-known hierarchical key management to eliminate the compromised shared secret. However, this scheme is designed only for wireless broadcast network where the base station can send/receive on different channels at the same time. In [20], the authors propose a concurrent coding scheme to form a communication primitive under jamming condition. This can be used as a way to setup a shared key from the asymmetric key by using some techniques like Diffie-Hellman [21]. This scheme, however, is only applicable for point-to-point communication.

In this work, we consider the problem of detecting the insider-based jammer in time slotted single-hop wireless networks. Specifically, in our attack model, the jammer knows the shared secret and any protocols used in the system (i.e. no security-by-obscurity). The jammer intelligently uses the reactive jamming strategy to hide himself from getting detected. We start from an important observation that a jammer cannot send on two different channels simultaneously. That means, within a time slot that is small enough, the jammer cannot send/jam on two different channels. This observation leads to the definition of "atomic jammer". By following this definition, stronger jammers, such as the ones that can send on multiple channels simultaneously, can always be broken down into multiple atomic jammers. Therefore, our exact problem is *detecting one insider-based atomic jammer in the single-hop wireless networks*.

We propose a novel approach to cope with this problem. Our basic idea to exploit the limitation of the atomic jammer by introducing an additional channel, called "alibi" channel, beside the main channel. The alibi channel is used for nodes to create alibis - proofs for the honesty. Specifically, an alibi for a node is a proof showing that in the specified time slot the node was seen, by some witnesses, sending a good message while the main channel was jammed, observed by some witnessed. Hence, the node is obviously not the atomic jammer. We design a class of randomized protocols in which only good nodes can create alibis while the atomic jammer will *never* be able to create the proof even though he knows the design of the protocol. In our protocols, the atomic jammer will eventually be identified when each good node has its alibi. We prove that it takes $O(n \ln n)$ slots for "Omniscient scheme" - the one that knows proofs and alibis immediately after created without any message exchange - to detect one atomic jammer. We practically propose two other schemes that need to exchange proofs and alibis and prove that they can still achieve $O(n \ln n)$ time slots for detecting the atomic jammer. We also verify our analysis in ns2 simulation.

In summary, our contributions in this paper are

- The concept of "atomic jammer" as the foundation of designing jamming defense.
- The concept of alibi and the design of alibi-based protocols to detect one atomic jammer.
- The theoretical analysis and simulation-based performance evaluation of alibi-based protocols

The rest of the paper is organized as follows. We start with the system model including network model, jammer model and problem formulation in Section II. We present the general alibi framework including the basic ideas, examples and desired properties for any alibi-based protocols in Section III. In Section IV, we propose four alibi-based protocols and comprehensive analysis for each protocol with respect to the desired properties. In Section V, we give some further extensions such as lossy channels and a more generalized attacker's strategy. We evaluate the proposed protocols in Section VI. In Section VII, we conclude our paper.

II. System Model

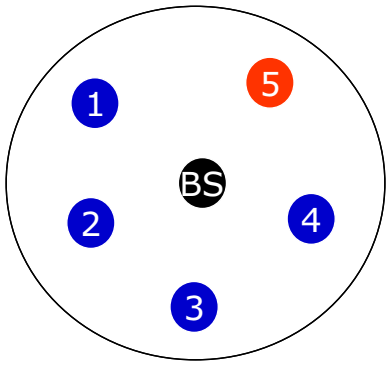
This section presents our system model. Notations used in this paper can be found in Table II.

A. Network Model

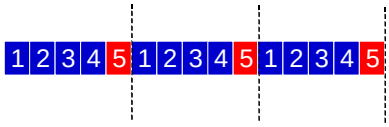
We consider a single-hop wireless network that has n nodes $N_1..N_n$ and a base station (BS) in which nodes can hear each other directly (i.e. in one hop) as shown in Figure 1(a). All nodes talk to the BS via a pre-defined channel, referred to as the *main channel* M. They use a simple Time-Division Multiple Access (TDMA) to access the shared wireless medium. Specifically, there will be n *slots* of size s in a *round*. Each slot is uniquely pre-assigned to a node. Nodes only transmit in their assigned time slots. Figure 1(b) gives an illustration of this simple TDMA scheduling. We assume nodes are time-synchronized (by GPS[2], for example). Thus, this simple TDMA scheduling will cause no collisions. However, it is worth noting that extension to other scheduling such as more complicated TDMA or CSMA/CA is possible but that is orthogonal to the problem addressed in this paper.

Nodes in the network have a set of m orthogonal channels $\Gamma = \{C_1, ...C_m\}$ that they can switch to. These channels may not be necessarily contingent in frequency. For 802.11b, this set is the channel $\{1, 6, 12\}$. We also assume a constant channel switching delay and denote it as τ.

[2]The accuracy of the clock synchronization using GPS is μs



(a) Single-hop wireless network with BS



(b) Simple TDMA scheduling

Fig. 1. Network Model

B. Jammer Model

In order to build an effective defense strategy, one must understand the capabilities of the attacker. For jamming attacks, there are various factors forming the capabilities of a jammer. Power level of the transmitter, the frequencies that the transmitter can transmit on, frequency switching delay and the knowledge about the network (i.e. insider-based or outsider-based) are several important factors for the jammer.

We model a jammer by the knowledge he knows about the network and the number of channels that he can jam simultaneously. The former concerns "'outsider-based" or "insider-based" knowledge. The latter will lead to a novel concept in our jamming model: the "atomic" jammer.

1) Outsider-based & Insider-based jammer: Outsider-based jammer is the one that jams the communication channel without knowing the shared secret such as the hopping pattern. Literally, this attacker can be defended against efficiently by the current state-of-the-art [13][22][19]. Insider-based jammer is the one that knows the shared secret such as the hopping pattern and can jam the communication very efficiently. In the previous example, where the network uses frequency hopping for jamming mitigation and the jammer knows the shared hopping pattern, the jammer can hop to the same channel as other nodes do and jams the communication at any time. Even worse, if the jammer uses the reactive jamming strategy in which he only jams after sensing on-going communication, he may even spend less power and achieve low probability of getting caught. In this work, we only consider the insider-based jammers (see Section II-C).

2) Atomic jammers: We also characterize the jammer by the *number of channels* in Γ that he can jam *simultaneously*. In this way, the *strongest* jammer is the one that can simultaneously jam all possible channels in Γ (or even more). The weakest jammer is the one that can jam only one channel at any time. We refer to the class of weakest jammers to as the "atomic jammers". They are called as "atomic" due to two reasons. Any stronger jammers can be viewed as multiple "atomic jammers" with *a perfect collusion and coordination*. Atomic jammers cannot be decomposed to any weaker jammers.

An important aspect in our definition is the notion of "simultaneous timing". By this, we consider time slots of size σ in which the jammer's capabilities are characterized. Specifically, with this notion, the characterization of a jammer becomes "the number of channels in Γ that the jammer can jam in a time slot of size σ". That means, the capabilities of a jammer are projected onto the plane (Γ, σ), and the exact definition of the atomic jammer N_J is the one that can *jam only one channel in* Γ *within the time slot of size* σ[3].

To formalize this notion, let $C_{N_J}(\Gamma, \sigma)$ denote the number of channels in Γ that the jammer N_J can jam within a time slot σ. N_J is said to be a jammer under (Γ, σ) if and only if $C_{N_J}(\Gamma, \sigma) > 0$, and is an atomic jammer under (Γ, σ) if and only if $C_{N_J}(\Gamma, \sigma) = 1$.

Let us also denote σ^{min} for the smallest σ that N_J is still a jammer under (Γ, σ^{min}) and σ^{max} for the largest σ that N_J is still an atomic jammer under (Γ, σ^{max}). It is followed that for any $\sigma' \in [\sigma^{min}, \sigma^{max}]$, N_J is an atomic jammer under (Γ, σ').

To illustrate how this concept maps to the reality, let us consider the scenario where the jammer uses an Atheros-based 802.11a/b/g wireless card as the transmitter to jam a 802.11a network. The only way for the jammer to jam on two channels is to jam the first channel, switch to the other channel and jam the second one. To do this, it takes at least the channel switch delay of the card for the jammer to do jamming, even if we ignore the time to damage packets on the targeted channels. Because the channel switching delay for an 802.11/a/b/g wireless card is between 1-4ms, the jammer cannot jam on two different channels within the time slot size of 1ms. Thus, if $\sigma = 1$ms, the jammer is the atomic jammer under $(\Gamma, \sigma = 1\text{ms})$.

The novel concept of atomic jammer is important in several ways. First, it abstracts the jamming capabilities of the jammer composed by several factors: the power level of the transmitter, the frequency set that it can transmit on, the channel switching delay and many more. This abstraction helps the jamming defense to avoid considering multiple factors at the same time and thus complicating the problem. Second, this concept even helps to quantify the strength of the jammer according to the number of channels in Γ he can jam simultaneously. In this way, a jamming defense can be specifically designed to defend against certain classes of jammers and thus a quantifiable jamming defense. Lastly, the relationship between atomic and

[3]The relationship between σ and s will be discussed in III-C

strong jammers is particularly helpful for jamming defense. Specifically, any jamming defense scheme that can deal with atomic jammers can always be extended to deal with stronger jammers.

C. Problem Formulation

The problem we consider in this paper is *detecting one insider-based atomic jammer*. The jammer N_J has the knowledge of any protocols used in the system and any shared secrets among the network. Thus, we assume he is one of the nodes in the network. That means $N_J \in \{N_1 ... N_n\}$ and the problem is to find him.

There will be several aspects of this problem to be considered. Certain properties such as detection time, false alarm probability, miss detection probability and overhead will be discussed in Section III-E. We will first consider this problem under the lossless channel condition and extend it to lossy channel in Section V.

III. Alibi's Framework

A. Basic Idea

The basic idea is to exploit the limited capabilities of the atomic jammer N_J: he cannot jam two channels simultaneously. Specifically, if he jams on the main channel in a time slot, he cannot send on another channel in the same time slot. This opens a chance for good nodes to *prove their honesty*. Nodes occasionally switch to and transmit on another channel (when idle) to prove that they were transmitting on another channel while the main channel was jammed in a time slot. In this way, *only good nodes can prove their honesty while the atomic jammer can never do that*. Eventually, all good nodes are proved to be honest and the jammer N_J is identified.

An analogy to this idea can be found in crime investigation where the detective can gather all possible suspects and knows for sure one of them must be the criminal. If any suspect can show a trusted proof showing that he was seen at another place at the time the crime was committed, he can be out of the investigation. Unless the detective can find out a trusted proof of the criminal, he has to keep gathering proofs until there is one suspect left in the pool who cannot get any trusted proof to make the conclusion. We call such trusted proofs as *alibi* and people seeing him as *witnesses*.

In alibi scheme, a new channel $A \in \Gamma \backslash \{M\}$, called as *alibi channel*, is used for good nodes to create proofs and alibis. The time slots of alibi channel also have size of s and are aligned with the main channel. The channel access scheduling and nodes' behavior are different. In any time slot, nodes in the network play only one in four possible roles in the alibi framework: *M-defendant* - the scheduled sender on the main channel M, *A-defendant* - the scheduled sender on the alibi channel A, *M-witness* - the nodes voluntarily deciding to become a witness on the main channel and *A-witness* - the nodes voluntarily deciding to become a witness on the alibi channel. Nodes randomly choose to play one of the role with a pre-defined probability. However, the jammer can play any role he likes. Also, for the shake of the simplicity, we assume each node is uniquely assigned to be the M-defendant in each time slot (i.e. no collisions on the alibi channel).

In any time slot, for a node that is not assigned to be M-defendant or A-defendant, it decides to switch to the main channel M or switch to alibi channel A, with a certain probability, to become a M-witness or A-witness, respectively. For any time slot, M-witnesses overhear the main channel M and record whether the main channel is jammed (by the jammer), is occupied (by the M-defendant) or is empty in this time slot. M-witnesses store these records, which are called *M-proofs*. A M-proof basically keeps the state of the main channel at a specific time slot. A-witnesses will also do a similar thing on the alibi channel A to create *A-proofs*. M-proofs and A-proofs of the same time slot are exchanged and combined. While there are various state combinations of M-proofs and A-proofs, the only useful proof combination is when a M-proof shows a jammed state at time slot t and an A-proof shows an occupied state by an A-defendant N_i at time slot t. Such combination shows that N_i cannot be the jammer and is referred to as *alibi* of N_i at the given time slot. Alibis for nodes are accumulatively created until the jammer is identified.

B. An Example

Figure 2 shows an example of how the basic alibi scheme works for a network of 5 nodes where the jammer is N_5. The figure has three parts: the details of the main channel at the top, the details of the alibi channel in the middle and the details of proofs and alibis at each local node at the bottom. The right most part explains the symbols used in the figure. Details are shown in time. A column going from the top part to the bottom part is the snapshot of every part in that time slot.

The main channel is scheduled as round-robin, i.e. each node is assigned a slot turn-by-turn. The jammer (N_5) also has a time slot. Similarly, on the alibi channel, each node is pre-assigned a slot to send in each round. "X" denotes for a jam action of the jammer (N_5). For example, in the first round, the first four slots are jammed and "X"symbols are placed on top of them. "E" denotes for an "empty" slot where no activity is recorded. An example is the 4th slot of the first round on the alibi channel, the jammer is busy jamming on the main channel and cannot send any other packets on the alibi channel. "M" and "A" denote M-defendant and A-defendant in that time slot.

Let us now go through the first three time slots of the first round. In the first time slot, node N_1 is the M-defendant and node N_2 is A-defendant. There is no M-witnesses and two A-witnesses: N_3 and N_4. In this time slot, the jammer (N_5) jams the main channel (the "X" symbol). By the end of this time slot, N_1, N_2 marks themselves as a M-defendant and A-defendant, respectively. N_3, N_4 hear N_2 on the alibi channel so they create a proof showing that N_2 is the A-defendant in this time slot. If we assume all proofs are gathered to a central entity (e.g. an "oracle" entity), nothing cannot be concluded

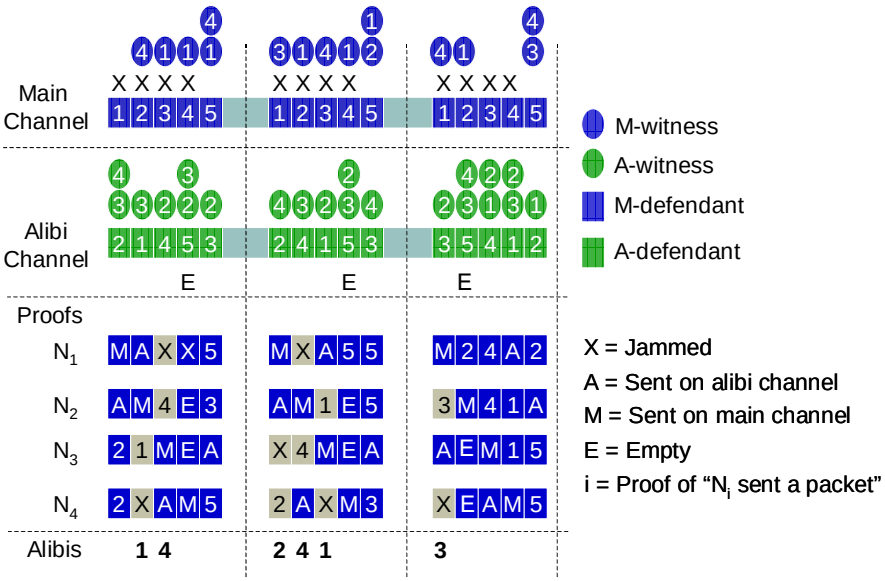


Fig. 2. An illustration of Alibi scheme

from this first time slot. The main channel was jammed but no one was the M-witness so no alibi can be made.

Let us now go to the second time slot. In this time slot, N_2 is M-defendant. N_1 is A-defendant. N_4 is M-witness. N_3 is A-witness. The main channel is jammed in this time slot. Thus, N_4 records a "X". N_3 hears N_1 on the alibi channel and creates a proof showing that he saw N_1 was sending on alibi channel at time slot 2. If proofs, created and held by N_3 and N_4 are combined, then one can conclude that N_1 cannot be the jammer. Thus, N_1 has an alibi.

Similarly for the third time slot, if proofs created by N_1 and N_2 are combined, one can conclude that N_4 cannot be the jammer and thus can create an alibi for N_4. For the rest of the time slots, nodes in the network will follow this protocol and keep creating alibis. Eventually, by the end of time slot 11, each honest node has an alibi and one can conclude that N_5 is the jammer.

Apparently, this example only illustrates how the basic protocol works. There are several issues that have to be considered. For example, how should proofs and alibis be exchanged and combined? How fast can the jammer be detected? Is there any false alarm or miss detection? These issues will be discussed in details in Section IV. In the next subsequent sections, we present a necessary condition for the alibi to work, a more detailed descriptions on roles, proofs and alibis and a set of desired properties to evaluate any alibi-based protocols proposed later.

C. Necessary condition

The necessary condition for the alibi scheme to work is that the slot size s of slotted scheduling has to be equal or less than the slot size σ of the atomic jammer N_J:

$$s \leq \sigma.$$

This obviously imposes a constraint - a required strength for the defense - on the network. In the previous example of 802.11, the slot size specified in the standard [19] is in the order of microseconds which is smaller than the $\sigma = 1$ms of the atomic jammer. Thus, this satisfies the necessary condition for any alibi-based protocol to possibly defend against the jamming with $\sigma = 1$ms.

Another condition is on the channel switching delay τ of the nodes. If $\tau \leq s$, then nodes can always decide and switch at the beginning of each slot. However, if $s < \tau \leq ks$, where $k = 2..n-1$, then nodes have to schedule in advance if they want to switch another channel. Specifically, if it wants to switch at t, it has to schedule to switch at time $t-\tau$. Obviously, this will limit the possible witnesses at each time slot and thus affect the performance of the detection algorithm. *However, in this paper we only consider the case where $\tau \leq s$ for the simplicity of the algorithms and proofs*. The effect of larger value of τ will be considered in the future work.

D. Roles, Proofs & Alibis

1) Roles: As just discussed, a node can play one of four roles *M-defendant*, *A-defendant*, *M-witness* and *A-witness* in a time slot. M-defendant role is basically the sender on the main channel which is already assigned by the TDMA scheduling. A-defendant role can be assigned similar to the simple TDMA scheduling on the main channel to avoid unnecessary collisions or can be done in a distributed manner by each node. We will show later that the former is vulnerable to slander attacks even though it can yield much faster detection time (see Section IV-A). In the latter assignment scheme, in each time slot a node decides to be an A-defendant with uniform probability p_{AD}. In this way, while there may be collisions on the alibi channel, it still works pretty well as shown in Section IV. Furthermore, it has an advantage of confusion because the attacker cannot predict who will be the A-defendant at any time slot. This is

crucial to avoid the slander attack successfully on the central assignment.

In time slots that nodes do not play any defendant roles, they will play witness roles. Specifically, each node becomes an A-witness with a probability p_{AW} and becomes an M-witness with a probability $(1 - p_{AW})$.

Defendants broadcast messages overheard by witnesses. The broadcast message sent by defendants has the *id* field (and other fields specified later in Section IV). The id is a unique identifier for a node such as the MAC address. For M-defendants, the id is already included in the messages sent to the BS on main channel M. For A-defendants, they have to explicitly send messages including id.

2) Proofs: Witnesses receive messages from defendants, extract id field if possible, include the timestamps and store them locally as *proofs*. Specifically, proofs have the following format

$$(state : 16bit, \#timeslot : 16bit)$$

where $state$ is the state of the channel and $\#timeslot$ specifies timestamp when the state was recorded. Possible values of $state$ is listed in Table I.

3) Alibis: Proofs are exchanged and combined to form alibis. Possible combinations of proofs are shown in Figure 3. While there are 16 possible combinations, only two of them can lead to creation of alibis. Those two cases - the two cells with shadow background - are when one channel (either main channel M or alibi channel A) is witnessed to be jammed and another channel has a defendant observed by at least one witness.

There are several interesting aspects of other combinations. The "M" column always leads to "not trusted" combination because the jammer can both jam and declare himself as a defendant in the same time slot. That is why the defendants have to be overheard by witnesses and only proofs about defendants created by witnesses are trusted. A similar conclusion applies to "A" row.

The "E" column can indicate a suspect behavior of the node assigned to send proofs for its M-defendant role in this time slot. Unfortunately, no conclusion can be made because we cannot distinguish this case with the case where the good node does not have anything to send on the main channel.

The cell of the first row and first column is special: it has two corrupted packets on both channels. This situation only happens when the jammer jams on the main channel and there is a collision on the alibi channel. Obviously, if we assume a perfect TDMA scheduling on the alibi channel where no collisions can happen, this combination is unreachable. However, as shown later, this situation can happen when a random access mechanism is used on the alibi channel to avoid slander attacks.

E. Desired Properties

Desired properties for any alibi-based protocols are as follows.

Channel A \ Channel M	"X"	"M"	"#I"	"E"
"X"	Channel M jammed/ Collisions on Channel A	Not trusted	***Alibi(I)***	Abnormal
"A"	Not trusted	Not trusted	Normal	Abnormal
"#J"	***Alibi(J)***	Not trusted	Normal	Abnormal
"E"	No A-defendants	Not trusted	Normal	Abnormal

Fig. 3. State Combination

1) Completeness: Completeness property specifies that all alive and honest nodes eventually conclude who the jammer is. This property basically implies the termination condition. In our work we will consider two termination conditions of proposed alibi-based protocols.

- Termination condition T_1: Each alive and honest nodes has at least one alibi held by some honest nodes.
- Termination condition T_2: All honest nodes conclude the identification of the jammer.

The condition T_2 is harder to achieve because each node has to collect enough alibis of other nodes to identify the jammer.

2) Accuracy: This property is concerned about the false alarm and miss detection of any alibi-based protocols. Specifically, any alibi-based protocols must show that $P[false\_alarm]$ and $P[miss\_detection]$ are bounded.

3) Detection time: This property is concerned about the time to detect the jammer. Specifically, any alibi-based protocols must show that the time to detect is bounded and smaller detection time implies better performance.

Intuitively, the detection time depends directly on the speed of creation new alibis. Alibis are created from useful combinations of proofs on main and alibi channels. Therefore, the detection time is affected by the number of slots jammed and the number of successful A-defendants.

4) Availability: This property defines fraction of time the main channel is available for communication. If the main channel is always jammed, the availability is zero. If it is not jammed at all, the availability is 100%. This property and the previous property - the detection time - altogether imply the jammer strategy.

The jammer may decide to jam the main channel in a fraction of time. The more he jams on the main channel, the more he can damage the main channel at the cost of being detected faster. The only way for him to avoid getting caught is to stop the jamming action, which apparently lead to 100% availability of the main channel. The jammer may also decide to jam on the alibi channel in a fraction of time. This is equivalent to reduce the fraction of time he jams on the main channel (due to atomic jammer's limited capabilities) and thus increases the availability of the main channel.

Choosing how much to jam and what pattern to jam forms the strategy of the jammer. A smart jammer may have an

State	Possible channels	Description
"X"	M (or A)	The node saw a corrupted packet at that time due to either jamming effect or a collision on channel M (or A)
"M"	M	The node was M-defendant
"A"	A	The node was A-defendant
"#i"	M (or A)	The node saw N_i broadcasted a defendant message on channel M (or A)
"E"	M (or A)	The node did not see any activities

TABLE I

POSSIBLE VALUES OF STATES RECORDED BY WITNESSES

adaptive strategy to maximize possible damage on the main channel while minimizing probability of getting caught. This is out of scope of this paper and will be considered in the future work. In this work, we consider *a simple strategy where the attacker jams the main channel all the time*.

5) Scalability: This property specifies how much overhead is incurred in an alibi-based protocol and thus how well it scales with the network size. Specifically, it measures how many extra messages have to be sent for alibi schemes for a given network size.

IV. ALIBI PROTOCOLS

In this paper, we propose four alibi-based protocols. The first one is *the TDMA-like shuffle protocol*. This protocol assumes a random TDMA scheduling on the alibi channel. While this protocol is vulnerable to the slander attack, it motivates the need for other three protocols. The proof of its detection time is also easy to follow and is the base proof of the detection time of other protocols. The other three protocols we proposed use the random access mechanism on the alibi channel. The *Omniscient protocol* assumes an "Omniscient entity" who can gather and combine proofs to create alibis. *The K-Propagation protocol* is more practical than the Omniscient protocol in that it does not assume proofs are globally known immediately after their creation. Essentially, in this protocol, proofs in K previous time slots are included in the messages sent by defendants and appropriately combined to create alibis. However, this protocol only addresses the T_1 termination condition. It does not work well with the T_2 termination condition like the *L-gossiping protocol* does. The L-gossiping protocol extends the K-propagation protocol in that it adds the exchange of alibis to speed up the detection time for the T_2 termination condition.

A. TDMA-like shuffle Protocol

The TDMA-like shuffle protocol assumes a random TDMA scheduling on the alibi channel. In each round, each node has a unique assigned time slot like TDMA but the order of time slot maybe different from round to round. In other words, in each round the slot assignment in the alibi channel is a random permutation of the TDMA schedule on the main channel. In the subsequent sections, first we will analyze the performance of this protocol. Then, we will show that this protocol is vulnerable to the slander attacks and thus leads to a need for more robust randomized alibi-based protocols.

Theorem 1. *Under the T_1 termination condition and lossless channel condition, the expected detection time of the TDMA-like shuffle protocol is $O(n\ln(n))$ time slots with high probability.*

Proof. See Appendix. □

Theorem 2 (Slander attacks)**.** *The TDMA-like shuffle protocol is vulnerable to the slander attacks. Specifically, there exists a strategy for the jammer to defame a good node and to make this protocol never terminate (i.e. detection time to infinity).*

Proof. Because the shuffle TDMA scheduling on the alibi channel is known for every node, including the jammer N_J. He can defame a node N_i ($i \neq j$) as follows. Whenever N_i becomes an A-defendant, N_J will stop the jamming action n the main channel. Thus, N_i will never be able to get an alibi because there is always no jamming activity when he is the A-defendant. N_J can also do this for a set of good nodes to make the protocol never terminate. □

B. Random access alibi-based protocols

As shown in the previous sections, the problem of TDMA-like shuffle protocol is the *predictable schedule* on the alibi channel. Thus, the scheduling on the alibi channel needs to be randomized to give more confusion to the attacker. In the subsequent sections, we explore the use of random access on the alibi channel. Interestingly, as shown later, this class of protocols can achieve $O(n\ln n)$ time slots for the detection time. Even for very practical protocols where proofs and alibis are exchanged, the detection time is still the same order of magnitude.

In an alibi-based protocol employing random medium access mechanism on the alibi channel, a node becomes an A-defendant with a probability $p_{AD} = \frac{1}{n}$ in each time slot. Also, because if a node is not a defendant, it will become an A-witness or a M-witness with probability $p_{AW} = \frac{1}{2}$. This simple strategy has an advantage of unpredictability of who is the A-defendant in each time slot. This advantage helps to avoid the slander attacks. However, it comes with the cost of slower detection time due to collisions on the alibi channel.

C. Omniscient Protocol

The Omniscient protocol assumes an "omniscient entity" who knows proofs right after they are created. This Omniscient entity then can combine proofs to make alibis. The protocol is terminated until the Omniscient gathers enough $n-1$ different alibis to make the conclusion about the jammer. Apparently,

this scheme should achieve fastest average detection time in this class of protocols because there is no delay for exchanging and combining proofs. It is also important to emphasize that the Omniscient protocol performs the same under either T_1 or T_2 condition because all proofs are centrally and omnisciently gathered.

Theorem 3 (Detection time of the Omniscient protocol)**.** *Under the T_1 termination condition and lossless channel condition, for the Omniscient scheme, the fastest detection time is when $p_{AD} = \frac{1}{n}$ and $p_{AW} = \frac{1}{2}$ and is $O(n \ln(n))$ time slots with high probability.*

Proof. See Appendix. □

D. K-propagation Protocol

K-propagation protocol removes the unrealistic assumption about the "omniscient entity". In K-propagation protocol, each node keeps proofs it has created from the last K slots. Each node also includes these K-proofs into the proof messages it sends when becoming a defendant (see Section III-D). Therefore, proofs will have following format

$$(state : 16bit, \#timeslot : 16bit, state_1, ..., state_k : 16bit).$$

This format contains the format shown in Section III-D plus the K *states* from the last K slots.

Obviously, an issue with this protocol is that the size of exchanged messages grow with K - the number of proofs each node keeps for exchanging and combining. A constraint for K is that it has to be small enough such that the slot size does not exceed the σ_{max} and thus meets the necessary condition specified in Section III-C. Because this constraint varies according to the system's and attacker's parameters, the performance of K-propagation protocol also changes with K. Thus, to get a more predictable performance, we now will give an analysis for the case when $K = 1$. 1-propagation has a deterministic overhead and its performance is an upper bound for any K-propagation protocol where $K > 1$.

Surprisingly, we found that the 1-propagation protocol still achieves $O(n \ln n)$ time slots for detection time under T_1 termination condition.

Theorem 4 (Detection time of 1-propagation protocol under T_1 termination condition)**.** *Under the T_1 termination condition and lossless channel condition, the fastest expected detection time is when $p_{AD} = \frac{1}{n}$ and $p_{AW} = \frac{1}{2}$ and is $O(n \ln n)$ time slots with high probability.*

Proof. See Appendix. □

However, 1-propagation is much slower under T_2 termination condition. Under T_2 condition, each node has to gather enough $n - 1$ different alibis of the other nodes to make the conclusion. This step will require at least $(n - 1) \times ((n - 1) \ln(n - 1) + O(n))$ slots for each honest node. Furthermore, the protocol only terminates when all honest nodes can make the conclusion. Thus, under the T_2 condition, this 1-propagation protocol performs much slower, at least slower with in the order of n. Therefore, we propose L-gossiping protocol to speed up the detection time under T_2 condition.

E. L-gossiping Protocol

L-gossiping protocol speeds up the detection time under T_2 termination condition by exchanging alibis among nodes. In K-propagation protocol, each node keeps a bitmap of size n in which bit ith indicates that node N_i has an alibi. Similar to K-propagation protocol, each node also includes an array of identifiers randomly picked in its bitmap of size n into the proofs it sends when becoming a defendant (see Section III-D). Specifically, each node uniformly randomly picks L bits in its bitmap and includes only identifers corresponding with the picked 1-bits. Therefore, the format of the proof message is now extended to

$$(state : 16bit, \#timeslot : 16bit, state_1, ..., state_k : 16bit, no.id : 16bit, id_1, ...id_{no.id} : 16bit)$$

where *no. id* is the number of identifiers following after this field and id_i is the identifier of node N_{id_i} that has alibi.

Similar to K-propagation protocol, L-protocol gossiping also has an issue of message size keeping growing with L. Thus, L has to be small enough so that the necessary condition in Section III-C is not violated. Also, because the performance of this protocol depends much on the chosen value K according to system's and attacker's parameters, it is more interesting to investigate the performance of *1-gossiping* protocol.

Theorem 5 (Detection time of 1-gossiping protocol under T_2 termination condition)**.** *Under the T_2 termination condition and lossless channel condition, for 1-gossiping protocol, the fastest detection time is when $p_{AD} = \frac{1}{n}$ and $p_{AW} = \frac{1}{2}$ and is $O(n \ln(n))$ time slots with high probability.*

Proof. See Appendix. □

V. Other properties and extension of random access alibi-based protocols

A. False Alarm and Miss Detection rate

Theorem 6. *The false alarm and miss detection rates of the Omniscient protocol, the K-propagation protocol and the L-gossiping protocol are zero under lossless channel condition. For lossy channel model with p_l loss rate for both channels, the false alarm and miss detection rates are p_l.*

Proof. See Appendix. □

B. Extension to a generalized jammer's strategy

In the strategy considered in the above sections, the jammer is assumed to jam the main channel all the time. We now consider a generalized strategy in which the jammer only jams a p_m fraction of time on the main channel and a p_a fraction of time on the alibi channel. It is important to note that it does not matter the exact slots the jammer jams - only the fraction matters.

In general, the results derived for random access alibi-based protocols do not change much. Specifically, it only slows down the detection process because the number of potential slots that can lead to creation of alibis is proportionally reduced as shown in the following lemma. This will also affect the false alarm and miss detection probability.

Theorem 7 (Generalized jammer's strategy)**.** *If the atomic jammer jams p_m fraction of time on the main channel and p_a fraction of time on the alibi channel ($p_m + p_a \leq 1$), the detection speed of Omniscient protocol, K-propagation protocol and L-gossiping protocol is reduced by a factor of p_m and the availability is $1 - p_m$.*

Proof. See Appendix. □

VI. Evaluation

We evaluate the proposed protocols in ns2. We extend the built-in TDMA protocol in ns2 to implement the proposed alibi-based protocols. The packet size is 128 bytes. The bandwidth is 1Mbps. The slot size can handle a 256-byte packet. The number of nodes n is varied from 10 to 500. The attack jams the main channel all the time. We repeat the experiments 5 times to get the average and plot them on the graphs.

The detection time is shown in Figure 4. Figure 4(a) and 4(b) show the detection time in number of slots and seconds, respectively. Omniscient protocol has the smallest detection time as expected. 1-propagation is the next fast scheme. 1-gossiping is the slowest because it needs both 1-propagation process and gossiping process. It is important to note that the detection time of all schemes are bounded within $20n\log(n)$ as shown in the Figure 4(a).

The message overhead incurred by the three alibi-based protocols is shown in Figure 5. Omniscient protocol has least message overhead and is the base line for any alibi-based protocol due to the assumption of global knowledge. The message overhead of 1-propagation and 1-gossiping is not much compared to the Omniscient protocol. The crucial point in this figure is that the message overhead grows linear with the network size. It shows a good scalability of alibi-based protocols.

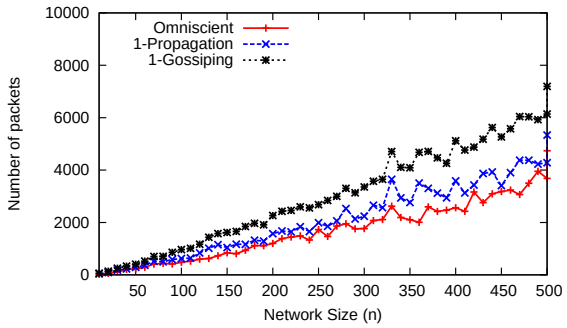


Fig. 5. Message Overhead

VII. Conclusion

We have shown a novel way to deal with insider-based jamming attacks. We have proposed a class of alibi-based protocols to detect the atomic jammers. Omniscient, 1-propagation and 1-gossiping protocol are shown to achieve $O(n\ln(n))$ time slots, zero false alarm and zero miss detection under the lossless channel condition. We also consider some practical aspects of these protocols under lossy channel condition and a more generalized jammer's strategy. We also show that simulation results in ns2 confirm our analysis.

The encouraging results in this paper are just a starting point. Dealing with stronger jammers, more general MAC protocol, multi-hops are some possible research directions. Some practical aspects such as channel switching delay have to be taken into account. Tighter bounds of detection time can be further investigated.

VIII. Appendix

Fact 1. *For any $y \geq 1$ and $|x| \leq 1$, we have*

$$(1 - x^2 y)e^{xy} \leq (1 + x)^y \leq e^{xy}$$

Lemma 1. *Let $c > 0$ be a constant, $m = n\ln n + cn$ for a positive integer n. Then for any constant k, we have*

$$\lim_{n\to\infty} \binom{n}{k}(1 - \frac{k}{n})^m = \frac{\exp(-ck)}{k!}$$

Proof. By the formula above, we have

$$(1 - \frac{k^2 m}{n^2})\exp(-\frac{km}{n}) \leq (1 - \frac{k}{n})^m \leq \exp(-\frac{km}{n})$$

We have $\lim_{n\to\infty}(1 - \frac{k^2 m}{n}) = 1$ and $\exp(-km/n) = n^{-k}\exp(-ck)$. Also,

$$\lim_{n\to\infty} \binom{n}{k}\frac{k!}{n^k} = \lim_{n\to\infty} \frac{n(n-1)...(n-k+1)}{n^k} = 1$$

Thus,

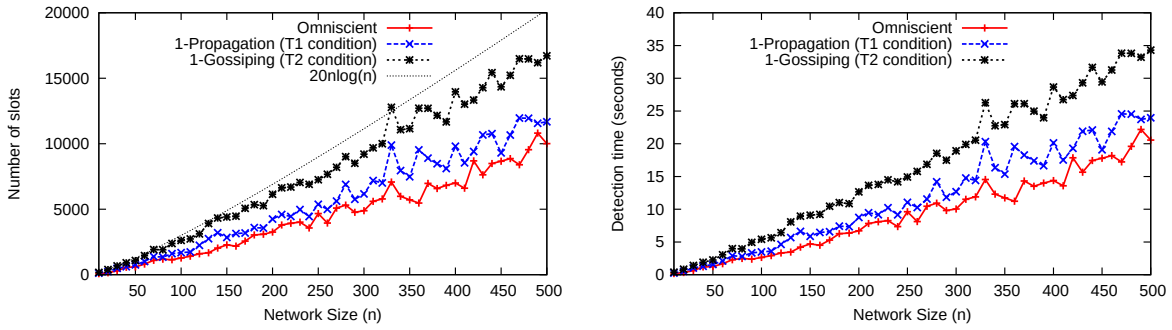
$$\lim_{n\to\infty} \binom{n}{k}(1 - \frac{k}{n})^m = \lim_{n\to\infty} \frac{n^k}{k!}\exp(-\frac{km}{n})$$

$$= \lim_{n\to\infty} \frac{n^k}{k!} n^{-k}\exp(-ck) = \frac{\exp(-ck)}{k!}$$

□

Proof of Theorem 1. The proof has two steps. In the first step, we will find out the probability of a given slot to get an (unnecessarily new) alibi and what is the maximum value of this probability. In the second step, we will calculate the expected number of slots such that each node gets an alibi by some nodes and prove that the expectation happens with very high probability. The analysis in the second step is similar to the analysis of the well-known *coupon collector's problem* [23].

Let us denote $p_{\text{alibi}}^{\text{shuffle}}$ as the probability of a given slot to get an alibi. Due to the TDMA-like shuffle protocol on the alibi channel, there is always an A-defendant in any time slot. Thus, for a given time slot, the alibi is created only when



(a) Detection time in #slots

(b) Detection time in seconds

Fig. 4. Detection Time

the main channel is jammed witnessed by at least one M-witness and the alibi channel has an A-defendant witnessed by at least one A-witness. The probability of the channel to get jammed is always 1 due to the considered jammer's strategy. The probability of at least one A-witness and one M-witness, denoted as p_W, is

$$p_W = 1 - p_{AW}^{n-3} - (1 - p_{AW})^{n-3}.$$

Thus,

$$p_{\text{alibi}}^{\text{shuffle}} = 1 \times p_W = 1 - p_{AW}^{n-3} - (1 - p_{AW})^{n-3}. \quad (1)$$

Because $p_{\text{alibi}}^{\text{shuffle}}$ is the probability of getting an alibi in any time slot, we want to maximize it. Because $p_{\text{alibi}}^{\text{shuffle}}$ is the function of p_{AW} for any given n, by applying first and second derivative it is easy to see that $p_{\text{alibi}}^{\text{shuffle}}$ is maximized when $p_{AW} = \frac{1}{2}$. Thus, $p_{\text{alibi}}^{\text{shuffle}} \to 1$ exponentially as n increases.

We now proceed to step 2. In this step, we want to calculate the expected number of slots E^{shuffle} to ensure that each node has at least one alibi. For any node N_i, $i = 1..n, N_i \neq N_J$,,

$\mathbb{P}r[N_i$ does not have any alibi in the first E^{shuffle} slots] = $(1 - \frac{p_{\text{alibi}}^{\text{shuffle}}}{n-1})^{E^{\text{shuffle}}} \approx e^{-\frac{p_{\text{alibi}}^{\text{shuffle}} E^{\text{shuffle}}}{n-1}}$

Thus, the expected number of *different* alibis after E^{shuffle} slots is

$$(n-1)(1 - e^{-\frac{p_{\text{alibi}}^{\text{shuffle}} E^{\text{shuffle}}}{n-1}})$$

Therefore, the expected number of different alibis when $E^{\text{shuffle}} = (n-1)\ln(n-1) + c(n-1), c > 0$ is

$$\mathbb{E}[\#\text{alibis}] = (n-1) - e^{-c} \quad (2)$$

The Equation 2 essentially shows that after $(n-1)\ln(n-1) + c(n-1)$, the expected number of different alibis is very close to $n-1$. In other words, each honest node gets at least one alibi after that many slots.

We now show that indeed, after $E^{\text{shuffle}} = (n-1)\ln(n-1) + c(n-1)$ slots, each node gets at least one alibi *with high probability*. Specifically, if we denote X_{n-1} the number of slots such that each node of $n-1$ nodes gets at least one alibi, we will prove that

$$\mathbb{P}r[X_{n-1} > E^{\text{shuffle}}] = 1 - \exp(-e^{-c})$$

Let E_i^{shuffle} denote the event that node N_i does not have any alibis after E^{shuffle} slots, we have

$$\mathbb{P}r[X_{n-1} > E^{\text{shuffle}}] = \mathbb{P}r[\bigcup_{i=1}^{n-1} E_i^{\text{shuffle}}]$$

. By inclusion-exclusion, we have

$$\mathbb{P}r[\bigcup_{i=1}^{n-1} E_i^{\text{shuffle}}] = \sum_{i=1}^{n-1} (-1)^{i+1} F_i^{n-1},$$

where

$$F_j^{n-1} = \sum_{1 \le i_1 < i_2 \ldots < i_j \le n-1} \mathbb{P}r[\bigcap_{k=1}^{j} E_{i_k}^{\text{shuffle}}]$$

Let $S_k^{n-1} = \sum_{i=1}^{k} (-1)^{i+1} F_i^{n-1}$. We know that

$$S_{2k}^{n-1} \le \mathbb{P}r[\bigcup_{i} E_i^{\text{shuffle}}] \le S_{2k+1}^{n-1}$$

.

By symmetry,

$$F_k^{n-1} = \binom{n-1}{k} \mathbb{P}r[\bigcap_{l=1}^{k} E_l^{\text{shuffle}}] = \binom{n-1}{k} (1 - \frac{k}{n-1})^{E^{\text{shuffle}}}.$$

Thus, $F_k = \lim_{n\to\infty} F_k^{n-1} = \exp -ck/k!$, by Lemma 1. Let

$$S_k = \sum_{j=1}^{k} (-1)^{j+1} F_j = \sum_{j=1}^{k} (-1)^{j+1} \frac{\exp(-cj)}{j!}$$

Clearly, $\lim_{k\to\infty} S_k = 1 - \exp(-e^{-c})$ by the Taylor expansion of $\exp(x)$ for $x = -e^{-c}$. Indeed,

$$\exp(x) = \sum_{j=0}^{\infty} \frac{x^j}{j!} = \sum_{j=0}^{\infty} \frac{(-e^{-c})^j}{j!} = 1 + \sum_{j=0}^{\infty} \frac{(-1)^j e^{-cj}}{j!}$$

Clearly, $\lim_{n\to\infty} S_k^{n-1} = S_k$ and $\lim_{k\to\infty} S_k = 1 - \exp(-e^{-c})$. Thus, we have

$$\lim_{n\to\infty} \mathbb{P}r[X > E^{\text{shuffle}}] = \lim_{n\to\infty} [\bigcup_{i=1}^{n-1} E_i^{\text{shuffle}}]$$

$$= \lim_{n\to\infty} \lim_{k\to\infty} S_k^{n-1} = \lim_{k\to\infty} S_k = 1 - \exp(-e^{-c})$$

.

□

Proof of Theorem 3. Similar to the proof of Theorem 1, this proof has also two steps. In the first step, we need to calculate the probability $p_{alibi}^{omniscient}$ to get an alibi in any given time slot and when it is maximized. Since the second step is very similar to the second step of Theorem 1, we will omit some redundant proofs.

An alibi is created only when there is an M-defendant, an A-defendant, at least one M-witness and one A-witness. For a given time slot, the probability of having an M-defendant is 1 due to the TDMA scheduling. The probability of having an A-defendant is $(n-2)p_{AD}(1-p_{AD})^{n-2}$ (n-2 because the jammer and the M-defendant is excluded). The probability of having at least an A-witness and an M-witness is

$$p_W = 1 - p_{AW}^{n-3} - (1-p_{AW})^{n-3}.$$

Thus,

$$p_{\text{alibi}}^{\text{omniscient}} = (n-2)p_{AD}(1-p_{AD})^{n-2}(1-p_{AW}^{n-3}-(1-p_{AW})^{n-3}).$$

We now want to see $p_{\text{alibi}}^{\text{omniscient}}$ is maximized at what value of p_{AD} and p_{AW}. Let us consider the term $p_{AD}(1-p_{AD})^{n-2}$. The derivative of the function with respect to the variable p_{AD} is

$$(1-p_{AD})^{n-3}(-np_{AD} + p_{AD} + 1)$$

Because $0 < p_{AD} < 1$, the term $p_{AD}(1-p_{AD})^{n-2}$ is maximized only when $p_{AD} = 1/(n-1)$. Similarly, as shown in Theorem 1, the term $(1-p_{AW}^{n-3}-(1-p_{AW})^{n-3})$ is maximized when $p_{AW} = 1/2$. By substituting $p_{AD} = 1/n$ and $p_{AW} = 1/2$, we will get

$$p_{\text{alibi}}^{\text{omniscient}} = \frac{n-2}{n-1}(1-\frac{1}{n-1})^{n-2}(1-(\frac{1}{2})^{n-2})$$

Since $(1-\frac{1}{n-1})^{n-2} = (\frac{n-2}{n-1})^{n-2} \geq \frac{1}{e}$ and $(\frac{1}{2})^{n-2}$ approaches 0 exponentially as n increases, the term $p_{\text{alibi}}^{\text{omniscient}}$ approaches 1 exponentially as n increases.

The second step of Theorem 1 is similar to the second step of the proof of Theorem 1. However, the detection time of the Omniscient scheme is larger than that of the TDMA-like shuffle scheme because the $p_{\text{alibi}}^{\text{omniscient}} < p_{\text{alibi}}^{\text{shuffle}}$.

□

Proof of Theorem 4. Similar to the proofs of Theorem 1 and Theorem 3, we start with calculating the probability $p_{\text{alibi}}^{\text{1-propagation}}$ of getting an alibi for any given time slot. Let us first calculate, at a given time slot k, the probability of two potential proofs (i.e. the proofs whose combination would turn into an alibi). Then, we will calculate the probability of those two proofs to be combined in the next time slot because they can be only propagated at most one time slot (1-propagation).

Let $S_M(k), S_A(k), R_M(k), R_A(k)$ be the set of M-defendants, A-defendants, M-witnesses and A-witnesses at time k, respectively. We have

$$\begin{aligned} R_A(k) \cup R_M(k) &= \{N_1 ... N_n\} \backslash \{S_A(k), S_M(k), N_J\} \\ |R_A(k)| + |R_M(k)| &= n-3 \\ R_A(k) \cap R_M(k) &= 0 \end{aligned}$$

Denote $p_{\text{alibi}}(k-1)$ as the probability of getting two potential proofs (i.e. those whose combination would turn into an alibi), $p_{\text{alibi}}(k|k-1)$ as the probability of getting an alibi at slot k from the two proofs propagated from slot $k-1$. We have

$$p_{\text{alibi}}^{\text{1-propagation}}(k) = p_{\text{alibi}}(k-1) \times p_{\text{alibi}}(k|k-1) \quad (3)$$

The first term $p_{\text{alibi}}(k-1)$ is the same as $p_{\text{alibi}}^{\text{omniscient}}$. The second term is

$$\begin{aligned} &p_{\text{alibi}}(k|k-1) = \\ &|R_A(k-1)|p_{AD}(1-p_{AD})^{n-2}(1-(1-p_{AW})^{|R_M(k-1)|}) \\ &+|R_M(k-1)|p_{AD}(1-p_{AD})^{n-2}(1-(1-p_{AW})^{|R_A(k-1)|}). \end{aligned}$$

The two terms in above equation refer to the two cases that can transform the two potential proofs to the alibi. The first case is when one of the A-witnesses in time slot $k-1$ becomes the M-defendant in time slot k and one of the M-witnesses in time slot $k-1$ becomes the A-witness in the time slot k. After getting out the common term $p_{AD}(1-p_{AD})^{n-2}$, we get

$$\begin{aligned} p_{\text{alibi}}(k|k-1) &= p_{AD}(1-p_{AD})^{n-2} \\ &\quad \times \Big(|R_A(k-1)|(1-(1-p_{AW})^{|R_M(k-1)|}) \\ &\quad + |R_M(k-1)|(1-(1-p_{AW})^{|R_A(k-1)|}) \Big) \\ &= p_{AD}(1-p_{AD})^{n-2} \\ &\quad \times \Big(|R_A(k-1)| + |R_M(k-1)| \\ &\quad - |R_A(k-1)|(1-p_{AW})^{|R_M(k-1)|} \\ &\quad - |R_M(k-1)|(1-p_{AW})^{|R_A(k-1)|} \Big) \end{aligned}$$

Because $|R_A(k-1)|(1-p_{AW})^{|R_M(k-1)|} + R_M(k-1)|(1-p_{AW})^{|R_A(k-1)|}$ is maximized when either $|R_A(k-1)| = 1$ or $|R_A(k-1)| = n-4$, we have

$$\begin{aligned} p_{\text{alibi}}(k|k-1) &\geq p_{AD}(1-p_{AD})^{n-2} \\ &\quad \times \big(1-(1-p_{AW})^{n-4} + (n-4)p_{AW}\big) \\ &\geq p_{AD}(1-p_{AD})^{n-2}(n-4)p_{AW} \end{aligned}$$

Substitute into Equation 3 and apply similar proofs shown in the Theorem 1 and Theorem 3, $p_{\text{alibi}}^{\text{1-propagation}}(k)$ is maximized when $p_{AD} = 1/n$ and $p_{AW} = 1/2$. Furthermore,

$p_{\text{alibi}}^{\text{1-propagation}}(k)$ will also approach 1 exponentially as n increases.

The second step of this proof is similar to the second step of the proof of Theorem 3. Thus, in 1-propagation scheme, after $(n-1)\ln(n-1)+f(n-1)$ slots, for any $f>0$, each node will get at least one alibi with high probability.

□

Proof of Theorem 5. The difference of this lemma and Theorem 4 is the termination condition. Thus, for any $f>0$, after $(n-1)\ln(n-1)+f(n-1)$, the probability that each node has an alibi kept by a node is $e^{-e^{-f}}$ (i.e. very high). Let us now consider the gossiping process separately. We want to see how many slots it takes to get all alibis gossiped to the whole network.

Let us now analyze how many steps for the alibis of an honest node N to get propagated to all other honest nodes. Denote X_t the number of honest nodes hold an alibi of node N. That means, $(n-1-X_t)$ is the number of honest nodes do not have any alibi of N. Without the loss of generality, we assume $X_0=1$. $t=0$ is the time when the first alibi of N is created.

The alibi gossiping process is similar to the epidemic process (i.e. the well-known S-curve). At the beginning when only few nodes hold the alibi of N, the alibi is propagated slowly. When there are reasonable of nodes holding alibi of N, the number of new nodes getting propagated will increase exponentially. At the end, when most of nodes already hold the alibi, the rate of getting the alibi of N propagated to new nodes is slow down. Formally, the rate of the alibi gossiping process can be expressed in the following differential equation.

$$\frac{d\mathbb{E}[X_t]}{dt}=\mathbb{E}[X_t]p_{AD}(1-p_{AD})^{n-2}\times\frac{1}{n-1}\times(n-1-\mathbb{E}[X_t])p_{AW} \tag{4}$$

In the Equation 4, the term in the left side expresses the rate of the alibi gossiping process. In the right hand side, the first term essentially shows the probability of a "gossiping" node in X_t that can successfully become an A-defendant and thus be able to propagate the alibi of N. The second term basically shows the probability that the alibi of N is picked and is included in the proofs of the A-defendant. $\frac{1}{n-1}$ is actually the lower bound of this probability because the number of alibis holding by any honest nodes are also less than or equal to $n-1$. The last term in the equation is the number of nodes among $(n-1-X_t)$ nodes that do not have the alibi of N to become the A-witness and thus will get the alibi of N.

Let $\beta=p_{AD}(1-p_{AD})^{n-2}\times\frac{1}{n-1}\times p_{AW}\approx\frac{1}{(n-1)^2}$ (substituting $p_{AD}=1/n$ and $p_{AW}=1/2$), Equation 4 becomes

$$\frac{d\mathbb{E}[X_t]}{dt}=\beta\mathbb{E}[X_t]\times(n-1-\mathbb{E}[X_t]). \tag{5}$$

This is a standard differential equation. Solving this equation gives us

$$\mathbb{E}[X_t]=\frac{n-1}{e^{-(n-1)\beta t+C_1}+1}. \tag{6}$$

where C_1 is a constant.

Since $X_0=1$, $C_1=\ln(n-2)$ and $e^{C_1}=n-2$. Thus, we have

$$\mathbb{E}[X_t]=\frac{n-1}{(n-2)e^{-(n-1)\beta t}+1}=\frac{n-1}{(n-2)e^{-t/(n-1)}+1}. \tag{7}$$

If we consider at $t'=g(n-1)\ln(n-1)$ for any $g>0$, we have

$$\begin{aligned}\mathbb{E}[X_{t'}] &= \frac{n-1}{(n-2)e^{-(g(n-1)\ln(n-1))/(n-1)}+1}\\ &\approx \frac{n-1}{1/n^g+1}\\ &\approx (n-1)(1-1/n^g)\\ &\approx (n-1)-1/n^g.\end{aligned}$$

That means, after $t'=g(n-1)\ln(n-1)$ time slots, the number of nodes know the alibi of N is very close to $n-1$. By applying the Markov inequality $\mathbb{P}r[X\geq 1]\leq\mathbb{E}[X]$, we have

$$\mathbb{P}r[X_{t'}\leq n-2]=\mathbb{P}r[n-1-X_{t'}\geq 1]\leq\mathbb{E}[n-1-X_{t'}]=n^{-g}. \tag{8}$$

Thus, it is shown that after $g(n-1)\ln(n-1)$ time slots, the alibi of N is propagated to $n-1$ nodes with very high probability. Furthermore, since we do not make any assumption about N, this statement applies for every honest node in the network.

Now, let us consider the overall process which consists of two sub-processes. The first process is the one 1-propagation process under T_1 condition. Denote T_1 the time this process to finish (i.e., when each node has at least one alibi). In Theorem 4, we proved that

$$\mathbb{P}r[T_1>(n-1)\ln(n-1)+c(n-1)]<1-e^{-e^{-c}}$$

The second process of the overall process is the gossiping process that will start right after the first process finishes (i.e., each node has one alibi holding by at least another node). Denote T_2 the number of time slots for the second process to propagate all the alibis to the whole network. We also have proved that

$$\mathbb{P}r[T_2>d(n-1)\ln(n-1)]<n^{-d}$$

Let $TT(n,c,d)=(n-1)\ln(n-1)+c(n-1)+d(n-1)\ln(n-1)$ and denote T the time that the overall process finishes (i.e. when all nodes in the network has all alibis of other nodes).

$$\begin{aligned}&\mathbb{P}r[T>TT(n,c,d)]\\ \leq\ &\mathbb{P}r[(T_1>(n-1)\ln(n-1)+c(n-1))\\ &\cup(T_2>d(n-1)\ln(n-1))]\\ \leq\ &\mathbb{P}r[T_1>(n-1)\ln(n-1)+c(n-1)]\\ &+\mathbb{P}r[T_2>d(n-1)\ln(n-1))]\\ =\ &1-e^{-e^{-c}}+n^{-d}\end{aligned}$$

That means, after $(n-1)\ln(n-1)(1+d)+c(n-1)$ time slots, all nodes are expected to get alibis of all other nodes with very high probability. □

Proof of Theorem 6. A false alarm happens when an honest node (i.e. victim) is concluded as the jammer. This only happens when the jammer has the alibi before the victim node gets its alibi. This implies that the jammer can get an alibi in some time slot. According to Figure 3, this only happens when one channel is jammed and the channel has a node, in this case, the jammer, to transmit its proof. This contradicts with the capability of the atomic jammer where the jammer cannot transmit on two channels in any given time slot. Therefore, under lossless channel condition, false alarm will be zero.

A miss detection happens when the jammer cannot never be concluded. In other words, the detection time goes to infinity. This means, there is at least one honest node beside the jammer that cannot get any alibis. This contradicts with the Theorem 1, Theorem 3 and and Theorem 5. Thus, under the lossless channel, no miss detection happens.

[Lossy channel] Under the lossy channel, false alarms and miss detections are possible. In our proposed schemes, a false alarm happens only when the jammer can get an alibi and thus avoid getting detected. Thus, the false alarm implies a miss detection. Similarly, a miss detection happens only when the jammer gets an alibi at finite time slot. Otherwise, if the jammer does get any alibis, he will be eventually detected according to our Theorem 1, Theorem 3 and Theorem 5. Thus, a miss detection also implies a false alarm.

We now calculate the probability of a jammer to get an alibi under the lossy channel with the rate p_l. In all of our proposed schemes, the jammer can get an alibi when a collision happens on the main channel and the jammer tries to follow the protocol to become the successful defendant on the alibi channel. In any proposed schemes that use random access mechanism (i.e. omniscient, 1-propagation and 1-gossiping), in any given time slot, the probability of a jammer to get an alibi is

$$p_{\text{alibi}}^{\text{jammer}} = p_l \times (1-p_{AD})^{n-1}(1-p_{AW}^{n-3}-(1-p_{AW})^{n-3}) \quad (9)$$

Because $p_{AD} = \frac{1}{n-1}$ and $p_{AW} = 1/2$ are chosen to maximize the detection probability, we have

$$p_{\text{alibi}}^{\text{jammer}} = p_l \times (1-\frac{1}{n-1})^{n-1}(1-(\frac{1}{2})^{n-2}) \quad (10)$$

Because $(1-\frac{1}{n})^{n-1} \geq \frac{1}{e}$, $p_{\text{alibi}}^{\text{jammer}} \rightarrow p_l$ exponentially as n increases. Thus, it becomes that for any time slot, the probability of a jammer to get an alibi approximates p_l and the expected number of slots for the jammer to get an alibi is $1/p_l$ (Bernoulli process).

□

Proof of Theorem 7. With the generalized jammer's strategy, for each time slot, the jammer will jam on the main channel with probability p_m, jam on the alibi channel with probability p_a and obey the protocol with probability $(1-p_m-p_a)$.

Intuitively, for the jammer, the less jamming activities implies the less damage to the network and the slower time to get detected. The main reason is the reduction of the number of slots getting corrupted by the jamming action on the main channel by the factor of p_m ($p_m = 1$ in the previous considered jammer's strategy). Apparently, the availability of the main channel now becomes $1-p_m$ (instead of 0 in the previous consider jammer's strategy).

We now consider the detection time of the Omniscient protocol under the generalized jammer's strategy. Similar to the proof in Theorem 3, we have

$$\begin{aligned} & \mathbb{P}r[\text{(n-1) nodes get at least one alibi in } E^{\text{omniscient}'} \text{ slots}] \\ \approx \quad & e^{-(n-1)e^{-\frac{p_{\text{alibi}}^{\text{omniscient}'} E^{\text{omniscient}'}}{n-1}}} \end{aligned}$$

where $p_{\text{alibi}}^{\text{omniscient}'} = p_m \times p_{\text{alibi}}^{\text{omniscient}}$ and $E^{\text{omniscient}'} > 0$. Thus, if

$$E^{\text{omniscient}'} = \frac{1}{p_m}E^{\text{omniscient}'} = \frac{1}{p_m}\big((n-1)\ln(n-1)+d(n-1)\big)$$

for any $d > 0$, we have

$$\mathbb{P}r[\text{(n-1) nodes get at least one alibi in } E^{\text{omniscient}'} \text{ slots}] \approx e^{-e^{-d}} \quad (11)$$

That means, under the considered generalized jammer's strategy, the detection time of the Omniscient protocol is slow down by the factor of p_m.

For the 1-propagation protocol and 1-gossiping protocol, a similar proof can be done. Therefore, under the generalized jammer's strategy, the detection time of the three random access alibi-based protocol (Omniscient, 1-propagation, 1-gossiping) is slow down by a factor of p_m and the availability of the main channel becomes $1-p_m$.

□

REFERENCES

[1] L. Fried, "Wave bubble: A design for a self-tuning portable rf jammer, http://www.ladyada.net/make/wavebubble/."

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

[3] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*. New York, NY, USA: ACM, 2005, pp. 46–57.

[4] W. Xu, T. Wood, W. Trappe, and Y. Zhang, "Channel surfing and spatial retreats: defenses against wireless denial of service," in *Proceedings of the 3rd ACM workshop on Wireless security (WiSe)*. New York, NY, USA: ACM, 2004, pp. 80–89.

[5] J. T. Chiang and Y.-C. Hu, "Dynamic jamming mitigation for wireless broadcast networks," in *IEEE Conference on Computer Communications (INFOCOM)*, 2008.

[6] W. Xu, W. Trappe, and Y. Zhang, "Channel surfing: defending wireless sensor networks from interference," in *Proceedings of the 6th international conference on Information processing in sensor networks (IPSN)*. New York, NY, USA: ACM, 2007, pp. 499–508.

[7] A. D. Wood, J. A. Stankovic, and G. Zhou, "Deejam: Defeating energy-efficient jamming in ieee 802.15.4-based wireless networks," in *IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, 2007.

Notation	Description
n	the number of nodes in the network
m	the number of channels that the network can switch to
Γ	the set of channels $\{C_1 ... C_m\}$ that the network can switch to
M, A	main channel and alibi channel, respectively
BS	the base station
N_i	the node ith
N_J	the jammer
p_{AD}	probability of nodes to become defendants at each time slot, $p_{AD} = \frac{1}{n}$
p_{AW}	probability of nodes to be come a witnesses at each time slot, $p_{AW} = \frac{1}{2}$
σ^{min}	the smallest slot size that the attacker is still a jammer
σ^{max}	the biggest slot size that the attacker is still an atomic jammer
τ	channel switching delay of a node
s	network slot size
p_m	number of slots in a round that the attacker jams on the main channel
p_a	number of slots in a round that the attacker jams on the alibi channel
p_l	loss rate of the main channel and alibi channel
p_W	probability of there is at least one A-witness and at least one M-witness
$S_M(k), S_A(k)$	the set of M-defendants, A-defendants at time slot k, respectively
$R_M(k), R_A(k)$	the set of M-witnesses, A-witnesses at time slot k, respectively

TABLE II

TABLE OF NOTATIONS

[8] G. Alnifie and R. Simon, "A multi-channel defense against jamming attacks in wireless sensor networks," in *Proceedings of the 3rd ACM workshop on QoS and security for wireless and mobile networks (Q2SWinet)*. New York, NY, USA: ACM, 2007, pp. 95–104.

[9] J. M. McCune, E. Shi, A. Perrig, and M. K. Reiter, "Detection of denial-of-message attacks on sensor network broadcasts," in *SP '05: Proceedings of the 2005 IEEE Symposium on Security and Privacy*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 64–78.

[10] Y. W. Law, L. van Hoesel, J. Doumen, P. Hartel, and P. Havinga, "Energy-efficient link-layer jamming attacks against wireless sensor network mac protocols," in *SASN '05: Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*. New York, NY, USA: ACM, 2005, pp. 76–88.

[11] R. K. Mallik, R. A. Scholtz, and G. P. Papavassilopoulos, "Analysis of an on-off jamming situation as a dynamic game," *IEEE Transaction on Communications*, vol. 48, pp. 1360–1373, 2000.

[12] M. Li, I. Koutsopoulos, and R. Poovendran, "Optimal jamming attacks and network defense policies in wireless sensor networks," in *IEEE Conference on Computer Communications (INFOCOM)*, 2007, pp. 1307–1315.

[13] V. Navda, A. Bohra, S. Ganguly, and D. Rubenstein, "Using channel hopping to increase 802.11 resilience to jamming attacks," in *IEEE Conference on Computer Communications (INFOCOM) Minisymposium*, Anchorage, AK, May 2007.

[14] T. X. Brown, J. E. James, and A. Sethi, "Jamming and sensing of encrypted wireless ad hoc networks," in *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*. New York, NY, USA: ACM, 2006, pp. 120–130.

[15] P. Kyasanur and N. H. Vaidya, "Detection and handling of mac layer misbehavior in wireless networks," in *International Conference on Dependable Systems and Networks (DSN)*, vol. 00. Los Alamitos, CA, USA: IEEE Computer Society, 2003, p. 173.

[16] S. Radosavac, J. S. Baras, and I. Koutsopoulos, "A framework for mac protocol misbehavior detection in wireless networks," in *Proceedings of the 4th ACM workshop on Wireless security (WiSe)*. New York, NY, USA: ACM, 2005, pp. 33–42.

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

[18] M. Z. Win and R. A. Scholtz, "Ultra-wide bandwidth time-hopping spread-spectrum impulse radio for wireless multiple-access communications," *IEEE Transactions on Communications*, vol. 48, 2000.

[19] "IEEE standard 802.11, http://standards.ieee.org/getieee802/802.11.html," September 2004.

[20] 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.

[21] E. Rescorla, "Diffie-hellman key agreement method," RFC 2631.

[22] X. Liu, R. Rajaraman, G. Noubir, B. Thapa, C. King, and E. Bayraktaroglu, "On the performance of ieee 802.11 under jamming," in *IEEE Conference on Computer Communications (INFOCOM)*, 2008.

[23] R. Motwani and P. Raghavan, *Randomized Algorithms*. Cambridge University Press, 1995.