Enhanced Packet Delivery Techniques Using Packet Hiding Schemes on Jamming Attacks

Surya. N
M.E Computer Science and Engineering, Final Year,
M.A.R College of Engineering and Technology,
Pudukkottai Dist-621316, Tamil Nadu

***Abstract*— The wireless medium leaves it vulnerable to intentional interference attacks, typically referred to as jamming. This intentional interference with wireless transmissions can be used as a launch pad for mounting jamming attacks on wireless networks. Typically, jamming bas been addressed under an external threat model. The internal knowledge of protocol specifications and network secrets can launch low-effort jamming attacks that are difficult to detect and counter. Here finding the problem of selective jamming attacks in wireless networks. The advantages of selective jamming in terms of network performance degradation and adversary effort by presenting two case studies; a selective attack on TCP and one on routing. An intuitive solution to selective jamming would be the encryption of transmitted packets (including headers) with a static key. However, for broadcast communications, this static decryption key must be known to all intended receivers and hence, is susceptible to compromise. Moreover, even if the encryption key of a hiding scheme were to remain secret, the static portions of a transmitted packet could potentially lead to packet classification.**

Index Terms- Selective jamming, denial-of-service, wireless networks, packet classification.

I. INTRODUCTION

WIRELESS networks rely on the uninterrupted availability of the wireless medium to interconnect participating nodes. However, the open nature of this medium leaves it vulnerable to multiple security threats. Anyone with a transceiver can eavesdrop on wireless transmissions, inject spurious messages, or jam legitimate ones. While eaves-dropping and message injection can be prevented using cryptographic methods, jamming attacks are much harder to counter. In the simplest form of jamming, the adversary interferes with the reception of messages by transmitting a continuous jamming signal [1], or several short jamming pulses [2]. Typically, jamming attacks have been considered under an external threat model, in which the jammer is not part of the network. Under this model, jamming strategies include the continuous or random transmission of high-power interference signals [6]. However, adopting an "al-ways-on" strategy has several disadvantages. First, the adversary has to expend a significant amount of energy to jam frequency bands of interest. Second, the continuous presence of unusually high interference levels makes this type of attacks easy to detect [2], [6]. Conventional anti jamming techniques rely extensively on spread-spectrum (SS) communications [1], or some form of jamming evasion.

Spread-Spectrum (SS) techniques provide bit-level protection by spreading bits according to a secret pseudo noise (PN) code, known only to the communicating parties. These methods can only protect wireless transmissions under the external threat model. Potential disclosure of secrets due to node compromise neutralizes the gains of SS. Broadcast communications are particularly vulnerable under an internal threat model because all intended receivers must be aware of the secrets used to protect transmissions. Hence, the compromise of a single receiver is sufficient to reveal relevant cryptographic information.

In this paper, we address the problem of jamming under an internal threat model. The adversary exploits his internal knowledge for launching selective jamming attacks in which specific messages of "high importance" are targeted. For example, a jammer can target route-request/route-reply messages at the routing layer to prevent route discovery, or target TCP acknowledgments in a TCP session to severely degrade the throughput of an end-to-end flow. To launch selective jamming attacks, the adversary must be capable of implementing a "classify-then-jam" strategy before the completion of a wireless transmission. the jammer may decode the first few bits of a packet for recovering useful packet identifiers such as packet type, source and destination address. After classification, the adversary must induce a sufficient number of bit errors so that the packet cannot be recovered at the receiver.

In this paper, our contributions that we investigate the feasibility of real-time packet classification for launching selective jamming attacks, under an internal threat model. We show that such attacks are relatively easy to actualize by exploiting knowledge of network protocols and cryptographic primitives extracted from compromised nodes. We investigate the impact of selective jamming on critical network functions. Our findings indicate that selective jamming attacks lead to DoS with very low effort on behalf of the jammer. To mitigate selective jamming attacks, we develop three schemes that prevent classification of transmitted packets in real time. Our schemes rely on the joint consideration of cryptographic mechanisms with PHY-layer attributes. We analyze the security of our schemes and show that they achieve strong security properties, with minimal impact on the network performance.

Fig. 1. (a) Realization of a selective jamming attack.

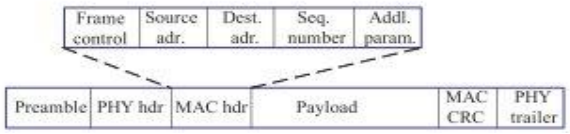

Fig. 1. (b) A generic frame format for a wireless network

II. PROBLEM STATEMENT AND ASSUMPTIONS

A. Problem Statement

Consider the scenario depicted in Fig. 1a. Nodes A and B communicate via a wireless link. Within the communication range of both A and B, there is a jamming node J. When A transmits a packet m to B, node J classifies m by receiving only the first few bytes of m. J then corrupts m beyond recovery by interfering with its reception at B. We address the problem of preventing the jamming node from classifying m in real time, thus mitigating J's ability to perform selective jamming. Our goal is to transform a selective jammer to a random one.

B. System and Adversary Model

1. Network Model

The network consists of a collection of nodes connected via wireless links. Nodes may communicate directly if they are within communication range, or indirectly via multiple hops. Nodes communicate both in unicast mode and broadcast mode. Communications can be either unencrypted or encrypted. For encrypted broadcast communications, symmetric keys are shared among all intended receivers. These keys are established using preshared pairwise keys or asymmetric cryptography.

2. Communication Model

Packets are transmitted at a rate of R bauds. Spread-spectrum techniques such as frequency hopping spread spectrum (FHSS), or direct sequence spread spectrum (DSSS) may be used at the PHY layer to protect wireless transmissions from jamming. SS provides immunity to interference to some extent (typically 20 to 30 dB gain), but a powerful jammer is still capable of jamming data packets of his choosing. Transmitted packets have the generic format depicted in Fig. 1b. The preamble is used for synchronizing the sampling process at the receiver. The PHY-layer header contains information regarding the length of the frame, and the transmission rate. The MAC header determines the MAC protocol version, the source and destination addresses, sequence numbers plus some additional fields. The MAC header is followed by the frame body that typically contains an ARP packet or an IP datagram. Finally, the MAC frame is protected by a cyclic redundancy check (CRC) code. At the PHY layer, a trailer may be appended for synchronizing the sender and receiver.

3. Adversary Model

We assume the adversary is in control of the communication medium and can jam messages at any part of the network of his choosing. The adversary can operate in full-duplex mode, thus being able to receive and transmit simultaneously. For analysis purposes, we assume that the adversary can proactively jam a number of bits just below the ECC capability early in the transmission. He can then decide to irrecoverably corrupt a transmitted packet by jamming the last symbol. In reality, it has been demonstrated that selective jamming can be achieved with far less resources [3]. A jammer equipped with a single half-duplex transceiver is sufficient to classify and jam transmitted packets. Our model captures a more potent adversary that can be effective even at high transmission speeds.

The implementation details of every layer of the network stack are assumed to be public. Furthermore, the adversary is capable of physically compromising network devices and recovering stored information including cryptographic keys, PN codes, etc. This internal adversary model is realistic for network architectures such as mobile ad hoc, mesh, cognitive radio, and wireless sensor networks (WSNs), where network devices may operate unattended, thus being susceptible to physical compromise.

III. REAL-TIME PACKET CLASSIFICATION

In this section, we describe how the adversary can classify packets in real time, before the packet transmission is completed. Once a packet is classified, the adversary may choose to jam it depending on his strategy. Consider the generic communication system depicted in Fig. 2. At the PHY layer, a packet m is encoded, interleaved, and modulated before it is transmitted over the wireless channel. At the receiver, the signal is demodulated, deinterleaved, and decoded to recover the original packet m.

The adversary's ability in classifying a packet m depends on the implementation of the blocks in Fig. 2. The channel encoding block expands the original bit sequence m, adding necessary redundancy for protecting m against channel errors. At the next block, interleaving is applied to protect m from burst errors. For simplicity, we consider a block interleaver that is defined by a matrix. The deinterleaver is simply the transpose of A. Finally, the digital modulator maps the received bit stream to symbols of length q, and modulates them into suitable waveforms for transmission over the wireless channel.

From our analysis, it is evident that intercepting the first few symbols of a packet is sufficient for obtaining relevant header information.

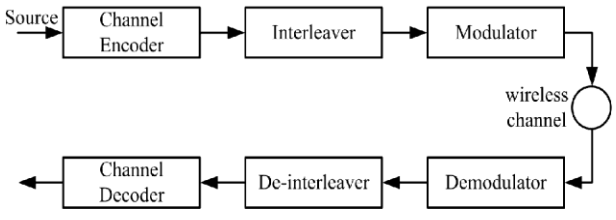

Fig. 2 A generic communication systems diagram.

An intuitive solution to selective jamming would be the encryption of transmitted packets (including headers) with a static key. However, for broadcast communications, this static decryption key must be known to all intended receivers and hence, is susceptible to compromise. One solution to the key compromise problem would be to update the static key whenever it is compromised. However, such a solution is not useful if the compromised node obtains the new key. This can only be avoided if there is a mechanism by which the set of compromised nodes can be identified. Such a task is nontrivial when the leaked key is shared by multiple nodes. Any node that possesses the shared key is a candidate malicious node. Moreover, even if the encryption key of a hiding scheme were to remain secret, the static portions of a transmitted packet could potentially lead to packet classification.

IV. PACKET HIDING SCHEME

To mitigate selective jamming, we combine cryptographic mechanisms.

The details of the scheme are presented in this section.

A. A strong hiding commitment scheme (SHCS)

We propose a strong hiding commitment scheme (SHCS), which is based on symmetric cryptography. Our main motivation is to satisfy the strong hiding property while keeping the computation and communication overhead to a minimum. The computation overhead of SHCS is one symmetric encryption at the sender and one symmetric decryption at the receiver. Because the header information is permuted as a trailer and encrypted, all receivers in the vicinity of a sender must receive the entire packet and decrypt it, before the packet type and destination can be determined.

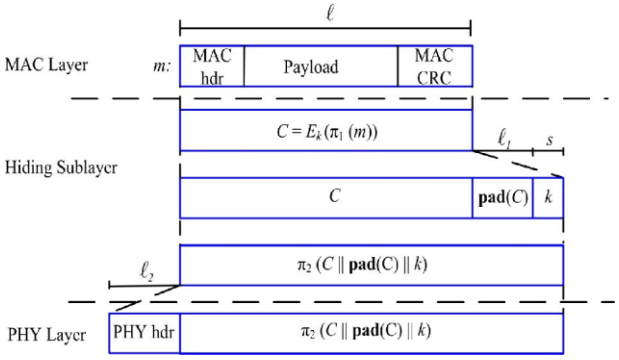

Fig. 3 Processing at the hiding sublayer

The proposed SHCS requires the joint consideration of the MAC and PHY layers. To reduce the overhead of SHCS, the decommitment value d (i.e., the decryption key k) is carried in the same packet as the committed value C. This saves the extra packet header needed for transmitting d individually. To achieve the strong hiding property, a sublayer called the "hiding sublayer" is inserted between the MAC and the PHY layers. This sublayer is responsible for formatting m before it is processed by the PHY layer. The functions of the hiding sublayer are outlined in Fig. 3.

A padding function pad () appends pad(C) bits to C, making it a multiple of the symbol size. Finally, C||pad(C) ||k is permuted by applying a publicly known permutation π2. The purpose of π2 is to ensure that the interleaving function applied at the PHY layer does not disperse the bits of k to other symbols.

B. Cryptographic Puzzle Hiding Scheme (CPHS)

Let a sender S has a packet m for transmission. The sender selects a random key k , of a desired length. S generates a puzzle (key, time), where puzzle() denotes the puzzle generator function, and tp denotes the time required for the solution of the puzzle. Parameter is measured in units of time, and it is directly dependent on the assumed computational capability of the adversary, denoted by N and measured in computational operations per second. After generating the puzzle P, the sender broadcasts (C, P). At the receiver side, any receiver R solves the received puzzle to recover key and then computes. Cryptographic Puzzle includes two types of scheme.

1. Time-lock Puzzles

2. Puzzles based on hashing

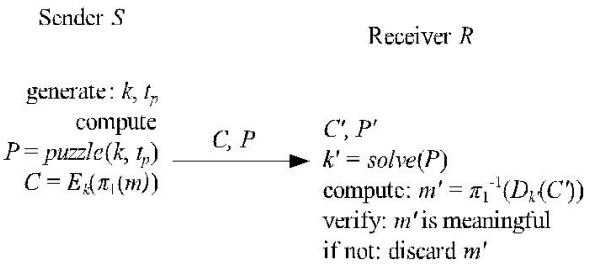

Fig. 4 The cryptographic puzzle-based hiding scheme

Time-lock Puzzles proposed a construction called time-lock puzzles, which is based on the iterative application of a precisely controlled number of modulo operations. Time-lock puzzles have several attractive features such as the fine granularity in controlling tp and the sequential nature of the computation. Moreover, the Puzzle generation requires significantly less computation compared to puzzle solving. Computationally limited receivers can incur significant delay and energy consumption when dealing with modulo arithmetic. Fig.4 shows the details of CPHS. In this case, CPHS can be implemented from cryptographic puzzles which employ computationally efficient cryptographic primitives. Client puzzles proposed in, use one-way hash functions with partially disclosed inputs to force puzzle solvers search through a space of a precisely controlled size. In our context, the sender picks a random key k with k = k1||k2. The lengths of k1 and k2 are s1, and s2, respectively. He then computes C = Ek(π1(m)) and transmits (C, k1, h(k)) in this particular order. To obtain k, any receiver has to perform On average 2s2−1 hash operations (assuming perfect hash functions). Because the puzzle cannot be solved before h (k) has been received, the adversary cannot classify m before the completion of m's transmission.

C. Hiding based on All-Or-Nothing Transformations (AONTs)

We propose a solution based on All-or- Nothing Transformations that introduces a modest communication and computation overhead. Such transformations were originally proposed by Rivest to slow down brute force attacks against block encryption algorithms [4].

The steps of AONT are shown in fig.5. The packets are pre-processed by an AONT before transmission but remain unencrypted. The jammer cannot perform packet classification until all pseudo-messages corresponding to the original packet have been received and the inverse transformation has been applied. The Packet m is partitioned to a set of x input blocks m = {m1, m2, m3….}, which serve as an input to a set of pseudo-messages m = {m1, m2, m3,…..} is transmitted over the wireless medium. We propose a solution based on All-Or- Nothing Transformations (AONT) that introduces a modest communication and computation overhead.

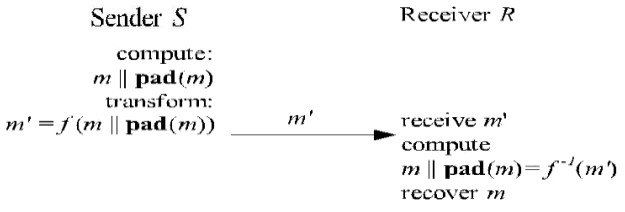

Fig. 5 The AONT-based hiding scheme

When a plaintext is preprocessed by an AONT before encryption, all ciphertext blocks must be received to obtain any part of the plaintext. Therefore, brute force attacks are slowed down by a factor equal to the number of ciphertext blocks, without any change on the size of the secret key. Note that the original AONT proposed in [4] is computationally secure. Several AONT schemes have been proposed that extend the definition of AONT to undeniable security [7]. Under this model, all plaintexts are equiprobable in the absence of at least one pseudomessage.

V. PERFORMANCE EVALUATION

In this section, we evaluate the impact of our packet-hiding techniques on the network performance via extensive simulations. We chose a set of nodes running 802.11b at the PHY and MAC layers, AODV for route discovery, and TCP at the transport layer. Aside from our methods, we also implemented a simple MAC layer encryption with a static key.

A. Impact on Real-Time Systems

Our packet-hiding methods require the processing of each individual packet by the hiding sublayer. We emphasize that the incurred processing delay is acceptable, even for real-time applications. The SCHS requires the application of two permutations and one symmetric encryption at the sender, while the inverse operations have to be performed at the receiver. Such operations can be implemented in hardware very efficiently. Symmetric encryption such as AES can be implemented at speeds of tens of Gbps/sec when realized with Application Specific Integrated Circuits (ASICs) or Field Programmable Gate Arrays (FPGAs) [5]. These processing speeds are orders of magnitude higher than the transmission speeds of most current wireless technologies, and hence, do not impose a significant delay. Similarly, the AONT-HS performs linear operations on the packet that can be efficiently implemented in hardware. We note that a non-negligible processing delay is incurred by the CPHS. This is due to the cryptographic puzzle that must be solved at the receiver. CPHS should only be employed when the symbol size at the PHY layer is too small to support the SHCS and AONT-HS solutions. The processing delays of the various schemes are taken into account in our experimental evaluations.

B. Experimental Evaluation

In the first set of experiments, we set up a single file transfer between a client and server, connected via a multihop route. The client requested a 1 MB file from the server. We evaluated the effects of packet hiding by measuring the effective throughput of the TCP connection in the following scenarios:

1. No packet hiding (N.H.).
2. MAC-layer encryption with a static key (M.E.).
3. SHCS (C.S.).
4. Time-lock CPHS (T.P.).
5. Hash-based CPHS (H.P.).
6. Linear AONT-HS (L.T.).
7. AONT-HS based on the package transform (P.T.).

In Fig.6a, we show the effective throughput averaged over 100 different traces. We observe that MAC-layer encryption, SHCS, and the linear AONT achieve an effective throughput close to the throughput in the absence of packet hiding. This is justified by the relatively small communication overhead of each hiding method and the small queuing delay at intermediate routers due to the absence of any cross traffic. The AONT based on the package transform achieved slightly lower throughput, because it occurs a per-packet overhead of 128 bits as opposed to 56 bits for SHCS. We also observe that hiding techniques based on cryptographic puzzles decrease the effective throughput of the TCP connection to half, compared to the no hiding case.

This performance is anticipated since the time required to solve a puzzle after a packet has been received at the MAC layer is equal to the transmission time of each packet. While this constitutes a significant performance reduction, we emphasize that cryptographic puzzles were suggested as a candidate solution only when the symbol size is so small that more efficient hiding methods do not provide adequate levels of security.

In the second set of experiments, the impact of packet hiding on the route discovery process in an ad hoc network. We generated a random topology of 54 nodes placed in an area of 500×400 m2. Nodes discovered routes using the AODV routing protocol. The average route discovery delay is shown in Fig.6b. This delay is defined as the time difference between the transmission of the first RREQ from a source and the reception of the corresponding RREP from the destination. We observe that the impact of packet hiding

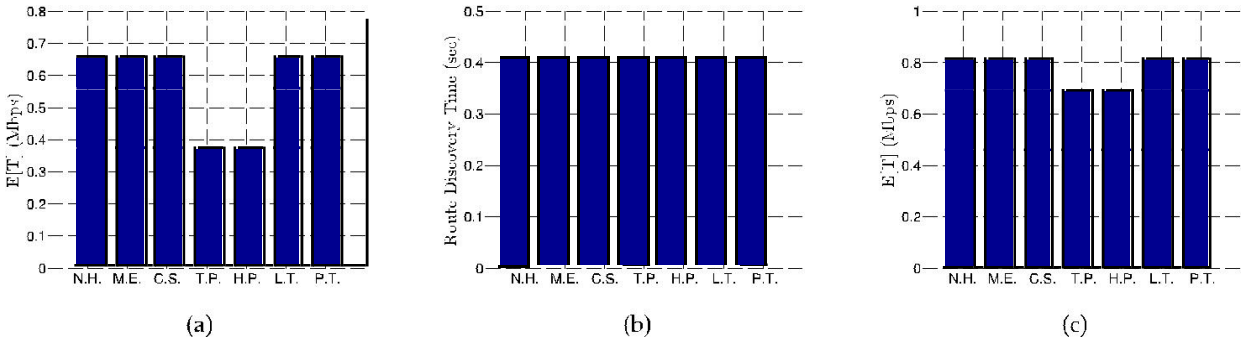

Fig. 6. (a) Average effective throughput. (b) Average route discovery time. (c) Average effective throughput.

on the route discovery delay is minimal compared to the case where no packet hiding is employed.

In order to discover a route, the originating node sends an RREQ with a time-to-live (TTL) value equal to one hop, and waits for the corresponding RREP. If the RREP is not received before a time-out value (to) expires, the originating node increases the TTL and the time-out to, and rebroadcasts the RREQ. This process is repeated until a valid RREP is received, or the TTL value exceeds the maximum diameter of the network.

In the third set of experiments, we evaluated the performance of TCP in a congested ad hoc network. We considered the same network topology used in the second set of experiments. In Fig. 6c, we show the effective throughput averaged over all 20 TCP connections. We observe that efficient packet-hiding techniques such as SHCS and AONT-HS have a relatively small impact on the overall throughput. This is because in a congested network, the performance is primarily dependent on the queuing delays at the relay nodes. The communication overhead introduced by the transmission of the packet-hiding parameters is small and hence, does not significantly impact the throughput. On the other hand, for CPHS, we observe a performance reduction of 25-30 percent compared to the case of no packet hiding. This reduction is attributed to the delay introduced by CPHS for the reception of each packet. Note that in the congested network scenario, the throughput reduction of CPHS is smaller compared to the noncongested one because nodes can take advantage of the queuing delays to solve puzzles.

VI. RELATED WORK

Jamming attacks on voice communications have been launched since the 1940s [1]. In the context of digital communications, the jamming problem has been addressed under various threat models.

In [7], the impact of an external selective jammer who targets various control packets at the MAC layer. To perform packet classification, the adversary exploits interpacket timing information to infer eminent packet transmissions. In [9], we proposed the estimation of the probability distribution of interpacket transmission times for different packet types based on network traffic analysis. Future transmissions at various layers were predicted using estimated timing information. Using their model, we proposed selective jamming strategies for well-known sensor network MAC protocols.

Selective jamming attacks have been experimentally implemented using software-defined radio engines [3], [10]. The success rate of a selective jamming attack against a802.15.4network was measured to be 99.96percent. selective jamming attacks against the rate adaptation mechanism of 802.11[3]. They showed that a selective jammer targeting specific packets in a point-to-point 802.11 communication was able to reduce the rate of the communication to the minimum value of 1 Mbps, with relatively little effort (jamming of five to eight packets per second). The results were experimentally verified using the USRP2/GNU radio platform.

We have suggested channel-selective jamming attacks, in which the jammer targets the broadcast control channel. It was shown that such attacks reduce the required power for performing a DoS attack by several orders of magnitude [11]. To protect control-channel traffic, the replication of control transmission in multiple channels was suggested in [11], [12], [13]. The "locations" of the control channels were cryptographically protected. In [14], a randomized frequency hopping algorithm to protect the control channel from inside jammers. A frequency hopping antijamming technique that does not require the existence of a secret hopping sequence, shared between the communicating parties [15].

Conventional methods for mitigating jamming employ some form of SS communications [16], [1]. The transmitted signal is spread to a larger bandwidth following a PN sequence. Without the knowledge of this sequence, a large amount of energy (typically 20-30 dB gain) is required to interfere with an ongoing transmission. However, in the case of broadcast communications, compromise of commonly shared PN codes neutralizes the advantages of SS. A jamming-resistant communication model for pairwise communications that does not rely on shared secrets. Communicating nodes use a physical- layer modulation method called Uncoordinated Direct- Sequence Spread Spectrum (UDSSS) [17]. They also proposed a jamming-resistant broadcast method in which transmissions are spread according to PN codes randomly selected from a public codebook [17].

VII. CONCLUSION

We address the problem of selective jamming attacks in networks. We considered an internal adversary model in which the jammer is part of the network under attack, thus being aware of the protocol specifications and shared network secrets. We showing that the jammer can classify the packets in real time by decoding the first few symbols of an ongoing transmission. We evaluate the impact of selective jamming attacks on network protocols such as TCP and routing. Our findings show that a selective jammer can significantly impact performance with very low effort. We are developing and survey on three schemes that transform a selective jammer to a random one by preventing real-time packet classification.

REFERENCES

[1] M.K. Simon, J.K. Omura, R.A. Scholtz, and B.K. Levitt, Spread Spectrum Communications Handbook. McGraw-Hill, 2001.
[2] G. Noubir and G. Lin, "Low-Power DoS Attacks in Data Wireless Lans and Countermeasures," Mobile Computing and Comm. Rev., vol. 7, no. 3, pp. 29-30, 2003.
[3] B. Thapa, G. Noubir, R. Rajaramanand, and B. Sheng, "On the Robustness of IEEE802.11 Rate Adaptation Algorithms against Smart Jamming," Proc. ACM Conf. Wireless Network Security (WiSec), 2011.
[4] R. Rivest, "All-or-Nothing Encryption and the Package Transform," Proc. Int'l Workshop Fast Software Encryption, pp. 210-218, 1997.
[5] K. Gaj and P. Chodowiec, "FPGA and ASIC Implementations of AES," Cryptographic Engineering, pp. 235-294, Springer, 2009.
[6] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The Feasibility of Launching and Detecting Jamming Attacks in Wireless Networks," Proc. ACM Int'l Symp. Mobile Ad Hoc Networking and Computing (MobiHoc), pp. 46-57, 2005.
[7] D. Stinson, "Something about All or Nothing (Transforms)," Designs, Codes and Cryptography, vol. 22, no. 2, pp. 133-138, 2001.
[8] D. Thuente and M. Acharya, "Intelligent Jamming in Wireless Networks with Applications to 802.11 b and Other Networks," Proc. IEEE Military Comm. Conf. (MILCOM), 2006.
[9] Y.W. Law, M. Palaniswami, L.V. Hoesel, J. Doumen, P. Hartel, and P. Havinga, "Energy-Efficient Link-Layer Jamming Attacks against WSN MAC Protocols," ACM Trans. Sensor Networks, vol. 5, no. 1, pp. 1-38, 2009.
[10] M. Wilhelm, I. Martinovic, J. Schmitt, and V. Lenders, "Reactive Jamming in Wireless Networks: How Realistic Is the Threat," Proc. ACM Conf. Wireless Network Security (WiSec), 2011.
[11] A. Chan, X. Liu, G. Noubir, and B. Thapa, "Control Channel Jamming: Resilience and Identification of Traitors," Proc. IEEE Int'l Symp. Information Theory (ISIT), 2007.
[12] P. Tague, M. Li, and R. Poovendran, "Probabilistic Mitigation of Control Channel Jamming via Random Key Distribution," Proc. IEEE Int'l Symp. Personal, Indoor and Mobile Radio Comm. (PIMRC), 2007.
[13] P. Tague, M. Li, and R. Poovendran, "Mitigation of Control Channel Jamming under Node Capture Attacks," IEEE Trans. Mobile Computing, vol. 8, no. 9, pp. 1221-1234, Sept. 2009.
[14] L. Lazos, S. Liu, and M. Krunz, "Mitigating Control-Channel Jamming Attacks in Multi-Channel Ad Hoc Networks," Proc. Second ACM Conf. Wireless Network Security, pp. 169-180, 2009.
[15] M. Strasser, C. Po ¨pper, and S. Capkun, "Efficient Uncoordinated fhss Anti-Jamming Communication," Proc. ACM Int'l Symp. Mobile Ad Hoc Networking and Computing (MobiHoc), pp. 207-218, 2009.
[16] Y. Desmedt, "Broadcast Anti-Jamming Systems," Computer Networks, vol. 35, nos. 2/3, pp. 223-236, Feb. 2001.
[17] C. Po ¨pper, M. Strasser, and S. Capkun, "Jamming-Resistant Broadcast Communication without Shared Keys," Proc. USENIX Security Symp., 2009.
[18] IEEE, IEEE 802.11 Standard, http://standards.ieee.org/ getieee802/download/802.11-2007.pdf, 2007.
[19] R.C. Merkle, "Secure Communications over Insecure Channels," Comm. ACM, vol. 21, no. 4, pp. 294-299, 1978.
[20] T.X. Brown, J.E. James, and A. Sethi, "Jamming and Sensing of Encrypted Wireless Ad Hoc Networks," Proc. ACM Int'l Symp. Mobile Ad Hoc Networking and Computing (MobiHoc), pp. 120-130, 2006.