Selective Jamming/Dropping Insider Attacks in Wireless Mesh Networks

Loukas Lazos, and Marwan Krunz
Dept. of Electrical and Computer Engineering, University of Arizona, Tucson, Arizona
Email: {llazos, krunz}@ece.arizona.edu

***Abstract*—Wireless mesh networks (WMNs) promise to extend high-speed wireless connectivity beyond what is possible with the current WiFi-based infrastructure. However, their unique architectural features leave them particularly vulnerable to security threats. In this article, we describe various forms of sophisticated attacks launched from adversaries with internal access to the WMN. We further identify possible detection and mitigation mechanisms.**

***Index Terms*—Security, wireless mesh networks, jamming, misbehavior, insider attacks, packet dropping.**

I. Introduction

Wireless mesh networks (WMNs) continue to receive significant interest as a possible means of providing seamless data connectivity, especially in urban environments [1]. Architecturally, such networks evolved from classic mobile ad hoc networks, targeting long-range transmissions with emphasis on network throughput and connectivity. WMN applications include stationary deployments (e.g., community networks, hierarchal sensor networks) as well as mobile ones (e.g., intelligent transportation systems, tactical military networks).

WMNs follow a two-tier network architecture [2]. The first tier consists of the end users, also referred to as stations (STAs), directly connected to mesh nodes, referred to as Mesh Access Points (MAPs). The second tier consists of a peer-to-peer network of the MAPs. Connectivity in the second tier is assisted by intermediate routers known as Mesh Points (MPs) which interconnect MAPs (MPs do not accept connections from end users). The network of MAPs and MPs is often static and uses separate frequency bands to communicate data and control information (MAPs are typically equipped with multiple transceivers). Finally, Mesh Gateways (MGs) provide connectivity to the wired infrastructure. An example of a WMN is shown in Fig. 1.

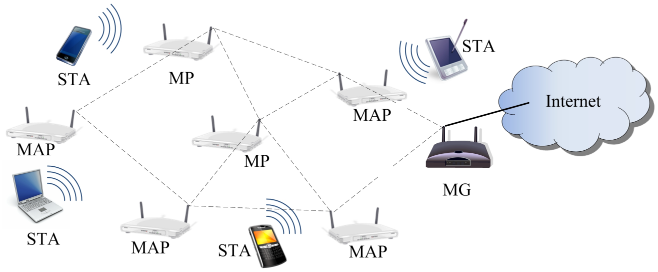

Fig. 1. WMN architecture.

WMNs are invariably vulnerable to “external” and “internal” attacks. External attacks take the forms of random channel jamming, packet replay, and packet fabrication, and are launched by “foreign” devices that are unaware of the network secrets (e.g., cryptographic credentials and pseudo-random spreading codes). They are relatively easier to counter through a combination of cryptography-based and robust communication techniques.

In contrast, internal attacks, which are launched from compromised nodes, are much more sophisticated in nature. These attacks exploit knowledge of network secrets and protocol semantics to *selectively and adaptively* target critical network functions. Attack selectivity can be achieved, for example, by overhearing the first few bits of a packet [3], or classification of transmissions based on protocol semantics [4]. Internal attacks, henceforth referred to as *insider attacks*, cannot be mitigated using only proactive methods which rely on network secrets, because the attacker already has access to such secrets. They additionally require protocols with built-in security measures, through which the attacker can be detected and its selective nature can be neutralized.

Vulnerabilities of WMNs: While all types of wireless networks are susceptible to insider attacks, WMNs are particularly vulnerable to them, for a number of reasons. First, MPs and MAPs are relatively cheap devices with poor physical security, which makes them potential targets for node capture and compromise. Second, given their relatively advanced hardware (e.g., multiple transceivers per MP and MAP), WMNs often adopt a multi-channel design, with one or more channels dedicated for control/broadcast purposes. Such static design makes it easier for an attacker to selectively target control/broadcast information. Third, the reliance on multi-hop routes further accentuates the WMN vulnerability to compromised relays which can drop control messages, in order to enforce a certain routing behavior (e.g., force packets to follow long or inconsistent routes).

In this paper, we discuss various forms of sophisticated attacks in WMNs, in which an insider adversary intelligently exploits knowledge of leaked cryptographic secrets and of protocol semantics to attack critical network functions such as channel access, routing, and end-to-end reliable data delivery. We focus our attention on insider attacks which take the form of selective jamming and/or dropping of "high-value" packets in any given layer or combination of layers. Whereas selective jamming aims at preventing reception while the packet is in transmission, selective dropping is applied post-reception. Besides describing such attacks, we also highlight possible detection and mitigation mechanisms.

II. Selective Jamming Attacks

The open nature of the wireless medium leaves it vulnerable to jamming attacks. Jamming in wireless networks has been primarily analyzed under an external adversarial model, as a severe form of denial of service (DoS) against the PHY layer. Existing anti-jamming strategies employ some form of spread spectrum (SS) communication, in which the signal is spread across a large bandwidth according to a pseudo-noise (PN) code. However, SS can protect wireless communications only to the extent that the PN codes remain secret. Insiders with knowledge of the commonly shared PN codes can still launch jamming attacks. Using their knowledge of the protocols specifics, they can selectively target particular channels/layers/protocols/packets. We describe two types of selective jamming attacks against WMNs, which employ channel and data selectivity.

A. *Channel-Selective Jamming*

In a typical WMN, one or more channels are reserved for broadcasting control information. These channels, referred to as *control channels*, facilitate operations such as network discovery, time synchronization, coordination of shared medium access, routing path discovery and others, without interfering with the communications of STAs with MAPs. An adversary who selectively targets the control channels can efficiently launch a DoS attack with a fairly limited amount of resources (control traffic is low-rate compared to data traffic). To launch a channel-selective jamming attack, the adversary must be aware of the location of the targeted channel, whether defined by a separate frequency band, time slot, or PN code. Note that control channels are inherently broadcast and hence, every intended receiver must be aware of the secrets used to protect the transmission of control packets. The compromise of a single receiver, be it a MAP or an MP, reveals those secrets to the adversary.

Example: We illustrate the impact of channel-selective jamming on CSMA/CA-based medium access control (MAC) protocols for multi-channel WMNs. A multi-channel MAC (MMAC) protocol is employed to coordinate access of multiple nodes residing in the same collision domain to the common set of channels. A class of MMAC protocols proposed for ad hoc networks such as WMNs follows a split-phase design (e.g., [5]). In this design, time is split into alternating control and data transmission phases. During the control phase, every node converges to a default channel to negotiate the channel assignment. In the data transmission phase, devices switch to the agreed on channels to perform data transmissions. The alternating phases of a split-phase MMAC are shown in Fig. 2.

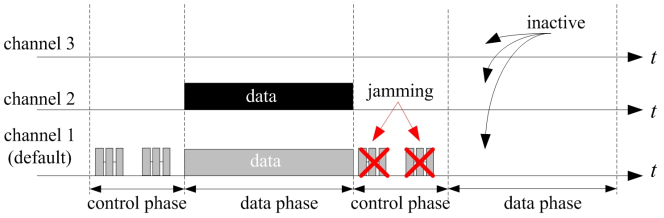

Fig. 2. A MMAC protocol that uses a split-phase design. Channel-selective jamming of the default channel during the control phase prevents the use of all channels during the data transmission phase.

By employing a channel-selective strategy, an inside adversary can jam only the default channel and only during the control phase. Any node that is unable to access the default channel during the control phase must defer the channel negotiation process to the next control phase, thus remaining inactive during the following data transmission phase. This attack is illustrated in Fig. 2. Note that the impact of this channel-selective jamming attack propagates to all frequency bands at a low energy overhead, since only a single channel is targeted and only for a fraction of time.

B. *Countering Channel-Selective Attacks*

Several anti-jamming methods have been proposed to address channel-selective attacks from insider nodes. All methods trade communication efficiency for stronger resilience to jamming. We give a brief description of such anti-jamming approaches.

Replication of control information: An intuitive approach to counter channel-selective jamming is to repeat control information on multiple broadcast channels [6]. In this case, an insider with limited hardware resources cannot jam all broadcasts simultaneously. Moreover, if

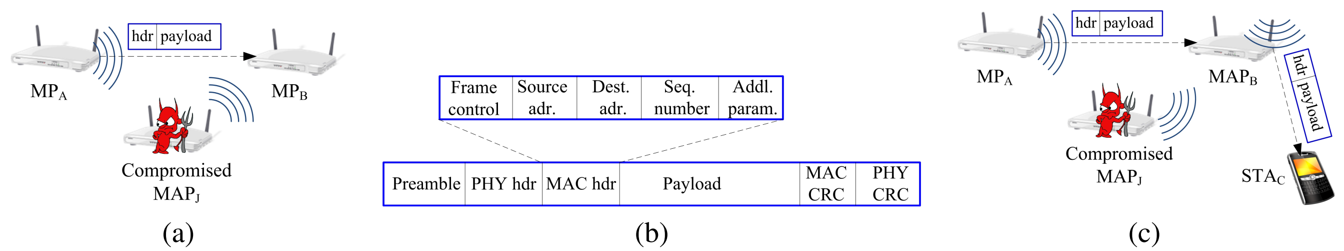

Fig. 3. (a) A data-selective jamming attack, (b) generic packet format, (c) inference of a RREP transmission on link MAP$_B$-STA$_C$ based on the RREP transmission on link MP$_A$-MAP$_B$.

each node has only partial knowledge of the locations of the broadcast channels, an insider can target only the subset of channels known to him. Due to the limited number of available channels, this scheme provides protection against a small number of colluding attackers.

Assignment of unique PN codes: An alternative method for neutralizing channel-selective attacks is to dynamically vary the location of the broadcast channel, based on the physical location of the communicating nodes [7]. The main motivation for this architecture is that any broadcast is inherently confined to the communication range of the broadcaster. Hence, for broadcasts intended for receivers in different collision domains, there is no particular advantage in using the same broadcast channel, other than the design simplicity. The assignment of different broadcast channels to different network regions leads to an inherent partitioning of the network into clusters. Information regarding the location of the control channel in one cluster cannot be exploited at another. Moreover, broadcast communication can be repaired locally should a jammer appear, without the need for re-establishing a global broadcast channel.

To protect the control channel within each cluster, following cluster formation, one mesh node is elected as the clusterhead (CH). The CH assigns its cluster members unique PN hopping sequences, that have significant overlap. The common locations among these PN sequences implement a broadcast channel. If an insider uses his PN sequence to jam this broadcast channel, it becomes uniquely identifiable by the CH. Once identified, the CH updates all nodes of the cluster with new PN sequences, except for the identified attacker.

The idea of assigning unique PN codes to various nodes in the network was also exploited in [8]. In this work, nodes of a cluster are represented by the leaves of a binary tree. Each node of the tree is assigned a unique key, corresponding to a seed for the generation of a unique PN code. Every node knows all the keys along the path from the corresponding leaf to the root. In the absence of jamming, the PN code known to all receivers (generated by the root key) is used. If jamming is detected, transmitting nodes switch to a PN code known only to a subset of nodes. The compromised node is uniquely identified in a number of steps that is logarithmic to the number of nodes within the cluster.

Elimination of secrets: Selective insider jamming attacks can be countered by avoiding secrets in the first place. In the design proposed in [9], a transmitter randomly selects a PN code from a public codebook. To recover a transmitted packet, receivers must record the transmitted signal and attempt decoding it using every PN code in the codebook. Because the PN code used to spread each packet is not known a priori, an inside adversary can only attempt to guess it, with a limited probability of success. Special care needs to be given to the synchronization between the communicating parties (knowing the PN code is essential for discovering and "locking onto" the transmitted signal).

C. Data-Selective Jamming

To further improve the energy efficiency of selective jamming and reduce the risk of detection, an inside attacker can exercise a greater degree of selectivity by targeting specific packets of high importance. One way of launching a data-selective jamming attack, is by classifying packets before their transmission is completed. An example of this attack is shown in Fig. 3(a). MP$_A$ transmits a packet to MP$_B$. Inside attacker MAP$_J$ classifies the transmitted packet after overhearing its first few bytes. MAP$_J$ then interferes with the reception of the rest of the packet at MP$_B$.

Referring to the generic packet format in Fig. 3(b), a packet can be classified based on the headers of various layers. For example, the MAC header typically contains information about the next hop and the packet type. The TCP header reveals the end-to-end source and destination nodes, the transport-layer packet type (SYN, ACK, DATA, etc.), and other TCP parameters.

Another method for packet classification is to anticipate a transmission based on protocol semantics. As an example, consider the routing function in WNMs, described in the IEEE 802.11s standard [2]. Routing is performed at the MAC layer according to the Hybrid Wireless Mesh Protocol (HWMP). The latter is a combination of tree-based routing, and on-demand routing based on AODV. Tree-based routing provides fixed path routes from the mesh nodes to the MGs. On demand routing is employed to discover routes to mobile STAs who associate with multiple MAPs due to their mobility. Consider the route discovery process depicted in Fig. 3(c). MP_A transmits a route reply (RREP) to MAP_B, which is overheard by MAP_J. MAP_J can conjecture that MAP_B will forward the RREP to STA_C, and hence, jam this RREP while it is in transit to STA_C.

Packet classification can also be achieved by observing implicit packet identifiers such as packet length, or precise protocol timing information [4]. For example, control packets are usually much smaller than data packets. The packet length of an eminent transmission can be inferred by decoding the network allocation vector field (NAV) of request-to-send (RTS) and clear-to-send (CTS) messages, used for reserving the wireless medium.

D. Countering Data-Selective Jamming Attacks

An intuitive solution for preventing packet classification is to encrypt transmitted packets with a secret key. In this case, the entire packet, including its headers, has to be encrypted. While a shared key suffices to protect point-to-point-communications, for broadcast packets, this key must be shared by all intended receivers. Thus, this key is also known to an inside jammer. In symmetric encryption schemes based on block encryption, reception of one ciphertext block is sufficient to obtain the corresponding plaintext block, if the decryption key is known. Hence, encryption alone does not prevent insiders from classifying broadcasted packets.

To prevent classification, a packet must remain hidden until it is transmitted in its entirety. One possible way for temporarily hiding the transmitted packet is to employ commitment schemes. In a commitment scheme, the transmitting node hides the packet by broadcasting a committed version of it. The contents of the packet cannot be inferred by receiving the commitment (hiding property). After the transmission is completed, the node releases a de-commitment value, which reveals the original packet. The commitment scheme must be carefully designed to prevent the classification of the original packet based on the partial release of the de-commitment value. Another approach is to use public hiding transformations that do not rely on secrets. An example of them is all-or-nothing transformations (AONTs), which were originally proposed to slow down brute force search attacks against encryption schemes. An AONT serves as a publicly known and completely invertible pre-processing step for a plaintext, before it is passed to an encryption algorithm. The defining property of an AONT is that the entire output of the transformation must be known before any part of the input can be computed. In our context, an AONT prevents packet classification when the AONT of a packet is transmitted over the wireless medium.

III. Selective Dropping Attacks

If selective jamming is not successful due to anti-jamming measures, an insider can selectively drop packets post-reception. Once a packet has been received, the compromised node can inspect the packet headers, classify the packet, and decide whether to forward it or not. Such an action is often termed as *misbehavior* [10]–[13]. Post-reception dropping is less flexible than selective jamming because the adversary is restricted to dropping only the packets routed through it. Nonetheless, the impact on the WMN performance can be significant.

Examples: Consider a compromised MP targeting the routing functionality in WMNs. By selectively dropping route request and route reply packets employed by the routing protocol, as defined in the of the 802.11s standard [2], the compromised MP can prevent the discovery of any route that passes through it, delay the route discovery process, and force alternative, possibly inefficient paths.

Alternatively, the compromised MP can allow the establishment of a route via itself, but throttle the rate of the end-to-end connection at the transport layer. This attack can be actualized by selective dropping of critical control packets that regulate the end-to-end transmission rate and effective throughput. For example, the dropping of cumulative TCP acknowledgments results in the end-to-end retransmission of the entire batch of pending data packets (see Fig. 4). In addition, packet loss is interpreted as congestion, resulting in the throttling of the sender's transmission rate.

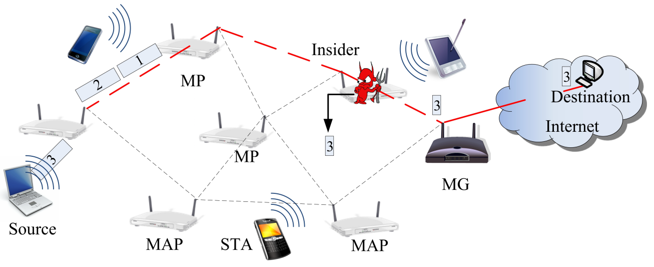

Fig. 4. An insider selectively drops cumulative TCP acknowledgments and forces end-to-end data retransmissions.

In another selective strategy known as the *Jellyfish attack*, a compromised mesh node that periodically drops a small fraction of consecutive packets can effectively reduce the throughput of a TCP flow to near zero [14]. This attack can be achieved even by inducing random delays to TCP packets, without dropping them, while remaining protocol compliant [14]. Similar selective dropping attacks can be constructed for other network functions such as the association/de-association of STAs, and topology management, to name a few.

A. Mitigation of Selective Dropping

Selective dropping attacks can be mitigated by employing fault-tolerant mechanisms at various layers of the protocol stack. At the routing layer, multi-path routing provides robust multi-hop communication in the presence of network faults, by utilizing more than one path from a source to a destination. Tree-based routing in HWMP already provisions for back-up paths to the MG [2]. At the transport layer, variants of the standardized TCP protocol have been specifically developed for dealing with the imperfections of the wireless medium [15]. These protocols differentiate between congestion and wireless transmission losses. A selective dropper can always attribute his losses to congestion, in order to avoid detection as a malicious node. In this case, identification mechanisms employing long-term statistics, can accurately pinpoint selective droppers.

B. Identification of Selective Droppers

Current methods for detecting misbehavior in self-organizing systems such as WMNs, can be classified into reputation systems [12], credit-based systems [13], and acknowledgment systems [10].

Reputation Systems: Reputation systems identify misbehaving nodes based on per-node reputation metrics, computed based on interactions of each node with its peers. These systems typically incorporate two critical operations: (a) the collection of accurate observations of nodes' behavior and, (b) the computation of the reputation metric.

Behavioral information is collected based on first-hand observations provided by neighboring nodes and second-hand information provided by other interacting peers [12]. First-hand observations are collected by monitoring nodes which operate in promiscuous mode in order to verify the correct forwarding of transmitted packets. Overhearing becomes problematic in the case of multi-channel WMNs, because MPs and MAPs are scheduled to communicate in parallel over orthogonal frequency bands, and hence, they might not be available to monitor the behavior of other nodes. Several schemes have been proposed for managing second-hand information. A node may flood warnings to the entire network, if it detects a misbehaving node. Alternatively, information can be provided on-demand, after a request from a particular node has been received. In the latter scenario, flooding of the request is necessary to discover nodes that possess second-hand information. Both methods consume considerable bandwidth resources due to the underlying flooding operations for the dissemination and collection of second-hand information.

Robust computation of reputation metrics is equally important for the identification of packet droppers. Simple aggregate metrics have been shown to be vulnerable to false accusations from colluding malicious nodes, and suddenly changing behavioral patterns. For instance, a misbehaving node can exhibit a long history of good behavior in order to build a high reputation metric, before it starts to misbehave. Such instances are dealt by assigning larger weights to recent behavioral observations and/or adopting additive increase-multiplicative decrease type of algorithms for updating the reputation metrics [12].

A critical challenge for any metric computation algorithm is the selective nature of packet droppers. When a very small fraction of packets is dropped, metrics that do not take into account the packet type are bound to have high rates of misdetection. Dropping selectivity can be detected with the use of storage-efficient reports (e.g., based on Bloom filters) of the per-packet behavior of nodes [11]. Based on these reports, it is possible to conduct multiple tests to identify malicious selective dropping patterns. These patterns are likely to have some deterministic structure compared to packet losses due to congestion or poor channel quality.

ACK-based systems: ACK-based schemes differ from overhearing techniques in the method of collecting first-hand behavioral observations. Downstream nodes (more than a single hop away) are responsible for acknowledging the reception of messages to nodes several hops upstream [10]. These systems are suitable for monitoring the faithful relay of unicast traffic, at the expense of communication overhead for relaying an additional set of ACKs. However, ACK-based schemes cannot be used to identify insiders that selectively drop broadcast packets. Such packets remain, in general, unacknowledged in wireless networks, to avoid an ACK implosion situation. Moreover, a small set of colluding nodes can still provide authentic ACKs to upstream nodes while dropping packets.

Credit-based systems: Credit-based systems alleviate selfish behavior by incentivising nodes to forward packets [13]. Nodes that relay traffic receive credit in return, which can be later spent to forward their own traffic. However, in the context of WNMs, MPs do not generate any traffic of their own, but act as dedicated relays. Hence, compromised MPs have no incentive for collecting credit. Moreover, in the case of selective dropping attacks, misbehaving nodes can still collect sufficient credit by forwarding packets of low importance, while dropping a few packets of "high value." In addition, the credit collected by a particular node depends on the topology of the network. A highly connected node is expected to collect more credit due to the increased volumes of traffic routed through it. An adversary compromising such a node is likely able to implement a selective dropping strategy without running out of credit. Finally, credit-based systems lack a mechanism for identifying the misbehaving node(s), allowing them to remain within the network indefinitely.

IV. Discussion and Conclusions

WMNs are prone to various external and internal security threats. While most external attacks can be mitigated with a combination of cryptographic mechanisms and robust communication techniques, internal attacks are much harder to counter because the adversary is aware of the network secrets and its protocols. Jamming-resistant broadcast communications in the presence of inside jammers remains a challenging problem. Current solutions attempt to eliminate the use of common secrets for protecting broadcast communications. Such secrets can be easily exposed in the event of node compromise. However, the heightened level of security comes at the expense of performance, because broadcasted messages have to be transmitted multiple times and on multiple frequency bands to guarantee robust reception.

Moreover, even if packet reception of critical messages is ensured, inside adversaries are in complete control of the traffic routed through them. A large body of literature addresses the problem of misbehavior in the form of packet dropping by developing reputation systems, credit-based systems, and communication-intensive acknowledgment schemes. Despite the relative wealth of literature on this problem, significant challenges are yet to be addressed. Most existing methods assume a continuously active adversary that systematically drops packets. These adversaries are detected by aggregate behavioral metrics such as per-packet reputation and credit. However, these metrics cannot detect attacks of selective nature, where only a small fraction of "high value" packets is targeted. Furthermore, when the adversary drops only a few packets, his behavior can be indistinguishable from dropping patterns due to congestion or poor wireless conditions. Further challenges include efficient behavioral monitoring mechanisms not relying on continuous overhearing and efficient maintenance and dissemination of reputation metrics.

Acknowledgments

This research was supported in part by NSF (under grants CNS-1016943, CNS-0844111, CNS-0721935, CNS-0904681, IIP-0832238), Raytheon, and the "Connection One" center. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] I.F. Akyildiz, X. Wang, and W. Wang. Wireless mesh networks: a survey. *Computer Networks*, 47(4):445–487, 2005.

[2] IEEE P802.11s/D1.01 standard. At https://mentor.ieee.org/802.11/dcn/07/11-07-0335-00-000s-tgs-redline-between-draft-d1-00-and-d1-01.pdf, 2007.

[3] Alejandro Proano and Loukas Lazos. Selective jamming attacks in wireless networks. In *Proceedings of the IEEE International Conference on Communications (ICC)*, 2010.

[4] 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*, 2006.

[5] J. So and N.H. Vaidya. Multi-channel MAC for ad hoc networks: handling multi-channel hidden terminals using a single transceiver. In *Proceedings of the ACM MobiHoc Conference*, pages 222–233, 2004.

[6] P. Tague, M. Li, and R. Poovendran. Probabilistic mitigation of control channel jamming via random key distribution. In *Proceedings of the International Symposium in Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 1–5, 2007.

[7] L. Lazos, S. Liu, and M. Krunz. Mitigating control-channel jamming attacks in multi-channel ad hoc networks. In *Proceedings of the Second ACM Conference on Wireless Network Security (WiSec)*, pages 169–180, 2009.

[8] Jerry Chiang and Yih-Chun Hu. Cross-layer jamming detection and mitigation in wireless broadcast networks. In *Proceedings of the ACM MobiCom Conference*, pages 346–349, 2007.

[9] Christina Pópper, Mario Strasser, and Srdjan Capkun. Jamming-resistant broadcast communication without shared keys. In *Proceedings of the USENIX Security Symposium*, 2009.

[10] K. Liu, J. Deng, P.K. Varshney, and K. Balakrishnan. An acknowledgment-based approach for the detection of routing misbehavior in MANETs. *IEEE Transactions on Mobile Computing*, 6(5):536–550, 2007.

[11] W. Kozma and L. Lazos. Dealing with liars: Misbehavior identification via Rényi-Ulam games. In *Security and Privacy in Communication Networks*, pages 207–227, 2009.

[12] Han Yu, Zhiqi Shen, Chunyan Miao, C. Leung, and D. Niyato. A survey of trust and reputation management systems in wireless communications. *Proceedings of the IEEE*, 98(10):1755 –1772, 2010.

[13] Y. Zhang, W. Lou, W. Liu, and Y. Fang. A secure incentive protocol for mobile ad hoc networks. *Wireless Networks*, 13(5):569–582, 2007.

[14] Imad Aad, Jean-Pierre Hubaux, and Edward W. Knightly. Impact of denial of service attacks on ad hoc networks. *IEEE/ACM Transactions on Networking*, 16(4):791–802, 2008.

[15] J. Liu and S. Singh. ATCP: TCP for mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 19(7):1300–1315, 2002.