Trust-Based Intrusion Detection in Wireless Sensor Networks

Fenye Bao, Ing-Ray Chen, MoonJeong Chang
Department of Computer Science
Virginia Tech
{baofenye, irchen, mjchang}@vt.edu

Jin-Hee Cho
Computational and Information Sciences Directorate
U.S. Army Research Laboratory
jinhee.cho@us.army.mil

***Abstract*—We propose a trust-based intrusion detection scheme utilizing a highly scalable hierarchical trust management protocol for clustered wireless sensor networks. Unlike existing work, we consider a trust metric considering both quality of service (QoS) trust and social trust for detecting malicious nodes. By statistically analyzing peer-to-peer trust evaluation results collected from sensor nodes, each cluster head applies trust-based intrusion detection to assess the trustworthiness and maliciousness of sensor nodes in its cluster. Cluster heads themselves are evaluated by the base station. We develop an analytical model based on stochastic Petri nets for performance evaluation of the proposed trust-based intrusion detection scheme, as well as a statistical method for calculating the false alarm probability. We analyze the sensitivity of false alarms with respect to the minimum trust threshold below which a node is considered malicious. Our results show that there exists an optimal trust threshold for minimizing false positives and false negatives. Further, the optimal trust threshold differs depending on the anticipated wireless sensor network lifetime.**

***Index Terms*—Trust management, intrusion detection, wireless sensor networks, security, false positives, false negatives.**

I. INTRODUCTION

A wireless sensor network (WSN) usually consists of a large number of tiny sensor nodes (SNs) deployed in an operational area for data sensing, aggregating, and processing. WSNs have been applied in transportation, agriculture, homeland security, and battlefield applications. The exposure to natural environments and the inherent unreliability of wireless transmission make a WSN vulnerable to many attacks [1]. SNs deployed in hostile environments for military applications also could be compromised through captures and become malicious. Moreover, due to severe resource constraints of SNs, such as energy, memory, and computational power, traditional energy-consuming defense mechanisms like public-key infrastructure [10] and host-based intrusion detection techniques [6] may not be feasible.

Malicious attacks to WSNs can be classified into outsider attacks and insider attacks. While most outsider attacks such as spoofing, replay, and Sybil attacks can be prevented by authentication and cryptography, insider attacks are much harder to deal with. In this paper, we develop a trust-based intrusion detection system (IDS) scheme utilizing a highly scalable hierarchical trust management protocol for clustered wireless sensor networks to detect inside attackers.

Unlike existing work, we consider not only QoS trust (*energy* and *cooperativeness*) derived from communication networks but also social trust (*honesty*) derived from social networks [2] to judge if a node is compromised. We develop a probability model based on the stochastic Petri nets (SPN) to describe the behaviors of each SN or cluster head (CH). In our protocol, each node subjectively evaluates other peers periodically. With peer-to-peer trust evaluations reported from SNs, a CH obtains a comprehensive trust report toward all SNs in its cluster and can perform statistical analysis to identify and exclude malicious nodes in the network. CHs themselves are evaluated by the base station taking in peer-to-peer trust evaluation inputs from other CHs. This hierarchical structure reduces network traffic by eliminating cross-cluster communications among SNs. More importantly, we develop a statistical method to predict false positive and false negative probabilities and identify optimal design settings under which false positives and false negatives are minimized.

In the literature, Wang et al. [8] proposed an intrusion detection mechanism based on trust (IDMTM) for mobile ad hoc networks (MANETs). They employed the concepts of *evidence chain* and *trust fluctuation* to evaluate a node in the network, with the *evidence chain* detecting misbehaviors of a node, and the *trust fluctuation* reflecting the high variability of a node's trust value over a time window. Ebinger et al. [3] introduced a cooperative intrusion detection method also for MANETs based on trust evaluation and reputation exchange. They split the reputation information into *trust* and *confidence* for reputation exchanges and then combine them into *trustworthiness* for intrusion detection. In WSNs, several trust management protocols [4, 5, 7] have been proposed for network security, data integrity, and secure routing. However, most work only considered a flat WSN structure. Notably Shaikh et al. [7] proposed a group-based trust management scheme for clustered WSNs. Their hierarchical structure used is similar to our work. However, they only considered QoS metrics (e.g., message delivery ratio in a time window) based on direct observations and no IDS design or evaluation was discussed. To the best of our knowledge, our work is the first to use trust to implement intrusion detection functionality and evaluate its effectiveness for clustered WSNs.

II. SYSTEM MODEL

We consider a clustered WSN consisting of multiple clusters, each with a cluster head (CH) and a number of SNs in the corresponding geographical area with CHs having more computational and energy resources than SNs. The CH in each cluster may be selected based on an election protocol such as HEED [9]. A SN forwards its sensor reading to its CH and the CH then forwards the data to the base station or a destination node (or sink node) through other CHs.

Our trust-based IDS scheme considers the effect of both *social trust* and *QoS trust* on trustworthiness or maliciousness. In the literature, social trust may include friendship, honesty, privacy, similarity, betweenness centrality, and social ties (strengths) [2]. QoS trust may include competence, protocol conformance, reliability, task completion capability, etc. In this work, we adopt *honesty* to measure social trust derived from social networks and adopt *energy* (for measuring competence) and *cooperativeness* to measure QoS trust derived from communication networks, as these can be considered as indicators of trustworthiness. The *honesty* trust component is measured through evidences of dishonesty such as false self-reporting [1, 5], trust fluctuation [8] and abnormal trust recommendations (i.e., outliers relative to recommendations received from other recommenders). The energy trust component provides a piece of evidence because a compromised node usually performs energy-consuming attacks, such as disseminating bogus messages. Lastly, a compromised node usually manifests itself as being uncooperative because of selective forwarding or message dropping attacks to disrupt message routing in WSNs.

We assume a cognitive WSN in which a smart SN may adjust its behavior dynamically according to its own operational state and environmental conditions. A SN not necessarily compromised may become uncooperative just to save its energy. The uncooperative behavior is typically reflected by stopping sensing functions and arbitrarily dropping messages. If not compromised, an uncooperative SN may become cooperative to serve system goals such as service availability if few cooperative neighbor nodes are around. A SN is more likely to be compromised when it has low energy (because a node with high energy may perform better energy-consuming defenses against attackers), or when it has more compromised neighbors around. A compromised SN can perform strong attacks such as black hole attacks, good-mouthing attacks (recommending a bad node as a good node), and bad-mouthing attacks (recommending a good node as a bad node) through which it exhibits dishonest behaviors, and weak attacks such as message dropping, and selective packet forwarding through which it exhibits uncooperative behaviors. After a SN or CH is compromised, it will consume more energy to perform attacks. Such attack behaviors are manifested as evidences against the honesty, energy, and cooperativeness trust properties.

III. Hierarchical Trust Management for Intrusion Detection

Our hierarchical trust management protocol maintains two levels of trust: *SN-level* trust and *CH-level* trust. Each SN evaluates other SNs in the same cluster while each CH evaluates other CHs and SNs in its cluster. The peer-to-peer trust evaluation is periodically updated based on either direct observations or indirect observations. When two nodes are neighbors within radio range, they evaluate each other based on direct observations via snooping or overhearing. Each SN sends its trust evaluation results toward other SNs in the same cluster to its CH. Each CH performs trust evaluation toward all SNs within its cluster. Similarly, each CH sends its trust evaluation results toward other CHs in the WSN to the base station. The base station performs trust evaluation toward all CHs in the system.

Our peer-to-peer trust evaluation process considers three different trust components as described earlier, namely, honesty, energy, and cooperativeness. The trust value that node *i* evaluates toward node *j* at time *t*, $T_{ij}(t)$, is represented as a real number in the range of [0, 1] where 1 indicates complete trust, 0.5 ignorance, and 0 distrust. $T_{ij}(t)$ is computed by:

$$\begin{aligned} T_{ij}(t) = w_1 T_{ij}^{honesty}(t) + w_2 T_{ij}^{energy}(t) \\ + w_3 T_{ij}^{cooperativeness}(t) \end{aligned} \quad (1)$$

where w_1, w_2, and w_3 are weights associated with these three trust components with $w_1 + w_2 + w_3 = 1$.

A. Peer-to-Peer Trust Evaluation

This section describes how peer-to-peer trust evaluation is conducted, particularly between two SNs or two CHs. Specifically, when a trustor (node *i*) evaluates a trustee (node *j*) at time *t*, it updates $T_{ij}^X(t)$ where *X* indicates a trust component as follows:

$$T_{ij}^X(t) = \begin{cases} (1-\alpha)T_{ij}^X(t-\Delta t) + \alpha T_{ij}^{X,direct}(t), \\ \quad if\ i \text{ and } j \text{ are } 1-\text{hop neighbors}; \\ \underset{k \in N_i}{\text{avg}}\left\{\gamma T_{ij}^X(t-\Delta t) + (1-\gamma)T_{kj}^{X,recom}(t)\right\}, \\ \quad \text{otherwise.} \end{cases} \quad (2)$$

In Equation 2, if node *i* is a 1-hop neighbor of node *j*, node *i* will use its direct observations ($T_{ij}^{X,direct}(t)$) and past experiences ($T_{ij}^X(t-\Delta t)$ where Δt is a trust update interval) toward node *j* to update $T_{ij}^X(t)$. A parameter α ($0 \le \alpha \le 1$) is used here to weight these two contributions and to consider trust decay over time. If the application context knowledge justifies placing higher trust on recent direct observations over past experiences, a larger α (greater than 0.5) may be used; otherwise equal weighting with $\alpha = 0.5$ may be considered. Here $T_{ij}^{X,direct}(t)$ indicates node *i*'s trust value toward node *j* based on direct observations accumulated over the time period $[0, t]$ possibly with a higher priority given to recent interaction experiences over the time period $[t - \Delta t, t]$. Below we describe how each trust component value $T_{ij}^{X,direct}(t)$ can be obtained based on direct observations:

- $T_{ij}^{honesty,direct}(t)$: This refers to the belief of node *i* that node *j* is honest based on node *i*'s direct observations toward node *j*. Node *i* can monitor node *j*'s dishonesty evidences including abnormal trust recommendations, false self-reporting [1, 5], and trust fluctuation [8] over the time period $[0, t]$ to estimate $T_{ij}^{honesty,direct}(t)$.

- $T_{ij}^{energy,direct}(t)$: This indicates the percentage of energy remaining in node j that node i directly observes at time t. As a neighbor, node i can overhear or even monitor node j's packet transmission activities over the time period $[0, t]$ to estimate $T_{ij}^{energy,direct}(t)$.
- $T_{ij}^{cooperativeness,direct}(t)$: This provides the degree of cooperativeness of node j as evaluated by node i based on direct observations over the time period $[0, t]$. Node i can apply overhearing or snooping techniques to detect uncooperativeness behaviors such as packet dropping or selective forwarding and may give recent interaction experiences a higher priority over old experiences in estimating $T_{ij}^{cooperativeness,direct}(t)$.

On the other hand, if node i is not a 1-hop neighbor of node j, node i will use its past experiences ($T_{ij}^{X}(t - \Delta t)$) and recommendations ($T_{kj}^{X,recom}(t)$ where k is a recommender) to update $T_{ij}^{X}(t)$. A parameter γ is used here to weight these two contributions and to consider trust decay over time as follows:

$$\gamma = \frac{1}{1 + \beta T_{ik}(t)} \tag{3}$$

Here we introduce another parameter $\beta \geq 0$ to specify the impact of "indirect recommendations" on $T_{ij}^{X}(t)$ such that the weight assigned to indirect recommendations is normalized to $\beta T_{ik}(t)$ relative to 1 assigned to past experiences. Essentially, the contribution of recommended trust increases proportionally as either $T_{ik}(t)$ or β increases. Instead of having a fixed weight ratio $T_{ik}(t)$ to 1 for the special case in which $\beta = 1$, we allow the weight ratio to be adjusted by changing the value of β and test its effect on protocol resiliency against malicious recommendation attacks, such as good-mouthing and bad-mouthing attacks. Here, $T_{ik}(t)$ is node i's overall trust value toward node k as a recommender (for node i to assess if node k provides correct information). Note that node i can choose all its 1-hop neighbors (N_i) as recommenders. The new trust value $T_{ij}^{X}(t)$ in this case would be the average of the combined trust values of past trust information and recommendations collected at time t.

B. Trust-based Intrusion Detection

Each SN reports its trust evaluation toward other SNs in the same cluster to its CH. The CH then applies statistical analysis principles (such as Equation 4 below) to $T_{ij}(t)$ values received to perform CH-to-SN trust evaluation toward node j. Our trust-based IDS is based on selecting a system minimum trust threshold below which a node is considered compromised and needs to be excluded from sensor reading and routing duties. Specifically a CH, c, when evaluating a SN, j, will perform intrusion detection by comparing the system minimum trust threshold T^{th} with node j's trust value, $T_{cj}(t)$, obtained by:

$$T_{cj}(t) = \underset{i \in M_c \wedge T_{ci}(t) \geq T^{th}}{\text{avg}} \{T_{ij}(t)\} \tag{4}$$

where M_c is the set of SNs in the cluster. CH c will announce j as compromised if $T_{cj}(t)$ is less than T^{th}; otherwise, node j is not compromised. Note that we only take into account the trust values received from those SNs which are not considered compromised by the CH. The CH can also leverage $T_{ij}(t)$ values received from SNs in its cluster to detect if there is any outlier as a piece of evidence against dishonesty.

IV. Performance Model

We develop a probability model based on SPN techniques to describe the behaviors of each SN or CH, with the objective to yield energy, cooperativeness and maliciousness (for honesty) status of a node dynamically. We choose SPN as our analytical tool due to its capability to represent a large number of states for complex systems. Leveraging SPN model outputs which provide actual status of SNs and CHs in the system dynamically, we are able to accurately predict peer-to-peer trust values obtainable by each SN or CH, and, consequently, evaluate the performance of the trust-based IDS scheme.

A. A Probability Model for Describing Node Behaviors

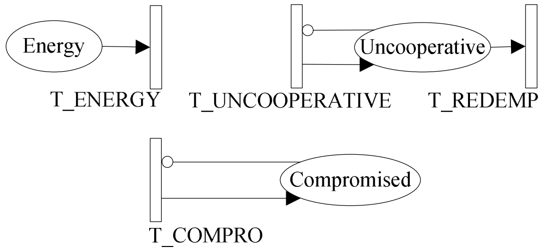

Figure 1: SPN Model for a Sensor Node or a Cluster Head.

Figure 1 shows the SPN model that describes the behaviors of a SN (or a CH). Without loss of generality, we consider a WSN consisting of N_{SN} SNs uniformly distributed in an M by M square-shaped operational area and N_{CH} CHs. Each SN is attached to a CH based on its location. CHs and SNs have radio range of R and r, respectively. The trust update interval is Δt. All nodes are stationary after the initial deployment. Below we explain how we construct the SPN model for describing the behaviors of a single node.

Energy: Place *Energy* represents the remaining energy level of the node. The initial number of tokens in place *Energy* is set to E_{init}. A token will be released from place *Energy* when transition *T_ENERGY* is triggered. The rate of transition *T_ENERGY* indicates the energy consumption rate. A CH consumes more energy than a SN. The energy consumption rate is affected by a node's state. It is higher when a node is compromised because it takes energy to perform attacks. We denote Δ_{E-SN}, Δ_{E-CH} and $\Delta_{E-compromised}$ as the amount of energy consumed per Δt time for a normal SN, a normal CH, and a compromised node, respectively, which can be obtained by analyzing historical data with $\Delta_{E-SN} < \Delta_{E-CH} < \Delta_{E-compromised.}$ The energy consumption rate is multiplied with ρ (0≤ ρ ≤1) if the node is uncooperative.

Uncooperativeness: We model the uncooperative behavior as follows: A node may become uncooperative to save energy.

An uncooperative node may stop reading data and drop packets it receives. A node will decide if it wants to become uncooperative upon every time interval T_s according to its remaining energy and the number of cooperative neighbors. Also a compromised node is likely to be uncooperative as it performs weak attacks such as packet dropping or selective packet forwarding. An uncooperative node can redeem itself to become cooperative upon every trust update interval (Δt). We model these behaviors by putting a token into place *Uncooperative* when transition *T_UNCOOPERATIVE* is triggered or by removing the token from place *Uncooperative* when transition *T_REDEMP* is triggered. A token in place *Uncooperative* thus indicates that the node is uncooperative. A node's uncooperative probability is modeled by:

$$P_{uncooperative} = \frac{1}{2}\left(\frac{E_{consumed}}{E_{init}} + \frac{N_{neighbor}^{cooeprative}}{N_{neighbor}}\right) \qquad (5)$$

where $E_{consumed}$ is energy consumed and E_{init} is the node's initial energy level. Thus $E_{consumed}/E_{init}$ represents the percentage of energy consumed. $N_{neighbor}^{cooperative}/N_{neighbor}$ is the percentage of cooperative neighbors where $N_{neighbor}^{coopertive}$ is the number of cooperative neighbors and $N_{neighbor}$ is the total number of neighbors. This models the behavior that a node's uncooperative probability tends to be lower when the node has more energy and higher when the node has more cooperative neighbors as there are sufficient cooperative neighbors around to take care of sensor tasks. It also models the behavior that a compromised SN is likely to be uncooperative when it has low energy, thus performing only weak attacks such as packet dropping. Thus, the rates of transitions *T_UNCOOPERATIVE* and *T_REDEMP* are given by $P_{uncooperative}/T_s$ and $(1 - P_{uncooperative})/\Delta t$ respectively. Initially all nodes are cooperative with no token in place *Uncooperative*.

Maliciousness: A node is compromised when transition *T_COMPRO* fires and a token is put in place *Compromised*. The rate of transition *T_COMPRO* is modeled by:

$$\lambda_c = \lambda_{c-init}\left(\frac{E_{init}}{E_{remain}} + \frac{N_{neighbor}^{compromised}}{N_{neighbor}^{healthy}}\right) \qquad (6)$$

where λ_{c-init} is the initial node compromising rate which can be obtained by first-order approximation based on historical data about the targeted network environment. E_{init} and E_{remain} indicate a node's initial energy and remaining energy, respectively. $N_{neighbor}^{compromised}$ and $N_{neighbor}^{healthy}$ are the numbers of compromised and healthy nodes in the neighborhood. $N_{neighbor}^{compromised}/N_{neighbor}^{healthy}$ refers to the ratio of the number of compromised 1-hop neighbors to the number of healthy (uncompromised) 1-hop neighbors. Equation 6 models the behavior that a node is more likely to become compromised when it has low energy because it may not spare its energy to perform energy-consuming defense mechanisms, or when there are many 1-hop neighboring compromised nodes around it because of collusive attacks. Note that all nodes are healthy, i.e., not compromised, initially.

The overall performance model for describing the collective behavior of a WSN consists of *N* SPN subnet models, one for each SN, and N_{CH} SPN subnet models, one for each CH, each with different energy consumption, uncooperative/redemption and compromise rates. Below we describe how one could leverage SPN outputs to obtain peer-to-peer trust values as the basis for performance evaluation of our proposed trust-based intrusion detection scheme.

B. Trust Evaluation

Recall that under our trust management protocol, node *i* will subjectively assess its trust toward node *j*, $T_{ij}(t)$, based on its direct observations and indirect recommendations obtained toward node *j* according to Equations 1 and 2. In particular, node *i* will apply snooping or overhearing techniques to monitor node *j* closely to compute $T_{ij}^{X,direct}(t)$ based on direct observations over the time period $[0, t]$. As a result, $T_{ij}^{X,direct}(t)$ computed by node *i* will fairly accurately reflect actual status of node *j* at time *t*. Leveraging the SPN model developed which provides actual status of each node dynamically, we can easily compute $T_{ij}^{honesty,direct}(t)$, $T_{ij}^{energy,direct}(t)$, and $T_{ij}^{cooperativeness,direct}(t)$ by simply checking the status of node *j* at time *t* in node *j*'s SPN model as listed in Table 1. Once $T_{ij}^{X,direct}(t)$ is computed, node *i* will compute $T_{ij}^{X}(t)$ based on Equation 2 and subsequently compute $T_{ij}(t)$ based on Equation 1. Each SN then reports its trust evaluation values to its CH for CH-to-SN trust evaluation and CH-SN intrusion detection based on Equation 4. The same procedure is applied for base station-to-CH intrusion detection.

Table 1: Computing $\mathbf{T_{ij}^{X,direct}(t)}$ for Component *X* based on Actual Node Status.

$T_{ij}^{X,direct}(t)$	**Value**	**Status (of node *j*)**
$T_{ij}^{honesty,direct}(t)$	1 0	*If mark*(*Compromised*) = 0 Otherwise
$T_{ij}^{energy,direct}(t)$	*mark*(*Energy*)/E_{init}	
$T_{ij}^{cooperativeness,direct}(t)$	1 0	*If mark*(*Uncooperative*) = 0 Otherwise

C. Performance of Trust-based Intrusion Detection

We develop a statistical method to predict the performance (i.e. false positives and false negatives) of our trust-based IDS scheme. Recall that in Equation 4, each CH, *c*, receives trust evaluation values toward node *j*, $T_{ij}(t)$'s, from *n* SNs (not diagnosed as compromised) and uses the mean value, $T_{cj}(t) = \overline{T_{ij}(t)}$, to decide whether node *j* is compromised or not. Statistically, *c* should announce node *j* as compromised if the expected trust value toward node *j*, $\mu_j(t)$, is below the threshold T^{th}; otherwise, node *j* is announced as healthy. Consider that the trust value toward node *j* is a random variable and $T_{ij}(t)$'s submitted by *n* SNs are *n* samples of this random variable. Then we have a random variable $X_j(t)$ following *t*-distribution with *n* - 1 degree of freedom:

$$X_j(t) = \frac{\overline{T_{ij}(t)} - \mu_j(t)}{S_j(t)/\sqrt{n}} \qquad (7)$$

where $\overline{T_{ij}(t)}$ and $S_j(t)$ are sample mean and sample standard deviation of node *j*'s trust value, respectively. Thus, the probability that node *j* is diagnosed as a compromised node at time *t* is:

$$\Theta_j(t) = \Pr\left(\mu_j(t) < T^{th}\right)$$
$$= \Pr\left(X_j(t) > \frac{\overline{T_{ij}(t)} - T^{th}}{S_j(t)/\sqrt{n}}\right) \qquad (8)$$

The false positive of the IDS can be obtained by calculating $\Theta_j(t)$ under the condition that node j is not compromised. Similarly, the false negative probability can be obtained by calculating $1 - \Theta_j(t)$ under the condition that node j is compromised.

$$P_j^{fp}(t) = \Pr\left(X_j(t) > \frac{\overline{T_{ij}^N(t)} - T^{th}}{S_j^N(t)/\sqrt{n}}\right) \qquad (9)$$

$$P_j^{fn}(t) = \Pr\left(X_j(t) \leq \frac{\overline{T_{ij}^C(t)} - T^{th}}{S_j^C(t)/\sqrt{n}}\right) \qquad (10)$$

Equations 9 and 10 give the false positive probability, $P_j^{fp}(t)$, and false negative probability, $P_j^{fn}(t)$, of our proposed trust-based intrusion detection scheme at time *t*, respectively. $\overline{T_{ij}^N(t)}$ and $S_j^N(t)$ are the mean value and standard deviation of node *j*'s trust values reported by other nodes in the same cluster, under the condition that node *j* is not compromised. $\overline{T_{ij}^C(t)}$ and $S_j^C(t)$ are the mean value and standard deviation, under the condition that node j is compromised. $T_{ij}^N(t)$ and $T_{ij}^C(t)$ can be easily obtained by applying the Bayes' theorem to the calculation of $T_{ij}(t)$.

$P_j^{fp}(t)$ and $P_j^{fn}(t)$ vary over time. The average false positive and false negative probabilities, denoted by P_j^{fp} and P_j^{fn}, can be obtained by weighting on the probability of node *j* being compromised at time *t*, i.e.,

$$P_j^{fp} = \frac{\sum_{t=0}^{L}\left(P_j^{fp}(t)\left(1 - P_j^C(t)\right)\right)}{\sum_{t=0}^{L}\left(1 - P_j^C(t)\right)} \qquad (11)$$

$$P_j^{fn} = \frac{\sum_{t=0}^{L}\left(P_j^{fn}(t)P_j^C(t)\right)}{\sum_{t=0}^{L}P_j^C(t)} \qquad (12)$$

where $P_j^C(t)$ is the probability that node *j* is compromised at time *t* which can be obtained from the SPN model output, and *L* is the anticipated WNS lifetime period over which the weighted calculation is performed.

V. Numerical Results

In this section, we show numerical results obtained from the performance model described in Section IV. Table 2 lists default parameters used. We consider a WSN with 400 SNs and 25 CHs uniformly distributed in a 400*m*×400*m* area. The WSN is deployed in a hostile environment with the node's average compromising interval in the range of [320*hrs*, 1440*hrs*]. We consider the worst case of good-mouthing (providing the highest trust value for a malicious node) and bad-mouthing attacks (providing the lowest trust value against a good node). The initial trust value is set to 1 since all nodes are initially healthy (uncompromised) and cooperative.

Table 2: Default Parameter Values Used.

Param	Value	Param	Value	Param	Value
M	400*m*	R	150*m*	r	50*m*
N_{SN}	400	N_{CH}	25	Δt	80*hrs*
α	0.5	β	1.0	$1/\lambda_{c\text{-}init}$	[320,1440]*hrs*
$\Delta_{E\text{-}SN}$	80*hrs*	$\Delta_{E\text{-}CH}$	160*hrs*	$\Delta_{E\text{-}compromised}$	240*hrs*
ρ	1/3	T_s	[80,480]*hrs*	w_1,w_2,w_3	1/3
E_{init}	[360,480] *days* for SNs, [720,960] *days* for CHs.				

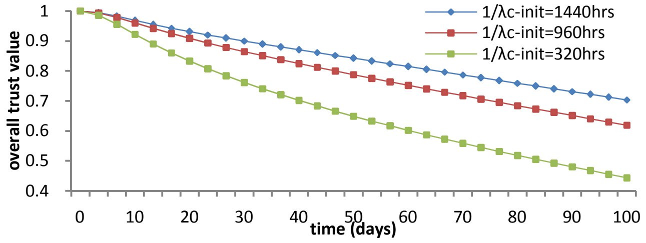

Figure 2: Trust Evaluation.

Figure 2 compares the overall trust (using equal weighting with w_1:w_2:w_3=1/3:1/3:1/3) toward a SN randomly picked with the node's compromising interval varying from 320*hrs* to 1440*hrs*. We observe that the trust value of a node with a higher compromising rate drops more quickly, which makes it easy to detect by our trust-based IDS scheme.

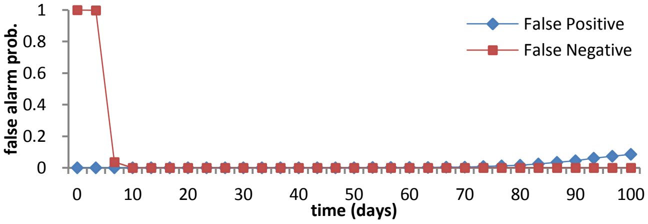

Figure 3: False Alarm Probabilities as a Function of Time.

Figure 3 shows the false positive and false negative probabilities of our trust-based intrusion detection scheme as a function of time *t* with *L* = 100 *days* and T^{th} = 0.8. We first note that *false negatives* are due to IDS bad nodes as good nodes, the effect of which is especially pronounced when *t* is small at which a bad node's trust level is likely to be high since all nodes have high energy and cooperativeness trust values initially. On the other hand, *false positives* are due to IDS misidentifying good nodes as bad nodes, the effect of which is especially pronounced when *t* is large at which a good node's trust value is likely to be low as much energy is consumed and a good node may exhibit uncooperative behaviors to save energy. This is the trend exhibited in Figure 3. When *t* is small, the false negative probability is high because initially the trust value of every node is high and thus IDS is more likely to miss a compromised node. As time progresses, the false negative probability drops but the false positive probability increases slowly since the trust value becomes lower and the IDS is more likely to misdiagnose a good node as compromised. We observe that after the initial warm-up period after nodes have a chance to perform peer-to-peer trust evaluation and the trust values are summarized for trust-based intrusion detection, we can obtain acceptable low false alarms during most of the useful network lifetime.

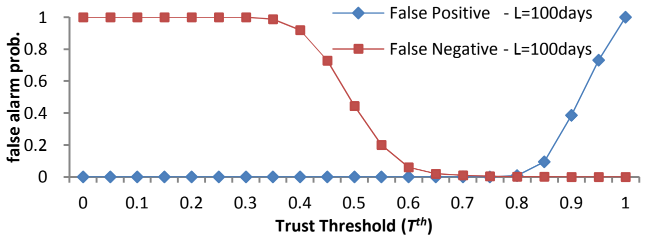

Figure 4: False Alarm as a Function of T^{th} with L=100 days.

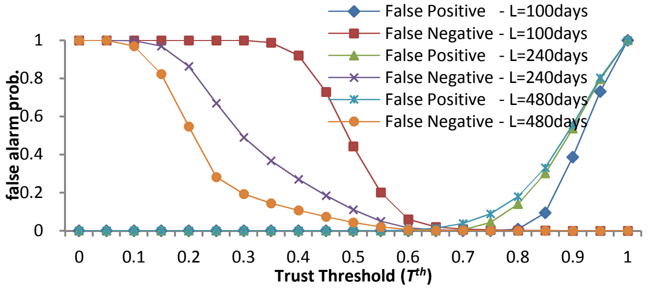

Figure 5: False Alarm as a Function of T^{th} with Varying L.

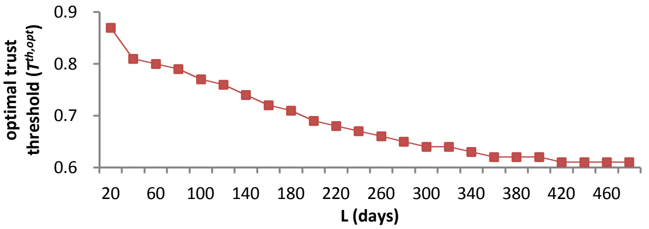

Figure 6: Optimal Trust Threshold as a Function of L.

Figure 4 shows the sensitivity of the false alarm probability with respect to the system minimum trust threshold T^{th} below which a node is considered compromised. We use Equations 11 and 12 for the weighted calculation of false positive and false negative probabilities over the time period [0, 100] *days*. One can note that as the minimum trust threshold, T^{th}, increases, the overall false negative decreases while the overall false positive increases. There exists an optimal trust threshold $T^{th,opt}$ at which both false negative and false positive probabilities are minimized. For L=100 *days* and the network environment characterized by the set of parameter setting as listed in Table 2, the optimal trust threshold $T^{th,opt}$ is in the range of [0.70, 0.80] at which both false positive and false negative probabilities are lower than 0.01. Figure 5 shows the same as Figure 4, except that we vary the anticipated network lifetime L from 100 to 480 *days* in the calculation of the false alarm probability. We observe from Figure 5 that the optimal trust threshold $T^{th,opt}$ shifts toward left (becoming lower) as the anticipated WSN lifetime L increases. Figure 6 shows that the optimal trust threshold $T^{th,opt}$ decreases as the anticipated network lifetime (L) increases because a node's trust value decreases over time due to energy depletion even if the node is not compromised.

VI. Conclusion

In this paper, we proposed a trust-based IDS scheme leveraging a hierarchical trust management protocol for WSNs. We considered a composite trust metric deriving from both social trust (honesty) and QoS trust (energy and cooperativeness) as an indicator of maliciousness. We developed a probability model based on SPN techniques to describe the behaviors of SNs or CHs for trust evaluation and intrusion detection, as well as a statistical method to predict the false alarm probabilities of the trust-based IDS scheme. The experimental results show that a node with high compromising rate can be easily detected, thus supporting the idea of using trust to implement IDS functionality. We analyzed the sensitivity of false alarm probabilities with respect to the minimum trust threshold below which a node is considered compromised, and we discovered that there exists an optimal trust threshold at which both false positive and false negative probabilities are minimized and that the optimal trust threshold decreases as the network lifetime increases.

There are several future research directions, including (a) considering more social trust components other than honesty and studying their effects on false alarms; (b) devising and validating a decentralized CH trust evaluation scheme for autonomous WSNs without base stations; (c) investigating the impact of the number of clusters and the trust update interval to protocol performance; (d) conducting a comparative performance analysis of existing trust-based IDS techniques for WSNs, and (e) investigating the feasibility of using trust to implement IDS functionality in more dynamic networks such as mobile WSNs or MANETs.

References

[1] X. Chen, K. Makki, K. Yen, and N. Pissinou, "Sensor network security: a survey," *IEEE Communication Surveys & Tutorials*, vol. 11, no. 2, June 2009, pp. 52-73.

[2] E.M. Daly and M. Haahr, "Social network analysis for information flow in disconnected delay-tolerant MANETs," *IEEE Transactions on Mobile Computing*, vol. 8, no. 5, May 2009, pp. 606-621.

[3] P. Ebinger and N. Bibmeyer, "TEREC: Trust evaluation and reputation exchange for cooperative intrusion detection in MANETs," *7th Annual Comm. Networks and Services Research Conf.* 2009, pp. 378-385.

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

[5] K. Liu, N. Abu-Ghazaleh, and K.-D. Kang, "Location verification and trust management for resilient geographic routing," *J. Parallel and Distributed Computing*, vol. 67, no. 2, 2007, pp. 215-28.

[6] A. Mishra, K. Nadkarni, and A. Patcha, "Intrusion detection in wireless ad hoc networks," *IEEE Wireless Communications*, vol. 11, no. 1, 2004, pp. 48-60.

[7] R.A. Shaikh, H. Jameel, B.J. d'Auriol, H. Lee, S. Lee, and Y.J. Song, "Group-based trust management scheme for clustered wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 11, Nov. 2009, pp. 1698-1712.

[8] F. Wang, C. Huang, J. Zhang, and C. Rong, "IDMTM: A novel intrusion detection mechanism based on trust model for ad hoc networks," *22nd IEEE International Conference on Advanced Information Networking and Applications*, 2008, pp. 978-84.

[9] O. Younis and S. Fahmy, "HEED: A hybrid energy efficient, distributed clustering approach for ad hoc sensor network", *IEEE Transaction on Mobile Computing*, vol. 3, no. 3, 2004, pp. 366-379.

[10] Y. Zhang, W. Liu, W. Lou, Y. Fang, and Y. Kwon, "AC-PKI: anonymous and certificateless public-key infrastructure for mobile ad hoc networks," *40th IEEE International Conference on Communications*, May 2005, pp. 3515-3519.