Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 205920, 16 pages
http://dx.doi.org/10.1155/2013/205920

Research Article

Detection and Prevention of Selective Forwarding-Based Denial-of-Service Attacks in WSNs

Youngho Cho and Gang Qu

Department of Electrical and Computer Engineering and Institute for Systems Research, University of Maryland, College Park, 20742, USA

Correspondence should be addressed to Gang Qu; gangqu@umd.edu

Received 2 May 2013; Accepted 18 June 2013

Academic Editor: S. Khan

Designing wireless sensor networks (WSNs) that can work reliably in the presence of inside packet drop attackers is very challenging. Current trust mechanisms and avoidance approaches are promising but have their limitations. Avoidance approaches transmit multiple copies of the packets to avoid attackers and cause high overhead. In trust mechanisms, each sensor monitors it neighbors, evaluates their trustworthiness, classifies them as either trustworthy or untrustworthy, and then discards untrustworthy sensors from the network. However, malicious insiders, which are legitimate members of the network and know exactly what their monitoring nodes know, can launch attacks carefully to avoid being detected and discarded from the network. In this paper, we first show that this is possible by introducing a selective forwarding-based denial-of-service (DoS) attack. We then propose an enhanced trust mechanism to detect such attackers and identify their victims. Furthermore, we design two attacker-aware protocols to reroute victim nodes' packets by avoiding the attackers. We conduct extensive OPNET simulations to validate our claims and demonstrate the advantages of our proposed approaches. Finally, as a complementary defensive method to our detection and avoidance approaches, we introduce a prevention routing algorithm that proactively prevents the attack and provide our preliminary results to evaluate its performance.

1. Introduction

In wireless sensor networks (WSNs), sensor nodes will generate data packets and send them to the base station (BS) in a multihop collaborative fashion due to their limited energy and transmission range. While being routed to the BS, data packets may be lost from collision, congestion, noise, or other network problems. The so-called *insider packet drop attacks* refer to a set of attacks where compromised nodes intentionally drop packets [1]. Such attackers disguise their malicious behavior behind the aforementioned natural packet loss phenomenon. This type of attack has become a serious security threat in WSNs [1–3]. A well-positioned malicious insider can be on the routing path of many sensor nodes and thus receive many data packets. It can simply drop them to cause damage to the network.

Selective forwarding attack, where the attacker drops only some packets and at some arbitrary time, is the most difficult insider packet drop attack to defend against [2]. Normally such an attacker seeks to achieve one of the following two goals. First, degrade the performance of the network in terms of packet loss rate. Second, prevent data collected by certain sensor nodes from reaching the BS. In the second case, the victim node will not be able to talk to the BS, and we name this attack *selective forwarding-based denial-of-service (DoS) attack.* Most reported studies on selective forwarding attacks focus on the detection of the attacker with the first goal [4–7]. As we will discuss later, these approaches are not effective against selective forwarding-based DoS attacks.

As a motivation for the importance of studying selective forwarding-based DoS attacks, we consider a WSN deployed in a territory for intruder detection. With the help of insiders that perform the selective forwarding-based DoS attack, an intruder will be able to enter the territory from the area monitored by victim nodes (to the selective forwarding-based DoS attacks) without being noticed by the BS. When the intruder can communicate with the inside attackers, they can launch the *synchronized insider-outsider colluding DoS attack* so the insider attackers can target different victims at different

times, and the intruder can explore the territory covered by the victim nodes only.

Trust mechanism has been proven as a promising approach to identify inside packet drop attackers [3, 8–12]. In such approach, each node will monitor its neighbor's packet forwarding behavior and use this observation to measure the trustworthiness of its neighbors. Once a neighbor's trust value falls below a predetermined threshold, the monitoring node will consider this neighbor as an inside attacker and eliminate it from the routing table.

Another conceptually different approach to defend against insider packet drop attacks is avoidance [2, 13], where multiple copies of the packets are sent to the BS through multiple disjoint paths. As long as there is a path that does not contain any attackers, the packets will be delivered to the BS successfully. However, this approach has a very high cost in terms of network traffic, transmission energy, and so forth [2, 7].

In this paper, we study the selective forwarding-based DoS attacks and propose effective detection and avoidance mechanisms as well as a prevention routing algorithm to defend against such attacks. Specifically,

(i) we first describe a simple selective forwarding-based DoS attack and show that the popular trust-based approaches (such as beta [14] and entropy [10] trust mechanisms) for inside attacker detection fail to detect such attack. We also analyze the potential damage this attack can cause to the network,

(ii) we then propose a *source-level trust evaluation scheme* to enhance the beta and entropy trust mechanisms for effective detection of the selective forwarding-based DoS attackers. Once the attacker is identified, we propose two *avoidance strategies* to reroute the victim's packets so they can reach the BS,

(iii) we validate our claims and evaluate the performance of our detection and avoidance mechanisms with extensive OPNET simulations,

(iv) as a complementary defensive mechanism to our detection and avoidance methods, we also introduce a prevention routing algorithm to proactively prevent the selective forwarding-based DoS attacks and show our preliminary results to evaluate its performance.

For simplicity, during the discussion of the threats and detection of insider packet drop attacks, we do not consider natural packet drops caused by network problems. However, our simulation settings include lossy networks and the natural packet drops due to that network problems will be reported.

The rest of this paper is organized as follows. Section 2 covers related work on insider packet drop attacks and the current countermeasures. Then in Section 3, we describe a selective forwarding-based DoS attack that none of the current defending approaches can detect to motivate our work. We propose our detection and avoidance approaches in Section 4 and evaluate their performance in the packet routing domain in Section 5. In Section 6, as a complementary defensive mechanism to our detection and avoidance methods, we introduce a prevention routing algorithm where an attacker has to choose between "not attacking" and "attacking and being caught." We conclude this paper in Section 7.

2. Related Work

Attackers to a network can be insiders, outsiders, or both. WSNs deployed for security applications (such as monitoring in the battlefield) are normally equipped with cryptography-based authentication and authorization mechanisms to prevent outside attackers from launching eavesdropping or packet modification. Thus outsider attacks are limited to direct physical damage of sensors or jamming the communication channel [15]. However, inside attackers have many advantages [2, 12, 15]. First, they are legitimate members of the network and will not be caught by authentication or authorization. Second, inside attackers can disrupt network operations by modifying packet information or dropping critical packets. Finally, inside attackers can collude with outside attackers to cause more severe damage to the network as we have described in the introduction [16].

Inside attackers can launch various types of attacks actively (such as modification, packet drop, or misrouting) or passively (such as eavesdropping). Among these, packet drop attacks not only can cause significant network performance degradation, but also cannot be prevented by authentication and authorization [2]. Below are three representative types of insider packet drop attacks [2, 8, 10].

Blackhole Attacks. The blackhole attacker drops all received packets. It will cause the most serious damage to the network among all types of packet drop attacks during the same amount of time. However, it can be easily captured by the monitoring neighbors as it consistently drops all their packets.

On-Off Attack. When attack is on, the attacker drops all received packets, then forwards all received packets when attack is off, and repeats this drop-forward pattern periodically. This attacker can appear suspicious to its neighbor during its attack period when it acts like blackhole attacks and can also be detected easily when the attack on period is long or the on-off pattern is discovered.

Selective Forwarding Attacks. As we described in the introduction, such attackers can either drop packets randomly or selectively. It is much more challenging to defend these attacks than blackhole and on-off attacks.

Current defending approaches against selective forwarding attacks are either *detection approach* or *avoidance approach*. The detection approaches will fail to detect the attacker and victims in our proposed selective forwarding-based DoS attack. The avoidance approaches will solve the problem, but it is very expensive and may not suit for WSN applications where each sensor has limited resource.

Most of the reported efforts focused on random selective forwarding attacks [4–7]. For example, Hai and Huh [4] presented a neighbor-based monitoring and detection

mechanism using two-hop neighbor knowledge where each exchanges its one-hop neighbors' packet forwarding behavior periodically. However, this approach introduces network overhead due to periodic information exchange between nodes and is vulnerable to false information provided by malicious neighbors. In the multihop acknowledgement scheme [7], each node in the forwarding path is responsible for detecting attackers. Specifically, some randomly chosen nodes (called ACK nodes) will report ACKs back to the source node (hop by hop) using the same but reversed routing path when they receive a packet. However, this approach has several problems. First, it is unclear how to locate the exact attacker. Second, their detection scheme depends on other nodes' observations, and thus their scheme is vulnerable to false accusation from malicious neighbors. The trust mechanisms with watchdog, as we have discussed earlier, solve these problems by monitoring whether the next node in the routing path forwards the packets or not [3, 9–11, 17]. Despite its many known limitations, the trust mechanism has been a promising solution to defend against insider packet drop attacks.

Instead of detecting the attackers, the avoidance approaches focus on how to deliver the packets successfully with the existence of the attackers. A popular way to achieve this is to use multipath routing paths [2, 13, 18, 19]. In [2], the authors pointed out that k disjoint multipath routing can completely defend against selective forwarding attacks with no more than $k-1$ compromised nodes. However, the multipath routing approach has a couple of drawbacks [7]. First, communication overhead significantly increases as the number of paths increases, and thus it may lead to increase collision and interference. As a result, the packet delivery performance of a routing can be dramatically degraded. Second, since this approach cannot catch and discard the attackers, this approach can be compromised if an adversary locates at least one attacker in each routing path. Similarly, a multiple data flow scheme using multiple disjoint topologies was introduced in [13]. In this scheme, a sending node sends its packets through one or more randomly chosen topologies among the preestablished multiple topologies to mitigate selective forwarding attacks.

3. A Selective Forwarding-Based DoS Attack

In this section, we first describe the current trust mechanisms and trust-based routing approaches to avoid inside attackers. Then we introduce a selective forwarding-based DoS attack and show that the current trust mechanisms fails to detect such attack.

3.1. Trust Mechanism. A trust mechanism defines a trust value (or trustworthiness) for each sensor node, and how each node measures the trustworthiness of its neighbors. It detects insider packet drop attacks in the following three stages.

Neighbor Behavior Monitoring. Each node monitors and records its neighbors' behavior such as packet forwarding. Watchdog [3] is a popular monitoring mechanism used in this stage. Each node M records all of its recently forwarded packets in a buffer. When M sends a packet to its neighbor node A, M monitors whether A forwards the packet toward the BS by overhearing A's packet transmission. Then, each overheard packet will be compared with the packet sent to A. When a match is found, M records that A has forwarded the packet and removes it from the buffer. If a packet remains in the buffer for a period longer than a predetermined time, the watchdog considers that A failed to forward the packet. In this paper, we use this watchdog mechanism in the OPNET network simulator [20].

Trust Measurement. Based on the data collected in the previous stage, a trust model will measure the trustworthiness of the node being monitored [8, 10, 14]. For example, when a node is observed to have forwarded the packet s times and dropped the packet f times, the beta trust model [14] will assign this node a trust value using the following formula:

$$T_{\text{Beta}} = \frac{s+1}{s+f+2}. \tag{1}$$

The entropy trust model [10] uses entropy function:

$$H(p) = -p\log_2 p - (1-p)\log_2(1-p), \tag{2}$$

where p is the trust value in beta trust model, and define the trust value by

$$T_{\text{Entr}} = \begin{cases} 1 - H(p), & \text{for } 0.5 \le p \le 1; \\ H(p) - 1, & \text{for } 0 \le p < 0.5. \end{cases} \tag{3}$$

Note that in (1), the trust value is between 0 and 1. But the trust value in (3) is between −1 and 1. To have a nonnegative trust value between 0 and 1, we define

$$T^*_{\text{Entr}} = \frac{1 + T_{\text{Entr}}}{2}. \tag{4}$$

Detection. By comparing the measured trust value with a predetermined threshold Θ_T, a node can decide whether its neighbor is trustworthy. If the neighbor's trust value is less than Θ_T, it will be considered as an inside attacker. Depending on the network's trust mechanism, the detection of inside packet drop attackers may or may not be broadcast to the rest of the nodes in the network. In this paper, we assume that the decision will not be broadcast for simplicity.

3.2. Trust-Based Routing. We use the popular greedy perimeter stateless routing (GPSR) [21] as an example to show how a trust mechanism can help to detect and avoid inside attackers.

Consider the WSN with 20 nodes shown in Figure 1. Node 3 relays the packets from nodes 4, 5, 9, 10, 15, and its own packets to node 2 which will then send them to the BS based on GPSR (depicted by solid lines with arrowhead). In a trust mechanism, node 3 will use its watchdog to monitor node 2.

When node 2 drops packets from node 3, node 3 will reevaluate the trust value of node 2. If the trust value falls below the threshold value, node 3 will treat node 2 as an

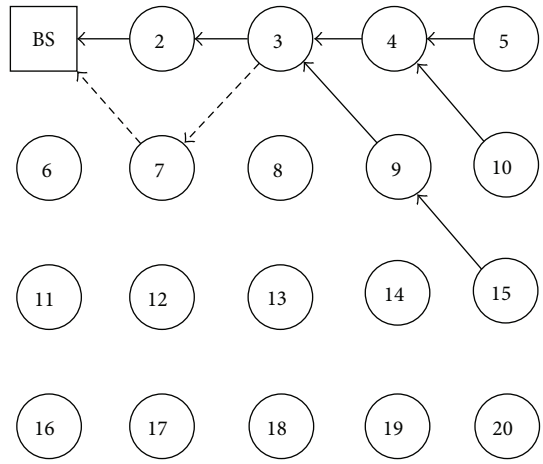


Figure 1: GPSR (solid lines) and trust-based GPSR (dotted lines).

inside attacker. A trust-based routing algorithm will then find a new routing path to avoid node 2. In this case, node 3 will forward the packets to node 7, hoping that node 7 will deliver the packets to the BS (the dotted lines with arrowheads in Figure 1).

Many researchers [9, 10, 17, 22–24] have shown that trust-based routing approaches can gracefully mitigate insider packet drop attacks by building trusted paths to the destination. Moreover, they showed that trust-based routing improves the packet's successful delivery under insider packet drop attacks over routing algorithms that do not consider trust. Clearly, the effectiveness of these trust-based routing algorithms is based on their underlying trust models. A good trust model will help the routing algorithm to quickly and accurately identify inside packet drop attackers and find alternate routes to avoid them.

3.3. A Selective Forwarding-Based DoS Attack and Its Analysis

3.3.1. Motivation. the current trust mechanisms and trust-based routing cannot detect all known insider packet drop attacks. For instance, an intelligent attacker who can keep its trust value above the threshold value Θ_T will not be detected. More weakness can be found in the literature such as [12]. Our proposed selective forwarding-based DoS attack comes from the following simple observation.

To attack victims and avoid being identified, the attacker node A will have to disguise itself by forwarding packets for some nodes. When a node M sends only its own packets to the attacker A and uses its watchdog to monitor A, apparently A cannot drop all the packets without being detected. However, if M also forwards packets from other nodes to A, then A may be able to drop all the packets from one or multiple victim nodes.

For example, in the WSN shown in Figure 1 where all the nodes generate packets with the same frequency and send them to the BS, node 2 can pick node 10 as its victim and drops all the packets from node 10. Therefore the BS can never hear messages from node 10 and hence comes the name "denial of service" for this attack. However, if node 2 forwards all the packets from nodes 3, 4, 5, 9, and 15 to the BS, when Beta trust model is used with $\Theta_T = 0.70$, the monitoring node 3 will hear node 2 forwarding $5/6 \approx 83\%$ of the packets and fails to identify node 2 as an attacker because node 2's trust value will be approximately 0.83, higher than the threshold $\Theta_T = 0.70$.

It is not hard to see that once an attacker positions itself on the routing path of many nodes, it can select multiple victim nodes and launch the denial of service attack without being noticed. This can easily cause a lot of damage to the network and so we need to find countermeasures to defeat such attack.

3.3.2. Protocol of the Attack. Steps shown in Algorithm 1 define the protocol for an inside attacker A to launch the selective forwarding-based DoS attacks against multiple victim nodes.

On each received packet (step 3), the attacker A first determines the direct sender of the message (node M) and original source node S that generates the packet (steps 4 and 5). If A has received packets from S before (i.e., S is not a new source node), A will either drop or forward the packet based on whether S is a victim or not (steps 6 and 7). If S is a new source node, A will update the number of nodes whose packets are routed to A through M by n_M++ (step 9). When n_M reaches a predetermined value, A will be able to select a new victim to launch the DoS attack (steps 10–14). We called this attack *selective forwarding-based DoS* because the attacker can selectively choose the victims and drop all the packets from the victims to mislead the BS to consider that the victim nodes are either out of service or disconnected.

3.3.3. Analysis of the Attack. For an inside attacker to launch the selective forwarding-based DoS attack against the victim nodes, the attacker needs to (i) be able to tell whether a received packet is from the victim nodes, and (ii) ensure that, after dropping all the packets from the victim nodes, the attacker will not be detected by the monitoring nodes.

We first show that assumption (i) is valid. In a geographic routing employed WSN, the receiver of a packet can obtain the source node (the node that creates the packet)

```
1  for each node M that forwards packets directly to A and monitors A with its watchdog, k_M = 0; n_M = 0;
2  while (both the network and node A are on) {
3  on the reception of a packet {
4    identify the node M that forwards the message;
5    identify the source node S that generates the packet;
6    if S is a victim node, drop the packet;
7    if S is a non-victim source node, forward the packet;
8    if S is a new source node {
9      n_M++;
10     if n_M = V[k_M] {
11       pick a new victim source node;
12       k_M++;
13       if S is the new victim node, drop the packet;
14       else forward the packet;
         }}}}
```

Algorithm 1: Steps for attacker A to launch selective forwarding-based DoS attacks against multiple victim nodes.

information from the packet because the receiver is a legitimate relay node that can access the packet's header where the source identification is stored [21, 25]. Even when the source node is protected by methods such as authorization, it is still possible for a malicious receiver to figure out the source node information by breaking the authorization mechanism or analyzing network traffics [26, 27].

Second, we will show that requirement (ii) can be satisfied. Because the inside attacker is a legitimate member of the WSN, it knows the trust model and the threshold value Θ_T used in the network. In a well-defined trust mechanism, when a node's packet drop rate increases, its trust value should not increase. A node will be considered as trustworthy if its trust value is above the threshold. Therefore, an attacker can evaluate its own trust value and drop a packet only when a drop will not bring its trust value below the threshold Θ_T. In our proposed protocol, the attacker selects a victim only when there are enough nonvictim source nodes to keep the attacker's trust value above Θ_T. This is guaranteed by the carefully determined array $V[\cdot]$ used in step 10 as we will explain next.

We define $V[j]$ as the minimum number of source nodes whose packets are routed to the attacker (A) through the same monitoring node (M) such that the attacker can drop packets from $j+1$ of these nodes without being detected by M. That is, $V[0]$ is the minimum number of nodes for attacker A to cover/disguise the first victim; $V[1]$ is the minimum number of nodes for A to attack two victims.

In the beta trust model, if attacker A attacks $(j+1)$ victims among $V[j]$ nodes and forward the packets for the other $(V[j] - (j+1))$ nodes, its trust value will be

$$\frac{V[j] - (j+1)}{V[j]} = 1 - \frac{j+1}{V[j]}. \tag{5}$$

To keep this trust value higher or equal to the trust threshold Θ_T, we can easily obtain the following:

$$V[j] = \left\lceil \frac{j+1}{1-\theta_T} \right\rceil. \tag{6}$$

Table 1: Values of $V[j]$ for three trust models.

$V[j]$	$V[0]$	$V[1]$	$V[2]$	$V[3]$	$V[4]$	$V[5]$
T_{Beta}	4	7	10	14	17	20
T_{Entr}	19	38	57	76	94	113
T^*_{Entr}	7	14	21	28	35	42

For the entropy trust models, there is no closed formula for $V[j]$. However, we can compute $V[j]$ numerically for any given Θ_T. Table 1 lists the values of $V[j]$ for the three different trust models where 0.70 is used as the trust threshold Θ_T.

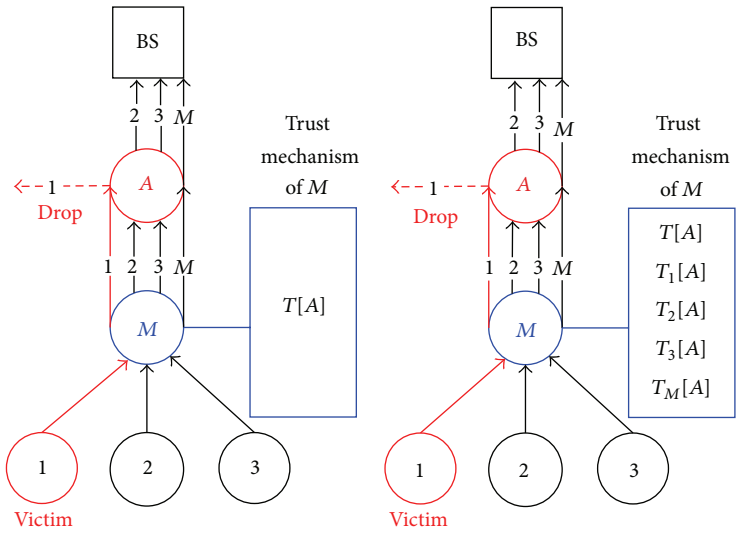
The small values of $V[0]$ indicate that the proposed selective forwarding-based DoS attack is a very serious threat. For the attacker (A) to launch the attack against a specific victim (V), it only requires the node (M) that forwards V's packets to A also forwards packets from 2 other nodes to A in the beta trust model ($V[0] = 4$: M, V, and 2 other nodes).

One can also see that $V[j]$ has a much larger value for the entropy trust models than the beta trust model. This is because earning a high trust value in entropy trust models (3) and (4) is much harder (i.e., a node must have very few packet drops) than earning a high trust value in the beta trust model (1).

4. The Proposed Defensive Mechanism

In this section, we propose and analyze our defensive mechanism, which is an enhancement of the beta and entropy trust mechanisms, against the above selective forwarding-based DoS attack. This defensive mechanism consists of two phrases: attacker detection and attacker-aware rerouting, which will be elaborated in Subsections 4.1 and 4.2 of this section, respectively. We analyze our approach and compare with existing methods in Subsection 4.3.

4.1. Source-Level Trust Evaluation and Attacker Detection. As depicted in Figure 2(a), in the existing trust mechanism [3, 9, 10, 14, 17, 22, 23, 28], a monitoring node M counts the



(a) Existing approach (b) Source-level trust evaluation

FIGURE 2: Existing trust evaluation approach [3, 9, 10, 14, 17, 22, 23, 28] and our proposed approach.

number of successes s and failures f that the next node A forwards packets from M. It then evaluates the trust value $T[A]$ of A based on s and f using the trust model adopted by the network. If $T[A] < \Theta_T$, M will consider A as an inside attacker. However, we have seen that this mechanism fails to detect intelligent attackers such as those launching the selective forwarding-based DoS attacks. For example, attacker A can drop all packets from node 1 but forwards packets from all other nonvictim nodes (in this case, nodes 2, 3, and M) to keep its trust value $T[A]$ high. When $T[A] \geq \Theta_T$, A's malicious attacking behavior will not be detected by M.

We can see that the current trust mechanism fails because the attacker can hide its malicious behavior behind its good behavior. As an attacker can identify the source node of a packet to launch the selective forwarding-based DoS attack, a monitoring node can also utilize the source node information to defend against such attack. This leads us to the following idea. If M uses separate counters to track not only A's overall packet forwarding behavior, but also how it delivers packets from each individual source node, then M will be able to tell whether A has launched the DoS attack against any node. This is shown in Figure 2(b) where M also evaluates A's trust value $T_i[A]$ for each source node i. We refer to this approach as *source-level trust evaluation*, and it can be easily integrated into the current 3-stage trust mechanism to improve its effectiveness of detecting inside attackers as follows.

Neighbor Behavior Monitoring. In addition to recording A's overall behavior s and f, for each packet that M overhears A is forwarding, M checks the source node information and updates a pair of separate counters, s_i and f_i, where i is the source node of the packet, to keep track the number of successes and failures for packets that A forwards from source node i, according to A's packet forwarding behavior to node i.

Trust Measurement. Based on the data collected in the first stage, M evaluates not only A's overall trust value $T[A]$ based on s and f, but also its source-level trust values $T_i[A]$ based on (s_i, f_i) to see how much M can trust A in forwarding packets from source node i. When the beta trust model is used, A's source-level trust value for source node i, $T_{\text{Beta},i}[A]$, can be calculated by using (1) as

$$T_{\text{Beta},i}[A] = \frac{s_i + 1}{s_i + f_i + 2}. \tag{7}$$

When the entropy trust model is used, A's source-level trust value for source node i, $T_{\text{Entr},i}[A]$, can be calculated by using (3) as

$$T_{\text{Entr},i}[A] = \begin{cases} 1 - H(p_i) & \text{for } 0.5 \leq p_i \leq 1; \\ H(p_i) - 1 & \text{for } 0 \leq p_i < 0.5, \end{cases} \tag{8}$$

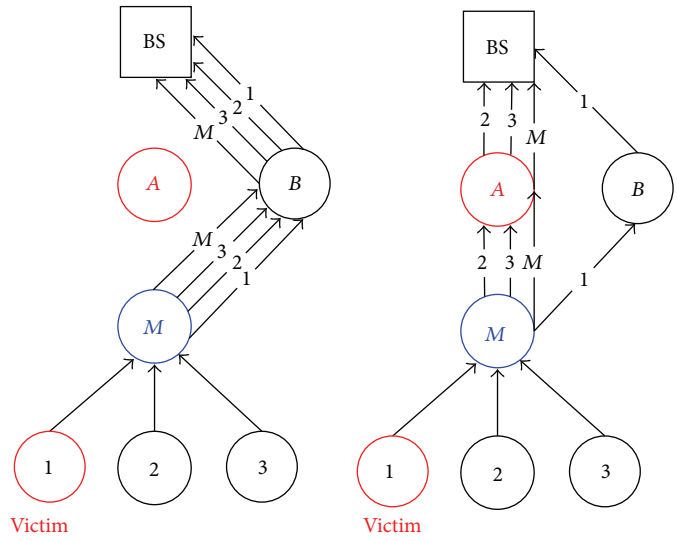
where $H(p_i) = -p_i \log_2 p_i - (1-p_i)\log_2(1-p_i)$ and $p_i = (s_i+1)/(s_i + f_i + 2)$. To have a nonnegative trust value between 0 and 1, we define

$$T^*_{\text{Entr},i}[A] = \frac{1 + T_{\text{Entr},i}[A]}{2}. \tag{9}$$

Detection. If any trust value $T_i[A]$ goes below the predetermined trust threshold Θ_T, M detects that A is a selective forwarding attacker against node i, the victim of such attack. When the overall trust value $T[A]$ of node A goes below the trust threshold Θ_T, A will be considered as an inside attacker just like the current trust mechanism will do.

Theorem 1. *The proposed source-level trust evaluation approach can successfully detect selective forwarding-based DoS attacks against any source node.*

Proof. By the definition of the selective forwarding-based DoS attack, if A launches attack against node i, it will behave



(a) Complete avoidance (CA) (b) Selective avoidance (SA)

FIGURE 3: Two avoidance strategies to reroute the victim's packets to BS.

like a blackhole attacker and drop all packets originated from node i. Hence, after the attack is launched, s_i will remain unchanged and f_i will increase by one whenever a packet from node i is dropped by attacker A. When node i generates sufficient number of packets, the packet drop rate $\alpha = f_i/(s_i + f_i)$ will increase and can be arbitrarily close to 1. This means that A's trust value with respect to node i, $T_i[A]$, will approach to the minimum trust value, which will be way below the trust threshold Θ_T. So the monitoring node M will be able to identify this DoS attack and its victim.

Formally, let $n_i = s_i + f_i$ be the total packets generated by a victim node I; this theorem is based on the following fact:

$$\lim_{n_i \to \infty} T_i[A](\alpha) = \lim_{n_i \to \infty} T_i[A]\left(\frac{f_i}{(s_i + f_i)}\right) = T_i[A](1) \ll \theta_T. \quad (10)$$

Because all the (s_i, f_i) pairs are kept independently, the selective forwarding-based DoS attack against any other source nodes can also be detected, depending on how fast the victim nodes generate packets. □

Our approach requires the number of delivery successes and failures for packets from each source node. This will introduce storage overhead. Fortunately, such overhead is negligible. Even in the case when a node is receiving packets from 100 different source nodes and wants to track the status of the last 1 million packets from each node, the memory requirements will only be 0.25 KB ($= 100 \times \log_2 2^{20}$ bits/8). This overhead is low for current sensors such as TelosB (10 KB RAM, 48 KB Flash, and 1 MB EEPROM) and Mica2/MicaZ (4 KB RAM, 128 KB Flash, and 512 KB EEPROM) [29].

4.2. Attacker-Aware Avoidance Routing Strategies. Once the attacker and a victim of the selective forwarding-based DoS attack are detected, approaches to reroute the victim's packet to the BS should be developed. In this section, we propose two attacker-aware rerouting algorithms, which we refer to as avoidance strategies.

When the value of a $T_i[A]$ becomes less than the network's trust threshold Θ_T, the monitoring node M will conclude that A is an inside attacker attacking node i. To avoid further damage that A may make to the network, M can use a *complete avoidance* (CA) strategy to reroute all the packets to another trustworthy neighbor node (such as B shown in Figure 3(a)). This ensures that all the packets received by M, not only those from node i, will avoid the attacker A. However, this strategy will increase the traffic on node B and may also introduce other routing overhead. For example, if node A was the best choice in an energy-efficient routing algorithm, rerouting all the packets to node B instead of A will cause increase in energy consumption. Furthermore, if A targets multiple victims, this strategy will help all of the victims to avoid the attacker A, but it can only identify the first victim. Finally, if M mistakenly claims the first victim, node A will be treated as an attacker. This will increase the false alarm rate in finding inside attackers.

In light of the fact that a selective forwarding-based DoS attacker (node A in this case) has targeted victims, the *selective avoidance* (SA) strategy will only reroute the discovered victim's packets to avoid the attacker A and keep the other packets running through node A (see Figure 3(b)). The monitoring node M will continue updating the trust values ($T_i[A]$) for all nodes except those discovered victims. So even when the attacker targets multiple victims, the SA strategy can discover all of them and help them avoid the attacker. This strategy will effectively solve CA's resource overhead problem. Its drawback is that it will take time for each of the victims to be identified, and the attacker can still drop packets from the victims and do damage to the network until all the victims are discovered.

Table 2: Comparison of the complete avoidance strategy (CA) and the selective avoidance strategy (SA).

	CA	SA
Reroute victim's packets	Yes	Yes
Reroute non-victim's packets	Yes	No
Time to reroute all victims' packets	Short	Long
Discover multiple victims	No	Yes
Probability of false alarm on attacker	Large	Small
Impact on the original routing solution	Large	Small

We summarize the features of the two proposed attacker-aware rerouting algorithms in Table 2.

4.3. Analysis of the Proposed Defensive Mechanism. The proposed defensive mechanism follows the 2-phase detection-avoidance framework. In the first phase, the source-level trust evaluation approach will detect victims of the selective forwarding-based DoS attack. In the second phase, the attacker-aware rerouting strategy will find a different path to deliver victim's packets to the BS.

4.3.1. Comparison with the Existing Trust Mechanisms. Our source-level trust evaluation is an enhancement of the existing trust-based mechanisms for inside attacker detection [10, 14]. The difference is that existing approaches do not consider the packet forwarding behavior of the receiving node (the node being monitored) for each individual source node. Therefore, it can detect whether the node is an inside packet drop attacker, but it will fail to detect the proposed selective forwarding-based DoS attack. In our proposed method, the monitoring node will evaluate the trust value with respect to each source node. As stated in Theorem 1, this enhancement enables us to identify not only the attacker, but also all the victims. The cost of our approach, compared with existing mechanisms, is the storage requirement to keep the delivery information for each source node, which we have analyzed after the proof of Theorem 1.

Now we compare the false alarm rate (FAR) of our approach with existing mechanisms. FAR measures how likely a good node will be tagged as an inside attacker. Let FAR, FAR_{CA}, and FAR_{SA} be the FAR of the existing detection approach, our approach with CA, and our approach with SA, respectively. We have the following.

Theorem 2. $\text{FAR}_{\text{CA}} \geq \text{FAR} \geq \text{FAR}_{\text{SA}}$.

Proof. Recall that the trust value $T[A]$ in the existing trust mechanism is defined based on the packet drop rate, which is the ratio of the total failures (f) over the total number of packets ($s + f$). A false alarm occurs when a good node's trust value $T[A]$ becomes smaller than the trust threshold Θ_T. In our approach, the monitoring node M also updates $T_i[A]$, the trust value with respect to source node i, which is determined by the drop rate of packets from node i or the pair of (s_i, f_i).

When we use CA strategy in the second phase, the first detected victim node j is the one that has the largest packet drop rate that results in the smallest $T_i[A]$ among all the node i's that send their packets to A through the same monitoring node M. That is, $T_j[A] = \min\{T_i[A]\}$. For the same set of node i's, we have $s = \sum s_i$ and $f = \sum f_i$. Clearly, $T[A] \geq T_j[A]$. Therefore, when the existing detection mechanism claims (regardless of the correctness of the claim) node A as an attacker (i.e., $T[A] < \Theta_T$), our approach should have already identified the first victim j of A's DoS attack because

$$T_j[A] \leq T[A] < \Theta_T. \tag{11}$$

However, when our approach claims an attacker, $T_j[A] < \Theta_T$, it is not necessarily true that $T[A] < \Theta_T$. A false alarm is an incorrect claim. So $\text{FAR}_{\text{CA}} \geq \text{FAR}$.

On the other hand, when SA strategy is applied, our approach will identify the DoS victims one by one and reroute the packets from these victims to nodes other than the attacker A. Note that victim nodes always have large packet drop rate; when their packets are rerouted, the trust value of A evaluated by our approach will be higher than that in the existing approach. This is because the existing approach will count the (s_i, f_i) pairs from these victims in $s = \sum s_i$, $f = \sum f_i$ in the calculation of $T[A]$. So when the same Θ_T is used, $T[A] < \Theta_T$ will always first happen in the existing approach before it happens in our approach with SA strategy, that is, $\text{FAR} \geq \text{FAR}_{\text{SA}}$. □

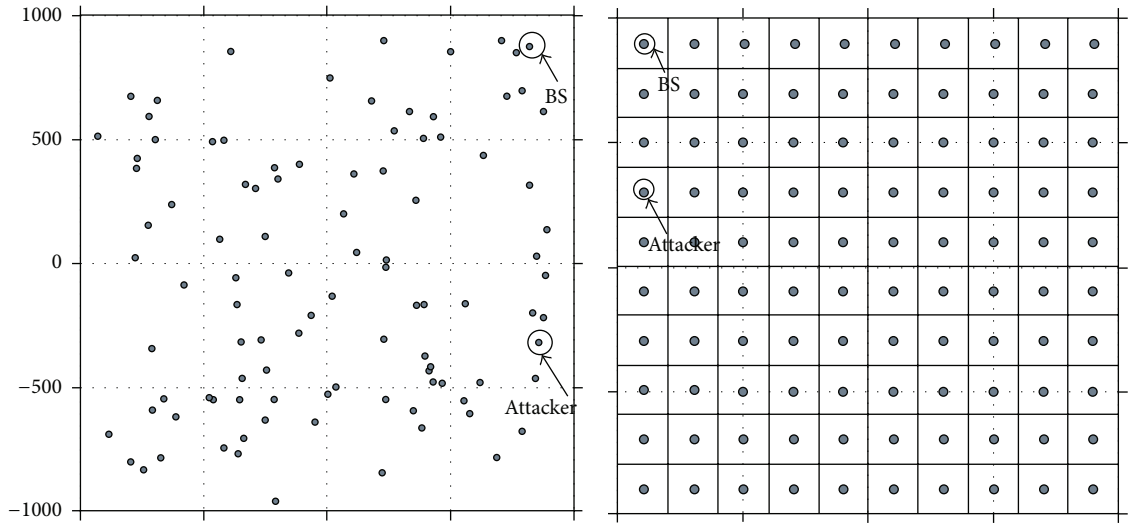
4.3.2. Comparison with the Avoidance Approaches. As we have mentioned in the introduction, the idea behind current avoidance approaches is to send packets from multiple disjoint paths in order to avoid inside packet drop attackers [2, 13, 18, 19]. These approaches cannot and are not intended to detect the attackers. We have also discussed in Section 2 that the overhead of such avoidance approaches can be prohibitively high. For example, when each packet is sent through multiple different paths, the transmission energy, the network traffic, and collision will all increase dramatically.

Despite the same name, the avoidance strategy in the second phase of our defensive mechanism is conceptually different from the above avoidance approaches. In our approach, the avoidance strategy is applied after both the victims and the attacker in the insider packet drop attack have been identified. Therefore we can efficiently find a path that does not involve the attacker to deliver victim's packets to the BS. Although the new path may not be as good as the initial path (where the attacker sits on) in terms of energy, delay, or channel quality, neither CA nor SA uses multiple paths. Hence, the large overhead problem in the conventional avoidance approaches does not exist in our defensive mechanism.

5. Simulation and Results Analysis

5.1. Simulation Goals, Setups, and Evaluation Metrics. There are two main goals of the simulation: validating that the current trust mechanisms fail to detect the proposed DoS attack and evaluating the performance of our defensive approach.

The parameters in Table 3 are used in our simulations. We conduct simulations with the commercial network simulator



(a) Random topology (b) Grid topology

Figure 4: Two WSN topologies in our simulations.

Table 3: OPNET simulation setup parameters for the validation of our detection mechanisms.

Parameters	Setting
General	
Terrain dimension	2 km × 2 km
Number of nodes	100
Topology	Random/grid
Max. simulation time	30 mins for single attacker; 40 mins for multiple attackers
Base routing algorithm	GRP
Max. retransmissions	7 (OPNET default)
Data packet generation	
Start time–stop time	100 seconds–end of simulation
Destination	Base station
Packet arrival interval	Every 10 second
Packet size	1,024 bits
Trust model	
Type	Beta/entropy
Initial trust value	0.99
Trust threshold (Θ_T)	0.7
Attack model	
Number of attackers	Single attacker/multiple attackers (=2)
Attack type	Selective forwarding-based DoS attack

OPNET Wireless Modeler v.17.1. 100 sensors are deployed in a 2 km × 2 km area randomly in one setting (Figure 4(a)) and in a 10 × 10 grid in another setting (Figure 4(b)). Each node except the BS generates packets randomly in each 10-second period. The packets are sent to the BS. We use some of the default settings in OPNET such as 1024 bit data packet and geographic routing protocol (GRP) with a maximum of 7 retransmissions before a packet is dropped. We set each node's initial trust value to be 0.99. We consider the cases of both single and multiple selective forwarding-based DoS attackers. The simulation time is set to be 30 minutes in the case of single attacker and 40 minutes for multiple attackers. We simulate the attacker(s) launch the proposed selective forwarding-based DoS attack to various numbers of victims. Both beta and entropy trust models (defined in (1) and (4)) as well as our enhanced trust mechanism (defined in (7) and (9)) with two avoidance strategies (CA and SA) are implemented in the OPNET Modeler for comparison purposes.

The main performance evaluation metrics are as follows.

(1) *Avoidance completion time (ACT)*: this is the time when all the victims have been rerouted to avoid the attacker.

(2) *False alarm rate (FAR)*: as discussed in the previous section, this is the probability that a good node is being considered as a selective forwarding-based DoS attacker.

(3) *Energy per packet (EPP)*: this is the average energy consumption to deliver a data packet, regardless of whether the packet reaches the BS or not. EPP is obtained by the total energy consumed for data packet transmissions divided by the total number of data packets generated by all source nodes.

5.2. Simulation Results and Analysis of Single Attacker. ACT is the most important metric as it indicates the ability of each approach in identifying the attacker and rerouting the victim's packets.

The simulation results on ACT in Table 4 reveal the following.

Beta or Entropy Trust Models Alone Fail to Detect the Attacker. In the grid topology, there are 21 source nodes that send

Table 4: Avoidance completion time (in seconds) for the Beta and Entropy trust model without any avoidance strategy (Pure), with the complete avoidance strategy (CA), and with the selective avoidance strategy (SA) when the attacker targets J victim nodes.

J	Beta trust model			Entropy trust model		
	Pure	CA	SA	Pure	CA	SA
			Grid topology			
1	Fail	542.5	542.5	Fail	269.0	269.0
2	Fail	539.0	551.5	Fail	266.0	279.5
3	Fail	538.5	552.0	Fail	268.0	278.5
4	Fail	538.0	553.5	255.5	255.5	255.5
5	Fail	543.0	562.5	184.5	184.5	184.5
6	Fail	541.5	552.5	152.5	152.5	152.5
7	581.0	541.5	561.0	151.5	151.5	151.5
			Random topology			
1	Fail	548.3	548.3	Fail	277.2	277.2
2	Fail	546.9	604.6	Fail	289.0	300.9
3	Fail	553.2	644.8	360.9	278.6	329.7
4	Fail	559.0	602.0	204.7	204.7	204.7
5	802.5	549.5	591.3	162.4	162.4	162.4
6	352.6	352.6	352.6	153.2	153.2	153.2

packets to the monitoring node and then to the attacker. From Table 1, when the selective forwarding-based DoS attacker targets 6 victims or less, the beta trust model will not detect it; when it targets 3 victims or less, the entropy trust model cannot detect it. The results in Table 4 confirm this. This is also true for the random topology where the monitoring node forwards packets from 16 source nodes (including itself) to the attacker.

Our Defensive Mechanisms Successfully Detect the Victims. Even when the attacker targets only one victim (the case of $J = 1$), our defensive mechanism can help both the beta trust model and the entropy trust model to identify the victim node. The entropy trust model is quicker because a dropped packet will cause more reduction in the trust value in the entropy trust model. It also takes more time for the SA strategy because it finds victims one by one.

Optimality of the Proposed Selective Forwarding-Based DoS Attack. We already discussed in the first item that our proposed selective forwarding-based DoS attack cannot be detected by the current trust model. Table 4 also shows that if the attacker becomes aggressive and targets more victims than the $V[j]$ values in Table 1 allow, then they will be detected by both the beta trust model and the entropy trust model.

FAR measures the likelihood an approach will mistakenly treat an honest node as attacker. In the grid topology, there are very few collisions and there is no false alarm. The FAR values for different approaches in the random topology are shown in Table 5. This result confirms the claim of $\text{FAR}_{\text{CA}} \geq \text{FAR} \geq \text{FAR}_{\text{SA}}$ we made in Theorem 2.

Finally, we report EPP. From Table 6, we can see that our proposed enhancement incurs very little energy overhead.

Table 5: False alarm rate in the random topology.

J	Beta trust model			Entropy trust model		
	Pure	CA	SA	Pure	CA	SA
1	0.010	0.036	0.010	0.048	0.069	0.048
2	0.010	0.034	0.010	0.045	0.067	0.044
3	0.011	0.033	0.010	0.065	0.065	0.046
4	0.011	0.031	0.010	0.064	0.064	0.064
5	0.031	0.035	0.010	0.061	0.061	0.061
6	0.028	0.028	0.028	0.061	0.061	0.061

Table 6: Energy per packet (mJ) for the Beta and Entropy trust model without any avoidance strategy (Pure), with the complete avoidance strategy (CA), and with the selective avoidance strategy (SA) when the attacker targets J victim nodes.

J	Beta trust model			Entropy trust model		
	Pure	Overhead (%)		Pure	Overhead (%)	
		CA	SA		CA	SA
			Grid topology			
1	37.53	3.65	0.40	37.53	4.42	0.51
2	37.42	3.87	0.77	37.42	4.73	0.99
3	37.27	4.19	1.23	37.27	5.12	1.50
4	37.14	4.52	1.67	39.19	0	0
5	37.03	4.70	2.05	39.25	0	0
6	36.89	5.04	2.49	39.32	0	0
7	38.65	0.13	−2.07	39.3	0	0
			Random topology			
1	78.14	1.10	0.44	78.40	1.38	0.54
2	77.56	1.55	0.98	77.74	2.14	1.40
3	76.96	2.27	1.68	79.23	0.06	−0.67
4	76.42	3.01	2.30	79.55	0	0
5	78.08	0.54	0.09	79.70	0	0
6	79.01	0	0	79.51	0	0

In the avoidance approach where multiple paths are used, for a single attacker, two disjoint paths will guarantee the successful avoidance of the attacker. However, the energy consumption will be doubled. From energy perspective, our approach is much better than the current avoidance approach.

In a couple of cases, when SA strategy is used, there is actually a small amount of energy savings. This is possible because the original geographical routing protocol does not guarantee energy efficiency. Moreover, as we have analyzed, SA strategy uses less energy than CA strategy because in SA strategy, only packets from detected victims will be rerouted.

5.3. Simulation of Multiple Attackers. For simplicity, we report the case of two attackers. When the two attackers are far away from each other, launching attacks to victim nodes independently, the result for each attack is almost identical to the single attacker case. Here we discuss the more interesting case when the two attackers are physically close to each other, for example, when the node to the right of the attacker in

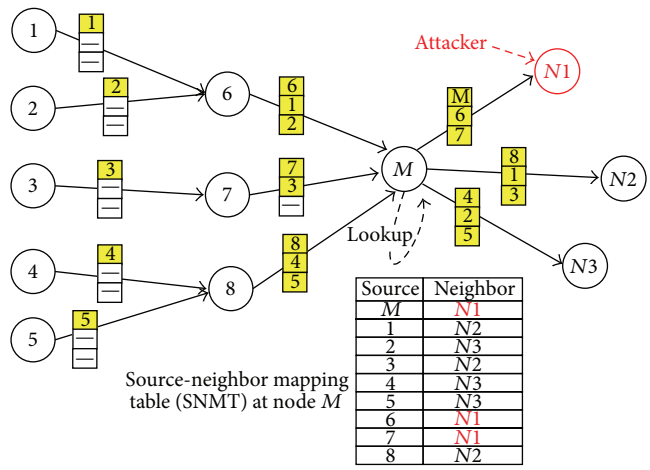


FIGURE 5: Our prevention routing algorithm against a selective forwarding-based DoS attacker ($N1$) when the beta trust model is used with $\Theta_T = 0.7$.

TABLE 7: Avoidance completion time (seconds) in the case of multiple attackers in the grid topology.

J	Beta trust model			Entropy trust model		
	Pure	CA	SA	Pure	CA	SA
1	Fail	1,896.6	1,895.3	Fail	805.3	807.3
2	Fail	1,876.6	1,447.3	Fail	780.6	632.0
3	Fail	1,862.6	1,306.6	Fail	772.6	574.6
4	Fail	1,861.3	1,230.0	8,750	742.6	624.6
5	Fail	1,869.3	1,189.3	617.3	563.3	475.3
6	Fail	1,866.6	1,158.0	372.0	372.0	372.0
7	Fail	1,864.0	1,137.3	298.0	298.0	298.0
8	Fail	1,374.0	999.3	248.6	248.6	248.6
9	1,438.3	1,092.6	835.3	215.3	215.3	215.3
10	740.0	740.0	740.0	196.6	196.6	196.6

Figure 4(b) is also an attacker and they both target the same set of victims.

As one can imagine, when a victim node is identified, either the CA or the SA strategy will try to reroute packets to avoid the attacker. However, because the attacker's neighbor is also an attacker, if the monitoring node happens to choose the second attacker to forward packets to, both ACT and EPP will increase. In particular, the ACT will be around doubled because it will take about the same amount of time for the monitoring node to recognize the second attacker and reroute again. We now study the simulation results below.

First, we see that the two attackers together can target more victims without being detected. For example, in Table 4, we know that a single attacker will be detected by the beta trust model if it attempts to attack 7 or more victims. However, Table 7 shows that the beta trust model can find the two attackers only when they are trying to attack 9 or more victims, which apparently indicates the improvement of attacking power.

Second, we see that the ACT is about tripled, instead of doubled, of the ACT in the single attack model. This is a little unexpected. However, the topology of the network and the position of the attackers are the main reason for this. In our case, when the monitoring node finds the second attacker, it will reroute the packets to a new node. The new node happens to forward the packets to the second attacker again; thus, it will take again time for the new node to identify the second attacker. This results in the ACT in the 2-adjacent attackers case is about three times of the ACT for single attacker.

6. Prevention Routing Algorithm

6.1. Motivation and Key Idea. As we have discussed earlier, when an inside attacker relays packets for many sensor nodes in the network, it can pick one or more victims to launch the selective forwarding-based DoS attack. This is because it can hide its malicious behavior by forwarding packets from other nodes and maintaining a high trust value. If an attacker is on the routing path of only one or two nodes and it attacks a victim, the chance that the attacker will be detected quickly is high. In such situation, the attacker may not take the risk to launch any attack. Based on this observation, we propose a prevention routing algorithm where an attacker has to choose between "not attacking" and "attacking and being caught." This is complementary to the detection and avoidance approach we described earlier. They can be used together as a more effective defensive mechanism.

The key idea of our prevention method is to limit the number of source nodes (N_{SMAX}) from which a node receives packets through the same monitoring node. As discussed in Section 3, if the attacker receives data packets from at least $V[0]$ source nodes from a monitoring node, it can launch the selective forwarding-based DoS attack against one of the source nodes without being detected by the monitoring node. Therefore, in our prevention method, we require each monitoring node forwards packets from at most $V[0] - 1$ source nodes; that is, $N_{\text{SMAX}} = V[0] - 1$. This will prevent the attacker from launching the selective forwarding-based DoS attack. If the attacker still launches the attack, it will be detected by the monitoring node.

Figure 5 shows how our prevention method successfully defends against a selective forwarding-based DoS attacker $N1$. Consider that the beta trust model with the trust

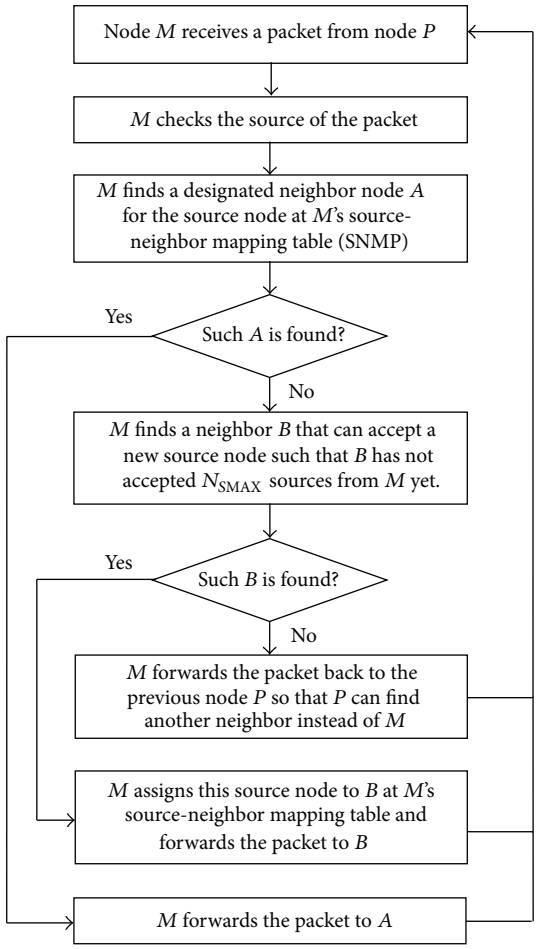


Figure 6: The flow chart of a trust-based routing algorithm with our prevention method to prevent the selective forwarding-based DoS attack.

threshold $\Theta_T = 0.7$ is used in a WSN. Node M receives packets from 8 nodes (nodes from 1 to 8). Although node $N1$ is M's best choice to relay the packets, our prevention routing algorithm will limit M to forward packets from only 3 nodes (nodes M, 6, and 7 in this case) to $N1$. Packets from the other 6 nodes will be forwarded to nodes $N1$ and $N2$, three each. That is, $N_{\text{SMAX}} = 3$.

We know that $V[0] = 4$ in this case from Table 1. As a result, the attacker $N1$ cannot launch the selective forwarding-based DoS attack against any of the three source nodes (M, 6, and 7) without being detected by M. If $N1$ starts attacking any of the three source nodes, $N1$'s trust value evaluated by M will be 0.67 (=2/3) and thus $N1$ will be caught by M because $N1$'s trust value is less than Θ_T (=0.7).

6.2. Proposed Prevention Routing Algorithm. Our prevention method can be easily integrated into any existing trust-based routing algorithm. Figure 6 shows the flow chart of a trust-based routing algorithm with our prevention method. Each time node M wants to forward a data packet toward the BS (regardless of its own packet or packets it receives from other nodes), M first checks the source node of the data packet and then finds a neighbor node A at M's *source-neighbor mapping table* (SNMT). SNMT is a look-up table that tells M which of M's neighbors will receive a certain source node's data packet to forward the packet toward the BS. If such node A is found at the SNMT for the source node, M will forward the packet to A. Otherwise, M will find a new neighbor node B such that the number of source nodes assigned to B is less than N_{SMAX}. If there are multiple neighbors satisfying such condition, the next hop selection algorithm of a base routing algorithm such as GPSR will choose the best one among them. If such node B is found, M registers B to its SNMT for the source node and then forwards the data packet to B. If M cannot find any neighbor satisfying such condition, M forwards the source's data packet back to the previous node P so that P can find other neighbor instead of M.

We explain how a relay node M assigns source nodes to its neighbor nodes as shown in Figure 5. Assuming that every source's data packet is equally important, we use the (First Come First Serve) FCFS manner for this source-neighbor assignment process. For example, in Figure 5, assume that M received the first data packets of its eight source nodes in the following order: M, 6, 7, 8, 1, 3, 4, 2, and 5. Then, M assigns firstly arrived three source nodes (M, 6, 7) to its best neighbor $N1$ chosen by its base routing algorithm. The next three source nodes (8, 1, and 3) and the remaining three source nodes (4, 2, and 5) are assigned to M's next best neighbors

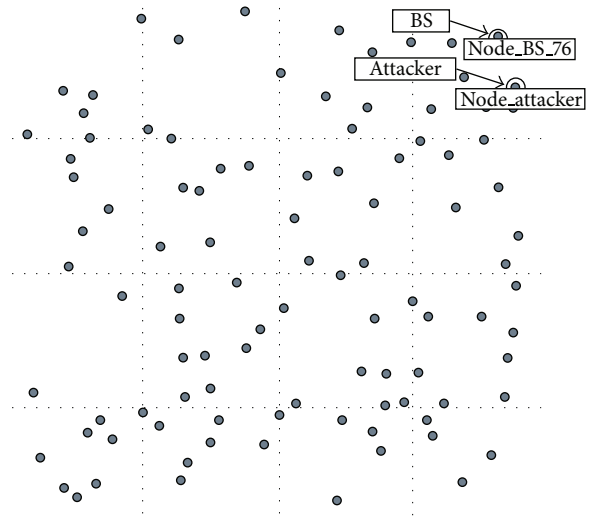


Figure 7: A WSN topology in our simulations. One hundred sensors are deployed in a 2 km × 2 km area randomly.

$N2$ and $N3$, respectively. Each source-neighbor pair is stored in M's SNMT. Whenever M receives a data packet, M will forward the packet to its designated neighbor associated with the source of the packet by using M's SNMT.

6.3. Simulation Setups and Preliminary Simulation Results. We use the simulation parameters described in Table 8. 100 sensors are deployed in a 2 km × 2 km area randomly as shown in Figure 7. Each node except the BS generates packets randomly in each 10-second period. The packets are sent to the BS. We use some of the default settings in OPNET such as 1024 bit data packet and geographic routing protocol (GRP) with a maximum of 7 retransmissions. We set each node's initial trust value to be 0.99. We choose one node near the BS as the selective forwarding DoS attacker. The simulation time is set to be 60 minutes. The attacker targets various numbers of victims. We implement two trust-based routing algorithms: trust-based GRP based on the beta trust model (Beta GRP) and our prevention routing algorithm combining the Beta GRP and our prevention method (Beta GRP-P). For our prevention method, N_{SMAX} is set to be 3 because the beta trust model with $\Theta_T = 0.7$ is used in simulations.

In addition to the three performance metrics (ACT, FAR, and EPP) used in Section 5, we use the following two performance metrics.

(1) Number of source nodes whose data packets route to the attacker through the same monitoring node ($N_{S,}$): by using this metric, as we discussed in Section 3, we can get the theoretical maximum number of victims (N_{VMAX}) which the attacker can stealthy target without being noticed by the BS.

(2) Packet delivery rate (PDR): this is the probability that a data packet is delivered to the BS. PDR is obtained by the total number of data packets delivered to the BS divided by the total number of data packets generated by all source nodes.

Table 8: OPNET simulation setup parameters for the validation of the our prevention routing algorithm.

Parameters	Setting
General	
Terrain dimension	2 km × 2 km
Number of nodes	100
Topology	Random
Max. simulation time	60 mins (3,600 seconds)
Base routing algorithm	GRP
Max. retransmissions	7 (OPNET default)
Data packet generation	
Start time–stop time	100 seconds–end of simulation
Destination	Base station
Packet arrival interval	Every 10 second
Packet size	1,024 bits
Trust model	
Type	Beta trust model
Initial trust value	0.99
Trust threshold (Θ_T)	0.7
Attack model	
Number of attackers	Single attacker
Attack type	Selective forwarding-based DoS attack

We first show how many source nodes' data packets can route through the inside attacker (located near the BS) in the simulation network topology. To see routing paths from source nodes to the BS via the attacker, we simulate the attacker forwarding packets normally toward the BS without attacking any source (attack off). Figures 8 and 9 show source nodes whose data packets route through the attacker and their routing paths to the BS when the beta GRP and our prevention routing algorithm (Beta GRP-P) are used, respectively. We can see that when beta GRP is used, the attacker receives data packets from many more source nodes as compared to our approach used.

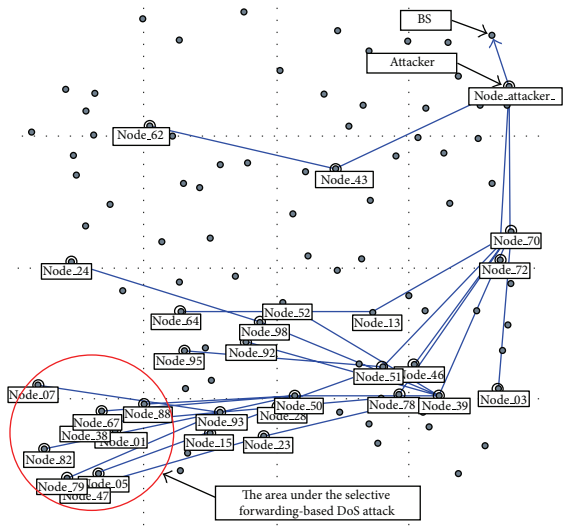


FIGURE 8: 30 potential victim source nodes and their routing paths to the BS when Beta GRP is used.

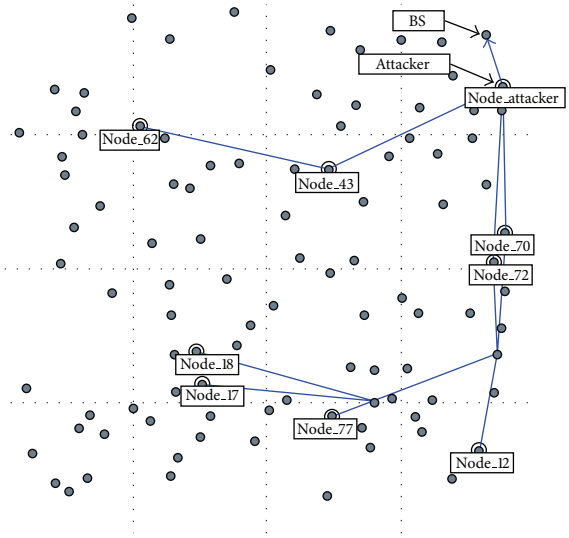


FIGURE 9: 8 potential victim source nodes and their routing paths to the BS when our prevention routing algorithm (Beta GRP-P) is used.

Specifically, Table 9 shows N_S and N_{VMAX} of the beta GRP and our approach (Beta GRP-P). For example, as the third row in bold shows, when the beta GRP is used, the attacker receives 27 source nodes' data packets from node 70. In this case, the attacker can drop up to 8 nodes' data packets completely without being detected by node 70's beta trust model theoretically ($N_{VMAX} = 8$). Meanwhile, when our approach is used, the attacker receives at most 3 source nodes from node 70 or 72. Consequently, the attacker cannot successfully launch the DoS attack against any source node without being detected by node 70 or 72.

Second, we examine ACT that indicates the ability of each approach in identifying the attacker and rerouting the victim's packets. We simulate the attacker launching the selective forwarding-based DoS attack by increasing the number of victims (J). We assume that the attacker intentionally targets source nodes from node 70 because it can have the largest number of victim source nodes. The simulation results on ACT in Table 10 reveal the following.

TABLE 9: The number of source nodes whose data packets route through the attacker (N_S) and the maximum number of victim source nodes (N_{VMAX}).

Monitoring node	Beta GRP		Our beta GRP-P	
	N_S	N_{VMAX}	N_S	N_{VMAX}
43	2	0	2	0
70	**27**	**8**	**3**	**0**
72	1	0	3	0
Total	30	8	8	0

(1) *Beta trust model alone fails to detect the attacker.* As shown in Table 10, the attacker can attack up to 8

Table 10: Avoidance completion time (in seconds) comparison when the attacker targets J victim nodes. In our prevention routing, the attacker can target at most 3 nodes.

J	Beta GRP	Our beta GRP-P
1	Fail	1,296
2	Fail	360
3	Fail	216
4	Fail	N/A
5	Fail	N/A
6	Fail	N/A
7	Fail	N/A
8	Fail	N/A
9	432	N/A

sources without being caught by node 70's beta trust model. As shown in Figure 8, the entire area monitored by the 8 victims (circled area) can be influenced by the DoS attack, and thus outside intruders can stay in or move around the area stealthily.

(2) *Our approach successfully defends against the attacker.* The proposed outsider-insider colluding attack is not effective when our prevention approach is used because the number of victims is very small. That is, since the attacker cannot target more than 3 source nodes when our approach is used, the victim area is significantly reduced compared to when the Beta GRP is used. As a result, outside intruders' movement will be limited by the small area monitored by victim nodes. In addition, if the attacker insists to attack any victim (the case of $J = 1$), the attacker will be detected by 1,296 seconds. This ACT can be reduced to around 540 seconds when our detection scheme is used together (see Table 4).

Third, FAR measures the likelihood an approach will mistakenly treat an honest node as attacker. Both approaches have almost similar FARs that range from 0.03 to 0.04. Thus, we consider that our approach does not increase FAR compared with the Beta GRP.

Finally, we report PDR and EPP. We show results when the number of victims (J) is less than 4 for comparison purposes. As shown in Table 11, our approach has a higher packet delivery performance than the beta GRP. This is because our approach can detect and avoid the attacker while the Beta GRP cannot defend against the attacker. In addition, from energy perspective, we can see that our approach is better than the Beta GRP.

7. Conclusions

In this paper, we first present a simple selective forwarding-based DoS attack and show that two representative trust mechanisms (namely, the beta trust model and the entropy trust model) fail to detect such attack. We also show the potential damage this attack could cause to the network. Second, we propose a source-level trust evaluation scheme to enhance the beta and entropy trust mechanisms to effectively detect the selective forwarding-based DoS attack. In addition, we propose two avoidance strategies to reroute the victim's packets so they can reach the BS and validate our claims and evaluate the performance of our detection and avoidance mechanisms with extensive OPNET simulations.

Table 11: Packet deliver rate (PDR) and energy per packet (mJ); J: number of victim source nodes.

J	Beta GRP		Our beta GRP-P	
	PDR	EPP	PDR	EPP
1	0.910	92.37	0.946	87.63
2	0.901	92.29	0.957	85.10
3	0.893	92.10	0.945	86.15

Finally, we introduce a prevention-routing algorithm to proactively prevent the selective forwarding-based DoS attack as a complementary defensive mechanism to our detection and avoidance methods and provide preliminary results to evaluate its performance.

There are also several directions for future work. First, how to further reduce ACT to minimize the attacker's damage to the network. Second, our preliminary results on network with lossy network show fairly large FAR. How to improve the accuracy of the proposed approach in such network is still a challenge. Finally, after the inside attackers become aware of our defensive mechanism, how they can respond to the challenge and launch more sophisticated attacks.

Acknowledgments

This material is based upon work supported in part by the Air Force Office of Scientific Research (AFOSR/RSL) under Award no. #FA95501010140 and a University Partnership with the Laboratory of Telecommunications Sciences, Contract no. H9823013D00560002. The authors would like to thank OPNET Technologies, Inc., for providing us with OPNET Wireless Modeler to validate our approaches. There is no conflict of interests.

References

[1] S. Djahel, F. Naït-Abdesselam, and Z. Zhang, "Mitigating packet dropping problem in mobile ad hoc networks: proposals and challenges," *IEEE Communications Surveys and Tutorials*, vol. 13, no. 4, pp. 658–672, 2011.

[2] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," *Ad Hoc Networks*, vol. 1, no. 2-3, pp. 293–315, 2003.

[3] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 255–265, August 2000.

[4] T. H. Hai and E. N. Huh, "Detecting selective forwarding attacks in wireless sensor networks using two-hops neighbor knowledge," in *Proceedings of the 7th IEEE International Symposium on Networking Computing and Applications (NCA '08)*, pp. 325–331, July 2008.

[5] I. Khalil, S. Bagchi, C. N. Rotaru, and N. B. Shroff, "UnMask: utilizing neighbor monitoring for attack mitigation in multihop

wireless sensor networks," *Ad Hoc Networks*, vol. 8, no. 2, pp. 148–164, 2010.

[6] X. S. Wang, Y. Z. Zhan, S. M. Xiong, and L. M. Wang, "Lightweight defense scheme against selective forwarding attacks in wireless sensor networks," in *Proceedings of the International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC '09)*, pp. 226–232, October 2009.

[7] B. Yu and B. Xiao, "Detecting selective forwarding attacks in wireless sensor networks," in *Proceedings of the 20th Parallel and Distributed Processing Symposium (IPDPS '06)*, April 2006.

[8] Y. Yu, K. Li, W. Zhou, and P. Li, "Trust mechanisms in wireless sensor networks: attack analysis and countermeasures," *Journal of Network and Computer Applications*, vol. 35, no. 3, pp. 867–880, 2012.

[9] A. A. Pirzada and C. McDonald, "Trusted greedy perimeter stateless routing," in *Proceedings of the 15th IEEE International Conference on Networks (ICON '07)*, pp. 206–211, November 2007.

[10] Y. L. Sun, W. Yu, and Z. Han, "Information theoretic framework of trust modeling and evaluation for ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 305–315, 2006.

[11] S. Ganeriwal, L. K. Balzano, and M. B. Srivastava, "Reputation-based framework for high integrity sensor networks," *ACM Transactions on Sensor Networks*, vol. 4, no. 3, article 15, 2008.

[12] Y. Cho, G. Qu, and Y. Wu, "Insider threats against trust mechanism with watchdog and defending approaches in wireless sensor networks," in *Proceedings of the IEEE Symposium Security and Privacy Workshops (SPW '12)*, pp. 134–141, 2012.

[13] H. M. Sun, C. M. Chen, and Y. C. Hsiao, "An efficient countermeasure to the selective forwarding attack in wireless sensor networks," in *Proceedings of the IEEE Region 10 Conference (TENCON '07)*, pp. 1–4, November 2007.

[14] A. Josang and R. Ismail, "The beta reputation system," in *Proceedings of the the 15th Bled Electronic Commerce Conference*, June 2002.

[15] X. Su and R. V. Boppana, "On mitigating in-band wormhole attacks in mobile ad hoc networks," in *Proceedings of the IEEE International Conference on Communications (ICC '07)*, pp. 1136–1141, June 2007.

[16] Q. Yan, M. Li, T. Jiang, W. Lou, and Y. T. Hou, "Vulnerability and protection for distributed consensus-based spectrum sensing in cognitive radio networks," in *Proceedings of the IEEE h International Conference on Computer Communications (INFOCOM '12)*, pp. 900–908, 2012.

[17] T. Zahariadis, H. Leligou, P. Karkazis et al., "Design and implementation of a trust-aware routing protocol for large WSNs," *International Journal of Network Security & Its Applications*, vol. 2, no. 3, pp. 52–68, 2010.

[18] B. Deb, S. Bhatnagar, and B. Nath, "ReInForM: reliable information forwarding using multiple paths in sensor networks," in *Proceedings of the IEEE Local Computer Networks (LCN '03)*, pp. 406–415, October 2003.

[19] J. Deng, R. Han, and S. Mishra, "Intrusion tolerance and anti-traffic analysis strategies for wireless sensor networks," in *Proceedings of the International Conference on Dependable Systems and Networks*, pp. 637–646, July 2004.

[20] "OPNET modeler wireless suite," 2012, http://www.opnet.com/solutions/network_rd/modeler_wireless.html.

[21] B. Karp and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for wireless networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 243–254, August 2000.

[22] S. Tanachaiwiwat, P. Dave, R. Bhindwale, and A. Helmy, "Location-centric isolation of misbehavior and trust routing in energy-constrained sensor networks," in *Proceedings of the 23rd IEEE International Performance, Computing, and Communications Conference (IPCCC '04)*, pp. 463–469, April 2004.

[23] N. Bhalaji, S. Banerjee, and A. Shanmugam, "A novel routing technique against packet dropping attack in adhoc networks," *Journal of Computer Science*, vol. 4, no. 7, pp. 538–544, 2008.

[24] P. Poonam, K. Garg, and M. Misra, "Trust based multi path DSR protocol," in *Proceedings of the 5th International Conference on Availability, Reliability, and Security (ARES '10)*, pp. 204–209, February 2010.

[25] I. Khalil and S. Bagchi, "Stealthy attacks in wireless ad hoc networks: detection and countermeasure," *IEEE Transactions on Mobile Computing*, vol. 10, no. 8, pp. 1096–1112, 2011.

[26] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, "Enhancing source-location privacy in sensor network routing," in *Proceedings of the 25th IEEE International Conference on Distributed Computing Systems*, pp. 599–608, June 2005.

[27] Y. Li, J. Ren, and J. Wu, "Quantitative measurement and design of source-location privacy schemes for wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 7, pp. 1302–1311, 2012.

[28] A. A. Pirzada and C. McDonald, "Establishing trust in pure ad-hoc networks," in *Proceedings of the the 27th Australasian Conference on Computer Science (ACSC '04)*, pp. 47–54, 2004.

[29] M. Johnson, M. Healy, P. Van De Ven et al., "A comparative review of wireless sensor network mote technologies," in *Proceedings of the IEEE Sensors 2009 Conference*, pp. 1439–1442, October 2009.