On-Off Attacks Mitigation against Trust Systems in Wireless Sensor Networks

Nabila Labraoui, Mourad Gueroui, Larbi Sekhri

On-Off Attacks Mitigation against Trust Systems in Wireless Sensor Networks

Nabila Labraoui [1], Mourad Gueroui [2], Larbi Sekhri [3]

[1]STIC Laboratory, University of Tlemcen, Algeria
nabila.labraoui@mail.univ-tlemcen.dz
[2] PRISM Laboratory, University of Versailles Saint-Quentin en Yvelines, France
mourad.guerroui@prism.uvsq.fr
[3] ICN Laboratory, University of Oran, Algeria
larbi.sekhri@univ-oran.dz

Abstract. Trust and reputation systems have been regarded as a powerful tool to defend against insider attacks caused by the captured nodes in wireless sensor networks (WSNs). However, trust systems are vulnerable to on-off attacks, in which malicious nodes can opportunistically behave good or bad, compromising the network with the hope that bad behavior will be undetected. Thus, malicious nodes can remain trusted while behaving badly. In this paper, we propose O^{2}Trust, On-Off attack mitigation for Trust systems in wireless sensor networks. O^{2}Trust adopts the penalty policy against the misbehavior history of each node in the network as a reliable factor that should influence on the calculation of the trust value. This punishment future helps to perceive malicious node that aim to launch intelligent attacks against trust-establishment and consequently on-off attack is mitigated efficiently.

1 Introduction

Wireless sensor networks (WSNs) [1] provide a technological basis for many different security critical applications such as critical infrastructure monitoring, healthcare and battlefield. However, WSNs are often deployed in unattended, harsh and hostile environment that makes them under the threat of various types of attacks, including node compromise. In a node capture attack, an adversary tries to physically tamper with a node in order to extract the cryptographic secrets. Hence, the compromised node can participate in the network as a legitimate node and cannot be identified whether it is genuine or not. This attack can give rise to many subsequent powerful insider attacks [2]. Unfortunately, traditional safety mechanisms based on cryptography, cannot adequately defend against network insider attacks, although they are effective to outsider attacks [3].

Trust and reputation systems have been regarded as a powerful tool to defend against insider attacks caused by the captured nodes in WSNs [4]. Generally, trust establishment is used to record feedback about the security evaluations of other nodes. Thus, efficient trust management systems can help well-behaved nodes to avoid working with misbehaving nodes, as well as to detect these malicious ones [5]. However, building a robust trust and reputation system presents several important challenges on its own [6], because it is susceptible to attacks such as bad-mouthing and on-off attacks [7, 8]. In this work we consider the on-off attack in which malicious nodes can opportunistically behave good or bad, compromising the network with the hope that bad behavior will be undetected. Malicious nodes can remain trusted while behaving badly. As it is mentioned in [8], almost all reputation-based trust models are vulnerable to on-off attack, because they focus more on recent behavior of the node rather than comprehensively combining the nodes' past behavior with its instantaneous behavior. As a consequence, a malicious node can easily dissimulate any misbehavior history by either displaying good behavior or waiting during later time periods to increase its trust value. By this way, it continues its attack.

To address the above problem, we present in this paper O^{2}Trust: On-Off attack mitigation for Trust systems in wireless sensor networks. O^{2}Trust adopts the *Penalty Policy* against the misbehavior history of each node in the network. Unlike previous trust models that focus on recent behavior and thus are not sensitive enough to perceive contradictory behavior, in our proposal, we focus on frequency misbehavior history as a reliable factor that should influence on the calculation of the trust value for a node. This punishment future helps to perceive malicious node that aim to launch intelligent attacks against trust-establishment and consequently on-off attack is mitigated.

The rest of this paper is organized as follows. In Section 2, we present an overview of related works. Section 3 describes the proposed trust model. Evaluation results and theoretical analyses of the proposed model are provided in Section 4 and Section 5. Section 6 concludes the paper.

2 Related works

Ganeriwal and Srivastava [9] proposed the first reputation and trust based model designed and developed exclusively for sensor networks; the RFSN (Reputation-based Framework for high integrity Sensor Networks) model uses the Beta distribution as a mathematical tool to represent and continuously update trust and reputation. To differently weight the old and new interactions, an aging factor is introduced for trust updating; more weight is given to recent interactions. Chen proposed in [10], a Task-based Trust framework for Sensor Networks (TTSN), where sensor nodes maintain reputation for neighbor nodes of several different tasks and use the reputation to evaluate their trustworthiness. The method for trust calculation and trust updating is almost the same as described in RSFN [9]. Sheikh et al. [11] proposed GTMS a Group-based Trust Management Scheme, in which the whole group will get a single trust value. He et al. [12] proposed attack-resistant and lightweight trust management scheme (ReTrust) for medical sensor network followed a hierarchical architecture, comprised of master nodes and sensor nodes. The authors use the window mechanism to forget previous actions. Moreover, they introduce an aging-factor parameter, which is different for each time unit m in the window.

3 The proposed trust model: O^{2}Trust

In this section, we will present a novel trust model for wireless sensor networks named on-off attack mitigation for trust systems in WSNs (O^{2}Trust).

3.1 Overview

The design of O^{2}Trust is based on *penalty policy* that is based on misbehavior history. In O^{2}Trust, the evaluation model reflects nodes' real-time trust state accurately and is very sensitive to past malicious actions. This policy deals efficiency with the dynamic and contradictory misbehavior of malicious nodes. Dynamicity of the misbehavior is not considered under traditional trust estimation models because trust values are obtained based on current behavior, which does not indicate continuity of misbehavior. In other terms, only weight of measured misbehavior is considered rather than periodicity of the misbehavior along with weight of measured misbehavior.

Unlike the previous trust models, the trust value computation in our scheme is based on two components: reputation evaluation and penalty check (see Fig.1). Reputation evaluation is based on direct and/or indirect observations, and represents the accumulative assessment of the long-term behavior, while the penalty check is based on misbehavior history that represents how much a node has misbehaved in the past.

3.2 Trust value computation

The calculation of a trust value needs two parts of information: direct trust value and indirect trust value. Direct trust value can be obtained when a node has direct transactions with a node. Let $T_{i,j}$ denotes the trust value from node i to node j. It is defined in (1).

$$Ti,=\alpha\ DTi,j+(1-\alpha)\ ITi,j \quad (1)$$

where $DT_{i,j}$ is the direct trust value from node i to node j, $IT_{i,j}$ represents the indirect trust value of node j, α is the confidence factor and $0\leq \alpha \leq 1$.

A) Direct trust evaluation

To calculate the direct trust value, we consider two factors: the reputation rating and the penalty factor. Let $DT_{i,j}^{t}$ denotes the current direct trust value of node j from the view point of node i and $DT_{i,j}^{t-1}$denotes the past direct trust value. $Rep_{i,j}^{t}$ and $PF_{i,j}^{t}$ denote the current reputation rating and the penalty factor respectively. Therefore the trust value for node j at node i is:

$$DT_{i,j}^{t} = \begin{cases} \frac{DT_{i,j}^{t-1}+(1-PF_{i,j}^{t})}{2+PF_{i,j}^{t}} & if \ \ Rep_{i,j}^{t} = 0 \\ \frac{DT_{i,j}^{t-1}+(1-Rep_{i,j}^{t})}{2+PF_{i,j}^{t}} & otherwise \end{cases} \qquad (2)$$

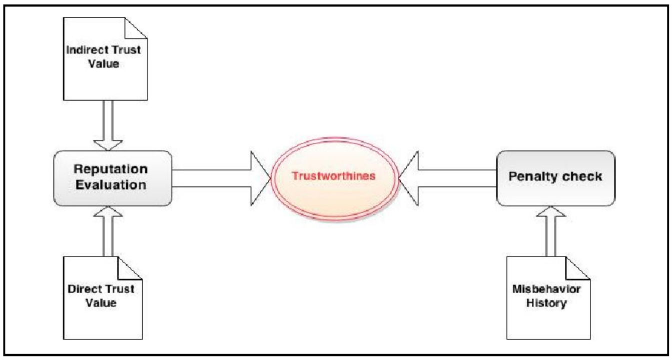

Fig. 1. Components of O^{2}Trust

If current reputation rating $Rep_{i,j}^{t}$ is equal to zero that means that the node j well-behaves at this moment, but there is no evidence that it is honest. To protect our trust model from on-off attacks, penalty factor that represents the misbehavior history, is used to calculate the current trust value.

In this paper, we can use one of the trust factors depending on the interactions between two neighbor nodes such as packet receive, send, delivery and consistency, to measure a node's reputation According to the quality of services provided by cooperating nodes, we classify interaction quality into two categories: successful (S) and unsuccessful (U).

In O^{2}Trust each sensor calculates individual trust values for only one-hop neighbors, contrary to GTMS [11] in which each sensor calculates individual trust values for all the cluster members. As a result, nodes do not keep trust information about every node in the network. Keeping neighborhood information implies significant lower energy consumption, less processing for trust computation, and less memory space.

Let $Rep_{i,j}^{t}$ denotes the current reputation rating which represents the current misbehavior of node j from the view point of node i at time t. It is defined in (3).

$$Rep_{i,j}^{t} = \frac{U_{i,j}}{U_{i,j}+S_{i,j}} \qquad (3)$$

$S_{i,j}$ denotes the total number of successful interactions of node i with j during a time period *t* and $U_{i,j}$ denotes the total number of unsuccessful interactions of node i with j during a time period *t*.

Due to the uncertainty of current reputation rating value based on recent interactions experience, we introduce the penalty factor to compute the trust value and to enhance the flexibility of our trust model. Penalty factor, accumulates measured misbehavior over time. It detects the dissimulated misbehavior. So, according to our proposed method if measured misbehavior is consistent, it is always greater than predefined threshold, and each time penalty factor will be increased until it reaches to maximum value (that is one). We define the penalty factor of node j estimated by node i as follow:

$$PF_{i,j}^{t} = \begin{cases} Min\{[Rep_{i,j}^{t} + (1-\theta) \times PF_{i,j}^{t-1}], 1\ \}, & if\ DT_{i,j}^{t} \geq THR1 \\ Min\{[Rep_{i,j}^{t} + \theta \times PF_{i,j}^{t-1}], 1\ \}, & otherwise \end{cases} \quad (4)$$

where θ is the forgetting factor for accumulated misbehavior, which ranges from $[0.5, 1]$ and $THR1$ is a threshold that can be tuned according to the system and security requirements.

Contrary to previous trust models, in which recent rating will carry more weight and therefore past misbehavior can be completely dissimulated, in our trust model we use an adaptive forgetting factor to improve on-off attack detection. According to Equation (4) once the node's trust value is under the trust threshold $THR1$, aging factors for previous accumulative misbehavior (penalty factor) will be different. In this case, we will weigh more on the penalty factor in order to more decrease the trust value. It means the malicious node that launches on-off attack, requires a longer time to recover its trust value once it has been defined as a malicious node.

B) Indirect trust value

The indirect trust value is computed based on the recommendations given by neighbors when it is often not possible for a node to directly assess the trust value of another node. However, the reliability of trust and reputation models could be easily compromised by various dishonest recommendation attacks, i.e., self-promoting, bad-mouthing and collusion.

To deal with the bad-mouthing attack and collusion attack, we use a lightweight averaging function to aggregate the indirect values. So, if node i needs a recommendation about node j, it will ask only trustworthy nodes (only one-hop neighbors) in unicast mode because it is more energy efficient than broadcast mode [13]. If the direct trust of a neighbor node is larger than the trust threshold value (for example 0.6), it is declared as trustworthy neighbor.

Let us assume that be the set of the trustworthy recommenders of the node j defined as:

$$\psi = \{\mathrm{DT}_{\mathrm{k,j}}, 0 \leq \mathrm{k} \leq \mathrm{M} - 1 \ \} \quad (5)$$

were M is the total number of recommenders and $DT_{k,j}$ is the direct trust from recommender k to node j. Then the indirect trust value of node j $IT_{i,j}$ can be defined as:

$$\mathrm{IT}_{\mathrm{i,j}} = \frac{1}{\mathrm{M}} \sum_{\substack{\mathrm{k}=0 \\ \mathrm{k} \neq \mathrm{j}}}^{\mathrm{M}-1} \mathrm{DT}_{\mathrm{k,j}} \quad (6)$$

In [13], Liang and Shi found that the lightweight average aggregation algorithm performs better than complex algorithms.

C) Decision making

After calculating the global trust value $T_{i,j}$ that relies on [-1, 1], each node i will classify trust into three states as follows:

$$Mp(T_{i,j}) = \begin{cases} T\!: trusted\ , & if \quad 1 \geq T_{i,j} \geq THR1 \\ U\!: uncertain, & if \ \ THR1 \geq T_{i,j} \geq THR2 \\ M\!: malicious, & if \ \ THR2 \geq T_{i,j} \geq -1 \end{cases} \quad (7)$$

where $THR2 < THR1 < 1$ and $THR1$, $THR2$ are a threshold that can be tuned according to the system and security requirements to determine the node's status. Since these values depend on network and security requirements, it will be set accordingly.

According to the trust state, each node can make a decision to cooperate or non-cooperate with the interacted node in the considered operation.

4 Performance evaluation

In this section, we present results of our simulations showing the effectiveness of our trust model. MATLAB software is used as simulation tool to assess the performance of our model. A comparative study between O^{2}Trust, *RFSN* [9] and *Retrust* [12] is given.

Concrete simulation scene is a square area of 100 m x 100 m, with 100 randomly deployed nodes. The communication radius is 25 m. An optimistic initialization strategy of trust value is adopted. So, the initial trust state of nodes is set as trusted (i.e., with initial trust value equal to 0.8).

Simulation is set up as follows. Each sensor node SN randomly selects one of its one-hop neighbors to transmit packets. Suppose that SN *i* ask SN *j* to forward packets, SN *i* can observe how many packets *j* has forwarded, i.e number of successful transactions. Next, SN i compute its direct trust value $\mathrm{DT}_{i,j}^{t}$ according to equation (2). We can summarize the simulation parameters in Table 1.

Parameter	Value	Description
$\alpha, (1-\alpha)$	(0.8, 0.2)	Weight ratio of direct and indirect value
θ	0.6	Forgetting factor
THR1	0.6 or 0.7	Trust threshold (for trusted nodes)
THR2	0,4	Trust threshold (for malicious nodes)
Initial trust value	0.8	The value assigned to a new node.

Table 1. Simulation parameters.

In on-off attack, strategic malicious nodes behave well and badly alternatively with the aim of remaining undetected while causing damage. Unfortunately, these malicious nodes may suddenly conduct attacks as they accumulate higher trust value. Thus, the attack cycle consists of two periods: on period and off period. When the attack is on, malicious node launches attacks; i.e. drops the received packets, and during the off period, performs well, i.e forwards received packets. Since the on period has an implication on the trust value of the malicious node, it will try to increase its trustworthiness during the off period.

4.1 Analysis of Penalty factor impact

In this section, we analyze the property of our trust model that combines penalty factor with reputation evaluation to derive trust value. We must demonstrate that the penalty factor helps to perceive the dissimulated misbehavior in on-off attack.

Our scheme has a feature whereby it continuously decreases the trust value of a malfunctioning or malicious node when it misbehaves in a repetitive manner. In order to validate the effectiveness of penalty factor and its influence on trust computation, we consider the actions of two types of nodes in the network: the benevolent nodes and the malicious ones. The benevolent nodes are the nodes that always behave well. While the malicious nodes are nodes that persistently misbehave.

The trust value's evolution of benevolent nodes and malicious nodes in O^2Trust is shown in fig. 2. In this experience, we calculate the average of trust values of fifty nodes of each type (benevolent and malicious). We can see that the trust value of the benevolent nodes in O^2Trust increases constantly. The factor penalty has no effect on the trust value since the behavior of trusted node is always good. However, the trust value's evolution of the malicious nodes decreases constantly as long as the malicious node persists in its misbehavior. We can see in the Fig.2, that in the first off period of attack (between 0 and 15 time units), the malicious node behaves well and its trust value follows the same evolution of the benevolent trust value. However, in the first on period (between 15 and 20 time units), it triggers the attack and its trust value falls off sharply. Consequently, its trust status changes from trusted to malicious in three time units. Since our proposed model always decreases the trust value of malicious node, the recovery rate in the off period is slower when the trust value is under the trust threshold. On the other hand, in the second on period (between 40 and 45 time units), the trust status changes from trusted to malicious in two time units. This can be explained by the fact that its last misbehavior is taken into account and as long as the malicious nodes repeat the on period, the penalty factor influences the trust value by checking the accumulated misbehavior in the past. So, it is difficult to the malicious node to recover its trust value in the off period, because the frequency of its past misbehavior is not discarded like in the previous trust models.

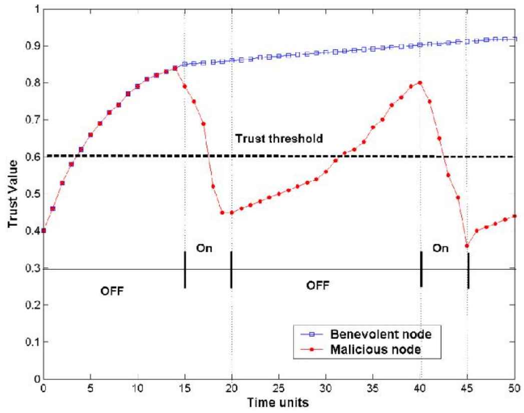

Fig. 2. Influence of penalty factor on trust computation.

Consequently, considering the penalty factor in trust computation can effectively make the trust model more sensitive to on-off attack.

4.2 On-off attack resilience

To evaluate how our trust model can mitigate on-off attack, we introduce the malicious detection rate metric called MDN that is defined as equation (8):

$$MDN = \frac{|D|}{|M|} \quad (8)$$

Where $|D|$ denotes the number of detected malicious nodes and $|M|$ denotes the number of total malicious nodes. It is typically used to evaluate the efficiency of a trust model.

Values of the system parameters such as trust threshold and forgetting factor, are selected based on heuristic and previously defined values in the literature [11, 14, 15, 16].

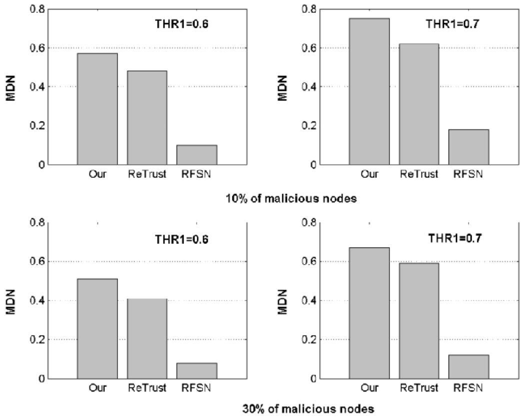

Fig. 3. Detection rate of on-off attack.

Fig. 3 depicts the detection rate of on-off attack under two trust threshold values: THR1=0.6 and THR1=0.7. For each trust threshold, we consider 10% and 30% of on-off attacker nodes among 100 nodes in the network. We can clearly see that our trust model outperforms ReTrust and RFSN. While considering 10% of malicious nodes, the detection rate in O^{2}Trust remains 57% and 75% with the trust threshold equal to 0.6 and 0.7 respectively. However, when the proportion of malicious nodes is equal to 30%, the detection rate of O^{2}Trust decreases quietly and remains 51% and 67% with the trust threshold equal to 0.6 and 0.7 respectively. This is a satisfactory detection rate in trust management. On the other hand, MDN of RFSN is very lower because it cannot efficiently deal with this kind of attack and cannot recognize malicious nodes sensitively since it focus on recent behavior. Therefore, the past misbehavior is discarded. We can also notice that when trust threshold is high, the on-off attack detection rate is also high. However, nodes might be assessed as untrustworthy even though they might not actually be malicious nodes.

We can conclude that O^{2}Trust is a fine-grained trust model that can portray unpredictable behaviors from malicious nodes and outperforms RFSN and ReTrust scheme. Consequently, on-off attack can be mitigated efficiently.

5 Conclusion

Trust systems are very useful mechanisms to thwart insider attacks. However, building a robust trust model is very challenging, because malicious nodes participate in the behavior rating process and can distort the trust value by cheating. In this paper, we proposed O^{2}Trust, a trust model to mitigate on-off attack. O^{2}Trust adopts the Penalty Policy against the misbehavior history of each node in the network. By considering misbehavior history, it is difficult to a malicious node to recover its trust value as long as it persists in its misbehavior. Simulation results show that O^{2}Trust is an efficient and on-off attack-resistant trust model. However, how to select the proper value of the weight and the defined threshold is still a challenge problem, which we plan to address in our future research endeavors.

References

1. Akyildiz I.F., Weilian Su. Sankarasubramaniam Y., Cayirci E.: A survey on sensor networks. IEEE Communications Magazine. Vol. 40, no. 8, pp. 102–14, (2002).
2. Krau C., Schneider M., Eckert C.: On handling insider attacks in wireless sensor networks. Information Security Technical Report. Vol. 13, no. 3, pp. 165–172, Elsevier, (2008).
3. Han G., Jiang J., Shu L., Niu J., Chao H.C.: Management and applications of trust in Wireless Sensor Networks: A survey. Journal of Computer and System Sciences. vol. 80, no. 3, pp. 602-617, Elseviers, (2014).
4. Labraoui N., Gueroui M., Aliouat M., Petit J.:Reactive and adaptive monitoring to secure aggregation in wireless sensor networks. Telecommunication Systems, vol. 54, no.1, pp. 3-17, Springer, Heidelberg , (2013).
5. Boukerche A., Ren y.: A trust-based security system for ubiquitous and pervasive computing environments. Computer Communications, Elseviers. Vol. 31, pp. 4343–4351, (2008).
6. Mármol F.G., Pérez G.M.: Providing trust in wireless sensor networks using a bio-inspired technique, Telecommunication Systems vol. 46, issue 2, pp 163-180, Springer, Heidelberg, (2010).
7. Lopez J., Roman R., Agudo I., Fernandez-Gago C.: Trust management systems for wireless sensor networks: Best Practices. Computer Communications. vol. 33 no. 9, pp. 1086-1093, Elseviers, (2010).
8. Alzaid H., Alfaraj M., Ries S., Jøsang A., Albabtain M., Abuhaimed A.: Reputation-based trust systems for wireless sensor networks: a comprehensive review. IFIP Advances in Information and Communication Technology, vol. 401, pp. 66-82, (2013).
9. Ganeriwal S., Srivastava M.: Reputation-based framework for high integrity sensor networks. ACM Transactions on Sensor Networks (TOSN). vol. 4, no. 3, (2008),
10. Chen H.: Task-based trust management for wireless sensor networks. International Journal of Security and Its Applications. Vol. 3, no. 2, pp. 21–26, (2009).
11. Shaikh R.A., Jameel H., d'Auriol B.J., Lee H., Lee S., Song Y.J.: Group-based trust management scheme for clustered wireless sensor networks. IEEE Transactions on Parallel and Distributed Systems. vol. 20, no. 11, pp. 1698-1712, (2009).
12. Daojing H., Chun C., Chan S., Bu J., Vasilakos A.V.: ReTrust: attack-resistant and lightweight trust management for medical sensor networks. IEEE Transactions on Information Technology in Biomedecine. vol. 16, no. 4, pp. 623-632, (2012).
13. Liang Z., Shi W.: Analysis of recommendations on trust inference in open environment, Performance Evaluation. vol. 65, no. 2, pp. 99-128, (2008).
14. Yu, H.; Shen, Z.; Miao, Ch.; Leung, C.; Niyato, D. Survey of trust and reputation management systems in wireless communications. In: Proceeding of IEEE, vol. 98, no. 10, pp. 1755–1772, (2010).

15. Bao, F., Chen, I.R., Chang, M.J., Cho, J.: Trust-Based Intrusion Detection in Wireless Sensor Networks. In: Proceedings of IEEE International Conference on Communications (ICC), pp. 1–6., (2011).
16. Sun, Y.L., Zhu, H., Liu K.J.R.: Defense of Trust management vulnerabilities in distributed networks. IEEE Communication Magazine.vol. 46, pp. 112–119, (2008).