

When WatchDog Meets Coding

Guanfeng Liang and Nitin Vaidya
 Department of Electrical and Computer Engineering
 University of Illinois at Urbana-Champaign
 Champaign, Illinois, USA
 Email: {gliang2, nhv}@illinois.edu

(Technical Report, May 27, 2009)

Abstract—In this work we study the problem of misbehavior detection in wireless networks. A commonly adopted approach is to utilize the broadcasting nature of the wireless medium and have nodes monitor their neighborhood. We call such nodes the *Watchdogs*. In this paper, we first show that even if a watchdog can overhear all packet transmissions of a flow, any linear operation of the overheard packets can not eliminate miss-detection and is inefficient in terms of bandwidth. We propose a lightweight misbehavior detection scheme which integrates the idea of watchdogs and error detection coding. We show that even if the watchdog can only observe a fraction of packets, by choosing the encoder properly, an attacker will be detected with high probability while achieving throughput arbitrarily close to optimal. Such properties reduce the incentive for the attacker to attack.

I. INTRODUCTION

In wireless ad hoc and sensor networks, paths between a source and destination are usually multihop, and data packets are relayed in several wireless hops from their source to their destination. This multihop nature makes the wireless networks subject to tampering attack: a compromised/misbehaving node can easily ruin data communications along the paths it is on by dropping or corrupting packets it should forward.

Watchdog mechanism proposed in [1] is a monitoring method used for ad hoc and sensor networks, and it is the base of many misbehavior detection algorithms and trust or reputation systems. The basic idea of watchdog is that watchdog node monitors whether its neighbor forwards the packets by overhearing. If the packet is not forwarded within a certain period or is forward but altered, the neighbor is regarded as misbehaving in this transaction. When the misbehaving rate surpasses certain threshold, the source is notified and subsequent packets will be forwarded along other routes.

The main challenge for most watchdog mechanisms is the unreliable wireless environment. Due to possible reasons such as channel fading, collision with other transmission, or interference, even when the source node and the attacker are both within communication range, the watchdog may not be able to overhear every transmission and therefore is unable to determine whether there is an attack.

To mitigate the misbehavior of the malicious nodes, a watchdog mechanism must achieve the following two goals: (1) A malicious node should be detected with high probability

if it attacks. (2) The throughput under the detection mechanism should be comparable to the throughput without detection if there is no attack. These two goals seem to have conflict in interest. On one hand, to improve the probability of detection, we need to introduce more redundancy. On the other hand, better throughput requires redundancy to be reduced.

In this paper, we show that both goals can be achieved simultaneously by introducing error detection block coding to the watchdog mechanism. This scheme is computationally simple, yet efficient. The watchdog only need to perform a compare operation. And by choosing the encoder properly, the probability of miss-detection can be made arbitrarily small while the throughput approaches optimal, even in the case when the attacker knows what encoder is being used and the watchdog can only overhear a fraction of the packets.

The remainder of the paper is organized as follows. Section II discusses related work. Section III proves any linear operation is inefficient in misbehavior detection. Section IV and V describe and analyse our watchdog scheme with error detection codes. Finally, Section VII concludes the paper.

II. RELATED WORKS

To ensure the reliability of packet delivery, trust for ad hoc and sensor networks has been investigated in a lot of literatures. The foundation of such dynamic trust system is the node behavior monitoring mechanism. The most frequently used one is the watchdog mechanism proposed in [1] and its variations.

The main idea of watchdog in [1] was overhearing. When a node sends a packet to its neighbor, it also cached it locally. Then the node listens to its neighbor's communication. If the neighbor does not forward the same packet to its next-hop node within a short period, it is regarded as misbehaving. By this way, a node can record the successful and failed forwarding history of its next-hop.

On the basis of watchdog, various misbehavior judging and handling mechanisms are proposed. [1] judges a node to be misbehaving when failure tally exceeds a certain threshold and it sends a packet backward to notify the source. Then the source would choose a new route free of misbehaving node with the aid of "pathrater".

[2] proposes to measure the next-hop's behavior with the local evaluation record which is defined as a 2-tuple: packet ratio and byte ratio, forwarded by the next-hop neighbor. Local

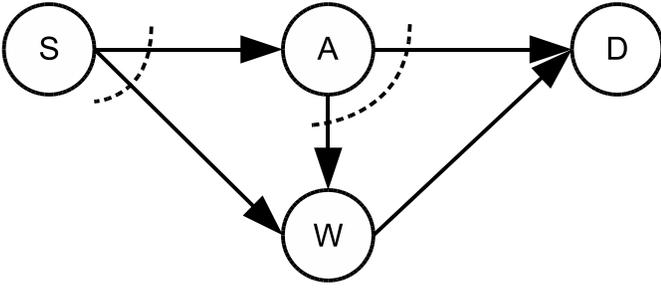


Fig. 1. Single Flow. Arrows (in or out) connected to the same node interfere with each other. The dash lines represent broadcast channels.

evaluation records are broadcast to all neighbors. The trust level of a node is the combination of its local observation and the broadcasted information. Trust level is inserted to the RREQ. Route is selected in the similar way to AODV [3]. Although many ad hoc trust or reputation systems [4], [5] and [6] adopt different trust level calculation mechanism, the basic processes are similar to [2], including monitoring, broadcasting local observation, combing the direct and indirect information into the final trust level.

Recently, the security issue in network coding systems has drawn much attention. Due to the *mixing* nature of network coding, such systems are subjects to a severe security threat, known as a *pollution attack*, where attackers inject corrupted packets into the network.

Several solutions to address pollution attacks in intra-flow coding systems use special-crafted digital signatures [7], [8], [9], [10] or hash functions [11], [12], which have homomorphic properties that allow intermediate nodes to verify the integrity of combined packets. Non-cryptographic solutions have also been proposed [13], [14]. [15] proposes two practical schemes to address pollution attacks against network coding in wireless mesh networks without requiring complex cryptographic functions and incur little overhead.

Most of the existing network coding scheme relies on random linear combination of data packets. And as we show in Section III, any linear operation cannot eliminate miss-detection even if all transmissions are reliable.

III. LIMITATION OF LINEAR CODING

In this section, we point out the limitation for linear coding in attack detection and show the advantage of non-linear coding. Let's consider the following example as in Fig.1. There are 4 nodes in this case: the source node S, destination node D, attacker A, and the watchdog node W. Transmissions are represented by arrows. Arrows (in or out) connected to the same node interfere with each other and cannot be schedule simultaneously. The dash lines represent broadcast channels.

Each packet consists of n symbols from the finite field \mathbb{F}_q . When S (A) sends a packets, it will be received by A and W (D and W). S wants to transmit data packets to D through A. We want any tampering by A to be detected by D. We assume all links are reliable, have the same transmission rate 1 symbol per unit time. We also assume an optimal centralized schedule is enforced. Under such assumptions, the watchdog W

is able to monitor every packet and send m checking symbols to D. The m checking symbols is a function of p and p' , vector representation of the original packet sent by S and the corresponding copy forwarded by A: $w = F(p, p')$. Under such assumptions the throughput is

$$T = \frac{n}{2n + m} \text{ (symbols/unit time)}. \quad (1)$$

For the case of linear coding, we assume F satisfies the following properties:

$$F(0, 0) = 0 \quad (2)$$

$$F(a, b) + F(c, d) = F(a + c, b + d) \quad (3)$$

$$F(\gamma a, 0) = \gamma F(a, 0) \quad (4)$$

$$F(0, \gamma a) = \gamma F(0, a). \quad (5)$$

Node D will miss an attacked packet if $F(p', p') = w$. Denote $p' = p + e$,

$$F(p', p') = F(p, p') \quad (6)$$

$$\Leftrightarrow F(p + e, p + e) = F(p, p + e) \quad (7)$$

$$\Leftrightarrow F(p, p) + F(e, e) = F(p, p) + F(0, e) \quad (8)$$

$$\Leftrightarrow F(e, e) = F(0, e) \quad (9)$$

$$\Leftrightarrow F(e, 0) = 0. \quad (10)$$

It is easy to show that $F(e, 0)$ is a linear function of e and can be expressed by a $m \times n$ matrix M in the finite field \mathbb{F}_q , and

$$F(e, 0) = Me. \quad (11)$$

If A chooses e from the null space of M, $Null(M)$, $F(e, 0)$ will be 0 and D will consider the packet safe. Suppose A has no knowledge of F , the best it can do is to pick a random e . Then the probability of miss an attack equals to the probability of picking e from the $q^{Rank(Null(M))} - 1$ non-zero vectors of $Null(M)$ out of $q^n - 1$ non-zero vectors in the n dimension space. Since $n - m \leq Rank(Null(M)) \leq n$, we have the following bounds of the probability of miss-detection for any linear coding scheme

$$\begin{aligned} \frac{q^{n-m} - 1}{2^n - 1} &\leq P_{miss} \leq \frac{q^n - 1}{2^n - 1} \\ \Leftrightarrow \frac{q^{n-m} - 1}{q^n - 1} &\leq P_{miss} \leq 1. \end{aligned} \quad (12)$$

So to achieve a target probability of miss-detection θ , W has to send at least $m \geq \lceil -\log_2 \theta \rceil$ checking symbols to D for every packet. On other hand, if we allow F to be nonlinear, only one symbol is enough to eliminate miss-detection completely. This can be easily done by setting $F(p, p') = \mathbb{1}_{\{p=p'\}}$, which equals to 1 if $p = p'$ and 0 otherwise.

Here we want to point out the same result also applies to linear network coding. The proof is similar by considering p as a *generation* of n coded packets and the watchdog sends m linear combinations of the packets it overhears to the receiver for verification.

IV. SINGLE FLOW CASE

Here we consider the same example in Section III. The watchdog W will compare packets that it overhears from both S and A, and will report an attack if they do not match. But we assume the watchdog W can detect tampering by A with probability q . In this case, W may not always be able to detect an attack. To enhance security, S encodes every k packets into a block of n coded packets with a (n,k) error detection code. We further assume the attacker knows what encoder is being used but does not know which packets W is able to overhear.

We assume MDS (maximum distance separable) codes are being used. With a (n,k) MDS code, an attack will always be detected as long as no more than $n - k$ packets are altered. As a result, A has to alter at least $n - k + 1$ packets in a block in order to avoid being detected by the decoder. And since the more packets A attacks the easier it will be caught by W, it is of A's interest to just attack the minimum number of packets per block: $n - k + 1$. In this case, it is easy to show that the probability of A not being caught is

$$P_{miss}(n, k, q) = (1 - q)^{n-k+1}. \quad (13)$$

We are interested in the highest coding rates we can achieve such that A has no incentive to attack. We construct a (n,k) encoder such that

$$k = n + 1 - \frac{f(n, q)}{q} \quad (14)$$

From Eq.13 we have

$$P_{miss}(n, k, q) \leq e^{-q(n-k+1)} = e^{-f(n, q)} \quad (15)$$

We can then choose the function $f(n, q)$ appropriately so that we can make P_{miss} arbitrarily small. For example, by making $f(n, q) = \beta \ln n$ for any positive constant β , we have

$$P_{miss}(n, k, q) \leq e^{-\beta \ln n} = n^{-\beta} \quad (16)$$

So we can reduce the incentive for A to attack by making the block longer. And the coding rate becomes

$$\begin{aligned} R &= \frac{k}{n} \\ &= \frac{n + 1 - \frac{\beta \ln n}{q}}{n} \\ &= 1 + \frac{1}{n} - \frac{\beta \ln n}{q n} \end{aligned} \quad (17)$$

Since the delay to verify a block equals to the time it takes to transmit n packets in the block, tradeoff between probability of miss-detection and n we plot in Figure 2 and Figure 3 is also the tradeoff between miss-detection and delay. We assume that for the n plotted in the figures, a suitable MDS (n,k) code exists for the block. We can see that by integrating a watchdog and error detection coding, we can reduce the incentive for the attacker to attack by allowing longer delay.

Notice that by making n large, the coding/decoding complexity increases. In the case complexity is a concern, the source can scramble coded packets of multiple (n, k) encoded

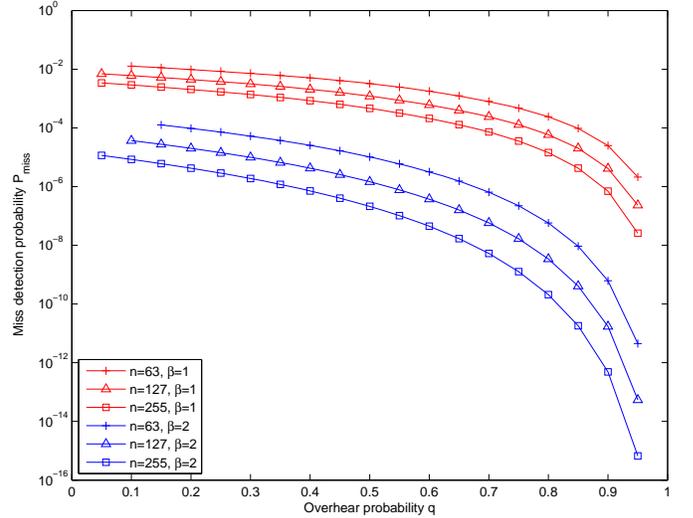


Fig. 2. Miss detection probability v.s. observe probability in the single flow example.

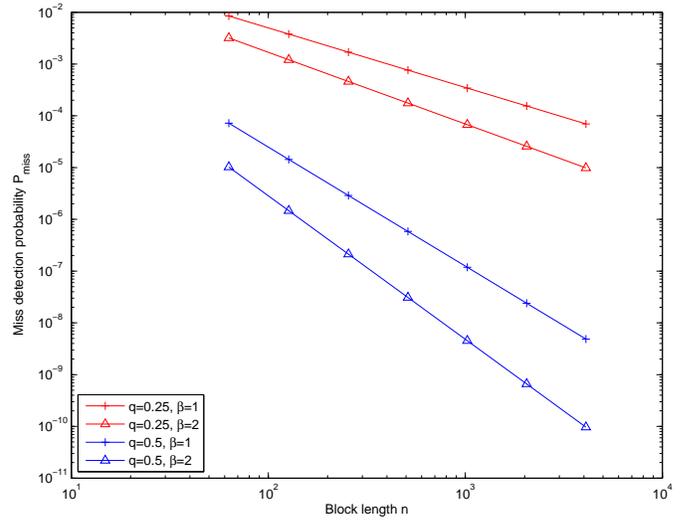


Fig. 3. Miss detection probability with $k = n + 1 - \frac{\beta \ln n}{q}$ in the single flow example.

blocks and transmit these packets in a random order. By doing so, the attacker will have to corrupt more packets in order to destroy a particular block, which makes it easier to be detected by the watchdog.

V. TWO FLOWS CASE

In the previous section, we assume the watchdog W can only compare a packet with probability q . Possible reasons for making this assumption are: a watchdog node may be intentionally turned off occasionally in order to save power, or interference from other nodes in the network makes the watchdog can observe only a fraction of the packets. In this section, we will look into the latter case. Since the level of interference is highly correlated to the traffic load in the system, we will mainly focus on the trade-off between throughput and security.

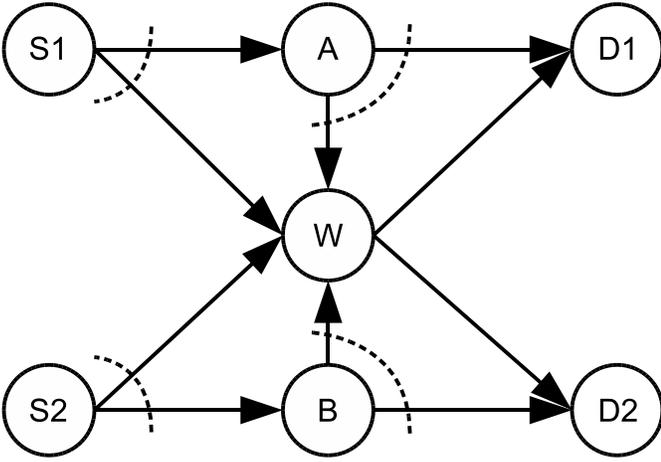


Fig. 4. Two Flows

Consider the following example. There are two flows in the system: S1-A-D1 and S2-B-D2. These flows are far away from each other so there is no inter-flow interference. But the watchdog W is sitting between the flows and can overhear transmissions on all the four links. So even though a transmission is successful along its path, it may collide with transmissions along the other flow at W. Suppose A is the attacker, we want to know the probability q in this case. For traffic pattern, we assume a slotted aloha with access probability α . To simplify the analysis, we further assume a node will access the channel by transmitting dummy packets when it has no data packet to send. Under these assumptions, we can compute the throughput and observe probability as

$$T = \alpha(1 - \alpha), \quad (18)$$

$$q = (1 - \alpha)^5. \quad (19)$$

The exponent in Eq.19 is 5 because given that the transmission from S1 to A is successful, W can overhear it if neither S2 nor B transmit which occurs with probability $(1 - \alpha)^2$. To compare this packet, W should overhear the transmission from A to D1 too, which happens with probability $(1 - \alpha)^3$ for S1, S2 and B to remain silent.

Similar to the one-flow example, we can make P_{miss} arbitrarily small by choosing

$$k = n + 1 - \frac{\beta \ln n}{(1 - \alpha)^5}. \quad (20)$$

And the effective throughput is

$$\begin{aligned} T_E &= TR \\ &= \alpha(1 - \alpha) \left(1 + \frac{1}{n}\right) - \frac{\alpha\beta \ln n}{(1 - \alpha)^4 n}. \end{aligned} \quad (21)$$

In Figure 5 and Figure 6 we plot the miss-detection probability and effective throughput when the error detection code is chosen according to Eq. 20. We only plot the result for $\alpha \leq 0.5$ because further increasing α will only reduce the throughput. We can see from Figure 5 the probability of miss-detection increases as the α increases and converges to roughly $n^{-\beta}$. Since the higher α is, the fewer packets the watchdog

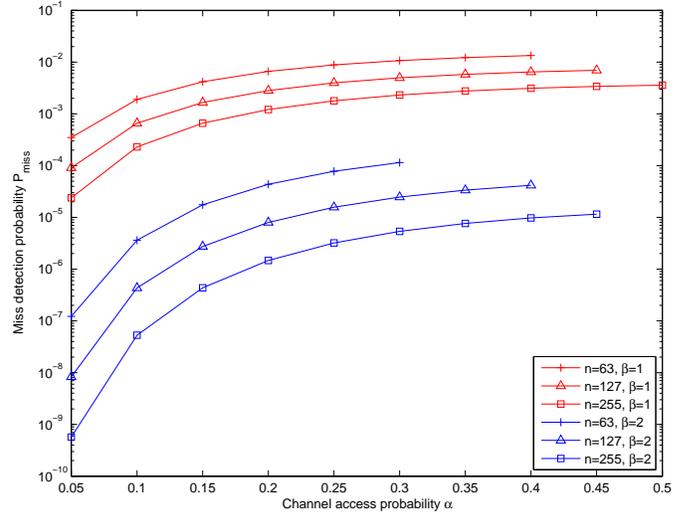


Fig. 5. Miss detection probability v.s. channel access probability with $k = n + 1 - \frac{\beta \ln n}{(1 - \alpha)^5}$ in the two flows example. Where the curves stop means no code is available.

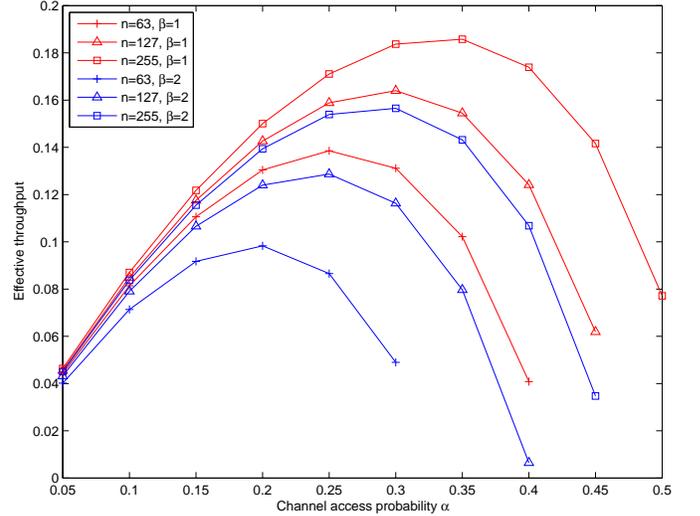


Fig. 6. Effective throughput v.s. channel access probability $k = n + 1 - \frac{\beta \ln n}{(1 - \alpha)^5}$ in the two flows example. Where the curves stop means no code is available.

can observe, the source has to sacrifice coding rate in order to maintain a certain probability of missing an attack as α increases. As it is shown in Figure 6, as α increases, the effective throughput increases up to a certain level then drops to zero as α gets larger.

We show the performance of some $(2^m - 1, 2^m - m - 1)$ Hamming codes in Figure 7 and Figure 8. In the case we cannot adapt the encoder to channel access probability, although there is no guarantee for miss-detection probability, a longer code always performs better in terms of both miss-detection probability and effective throughput. But such improvement comes with the cost of additional delay.

VI. DISCUSSION

In the previous sections, we have studied the case when the watchdog node is trustworthy. But in reality, it is also possible

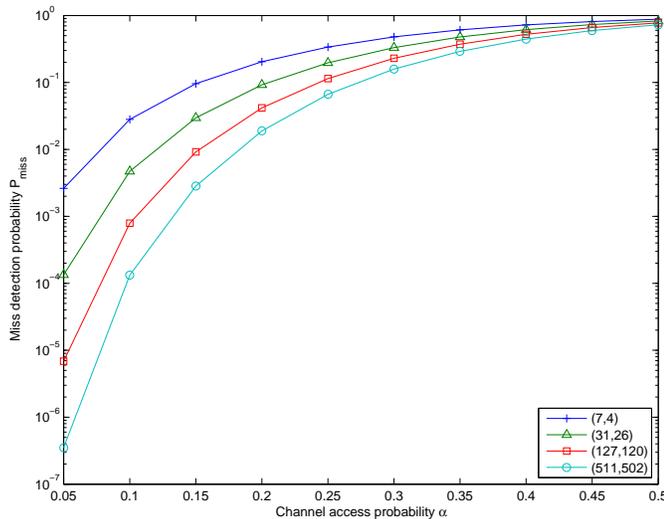


Fig. 7. Miss detection probability v.s. channel access probability for some Hamming codes.

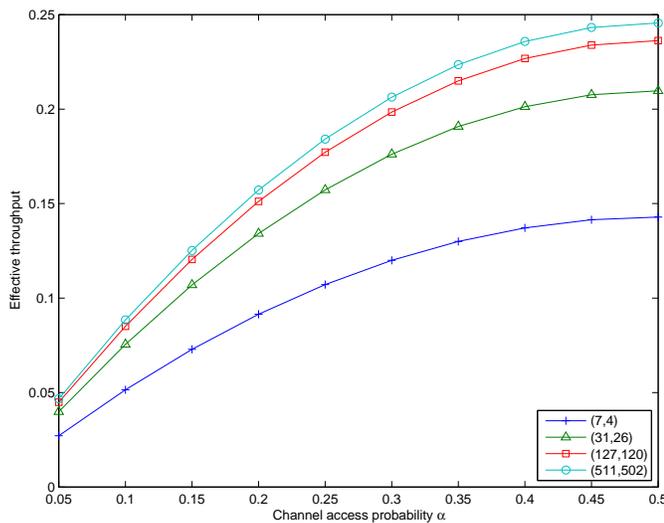


Fig. 8. Effective throughput v.s. channel access probability for some Hamming codes.

that the watchdog misbehaves. We admit that our scheme may fail detecting an attack both the watchdog and the forwarder can be malicious. In this case the relay node can alter the packets as much as possible without being detected as long as the faulty watchdog never declares an attack. However, in the case of single failure (at most one of the two nodes - forwarder or watchdog - is faulty), if the watchdog is faulty, the only way for it to attack the system is to accuse the relay node of attacking; and if the watchdog is well-behaving, it will declare an attack if and only if the relay node alters the packets. So under the assumption of single failure, we can be sure that either the watchdog or the relay is malicious. However, our scheme still cannot determine which node is misbehaving. To break the tie, the relay may have to be monitored by more than one watchdog and have a higher connectivity requirement. This is one of the potential directions, and we are currently working on it.

VII. CONCLUSION

In this work we study the problem of misbehavior detection in wireless networks. We first show that even if a watchdog can overhear all packet transmissions of a flow, any linear operation of the overheard packets can not eliminate miss-detection and is inefficient in terms of bandwidth. We propose a lightweight misbehavior detection scheme which integrates the idea of watchdogs and error detection coding. We show that even if the watchdog can only observe a fraction of packets, by choosing the encoder properly, an attacker will be detected with high probability while achieving throughput arbitrarily close to optimal.

REFERENCES

- [1] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2000, pp. 255–265.
- [2] T. Ghosh, N. Pissinou, and K. Makki, "Towards designing a trusted routing solution in mobile ad hoc networks," *Mob. Netw. Appl.*, vol. 10, no. 6, pp. 985–995, 2005.
- [3] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," *Mobile Computing Systems and Applications, 1999. Proceedings. WM-CSA '99. Second IEEE Workshop on*, pp. 90–100, Feb 1999.
- [4] S. Buchegger and J.-Y. Le Boudec, "Performance analysis of the confidant protocol," in *MobiHoc '02: Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*. New York, NY, USA: ACM, 2002, pp. 226–236.
- [5] C. Zouridaki, B. L. Mark, M. Hejmo, and R. K. Thomas, "A quantitative trust establishment framework for reliable data packet delivery in manets," in *SASN '05: Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*. New York, NY, USA: ACM, 2005, pp. 1–10.
- [6] S. Ganerwal and M. B. Srivastava, "Reputation-based framework for high integrity sensor networks," in *SASN '04: Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks*. New York, NY, USA: ACM, 2004, pp. 66–77.
- [7] D. C. Kamal, D. Charles, K. Jain, and K. Lauter, "Signatures for network coding," in *In Proceedings of the fortieth annual Conference on Information Sciences and Systems*, 2006.
- [8] Z. Yu, Y. Wei, B. Ramkumar, and Y. Guan, "An efficient signature-based scheme for securing network coding against pollution attacks," *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pp. 1409–1417, April 2008.
- [9] F. Zhao, T. Kalker, M. Mdard, and K. J. Han, "Signatures for content distribution with network coding," in *In Proc. of International Symposium on Information Theory (ISIT)*, 2007.
- [10] Q. Li, D.-M. Chiu, and J. Lui, "On the practical and security issues of batch content distribution via network coding," *Network Protocols, 2006. ICNP '06. Proceedings of the 2006 14th IEEE International Conference on*, pp. 158–167, Nov. 2006.
- [11] M. N. Krohn, "On-the-fly verification of rateless erasure codes for efficient content distribution," in *In Proceedings of the IEEE Symposium on Security and Privacy*, 2004, pp. 226–240.
- [12] C. Gkantsidis and P. Rodriguez Rodriguez, "Cooperative security for network coding file distribution," *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pp. 1–13, April 2006.
- [13] T. Ho, B. Leong, R. Koetter, M. Mdard, M. Effros, and D. Karger, "Byzantine modification detection in multicast networks using randomized network coding," 2004.
- [14] S. Jaggi, M. Langberg, S. Katti, T. Ho, D. Katabi, and M. Medard, "Resilient network coding in the presence of byzantine adversaries," *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pp. 616–624, May 2007.
- [15] J. Dong, R. Curtmola, and C. Nita-Rotaru, "Practical defenses against pollution attacks in intra-flow network coding for wireless mesh networks," in *WiSec '09: Proceedings of the second ACM conference on Wireless network security*. New York, NY, USA: ACM, 2009, pp. 111–122.