HAJAR, M.S., AL-KADRI, M.O. and KALUTARAGE, H. 2020. LTMS: a lightweight trust management system for wireless medical sensor networks. In Wang, G., Ko, R., Bhuiyan, M.Z.A. and Pan, Y. (eds.). *Proceedings of 19th Institute of Electrical and Electronics Engineers (IEEE) Trust, security and privacy in computing and communication international conference 2020 (TrustCom 2020), 29 Dec 2020 - 1 Jan 2021, Guangzhou, China*. Piscataway: IEEE [online], pages 1783-1790. Available from: https://doi.org/10.1109/TrustCom50675.2020.00245

LTMS: a lightweight trust management system for wireless medical sensor networks.

HAJAR, M.S., AL-KADRI, M.O. and KALUTARAGE, H.

2020

LTMS: A Lightweight Trust Management System for Wireless Medical Sensor Networks

Muhammad Shadi Hajar, M. Omar Al-Kadri, Harsha Kalutarage
School of Computing
Robert Gordon University
Aberdeen, United Kingdom
{m.hajar, o.alkadri, h.kalutarage}@rgu.ac.uk

***Abstract*—Wireless Medical Sensor Networks (WMSNs) offer ubiquitous health applications that enhance patients' quality of life and support national health systems. Detecting internal attacks on WMSNs is still challenging since cryptographic measures can not protect from compromised or selfish sensor nodes. Establishing a trust relationship between sensor nodes is recognized as a promising measure to reinforce the overall security of Wireless Sensor Networks (WSNs). However, the existing trust schemes for WSNs are not necessarily fit for WMSNs due to their different operation, topology, resources limitations, and critical applications. In this paper, the aforementioned factors are regarded, and accordingly, two different methods to evaluate the trust value have been proposed to fit in-body, on-body, and off-body sensor nodes. Our Lightweight Trust Management System (LTMS) provides a further line of defense to detect packet drop attacks launched by compromised or selfish sensor nodes. Moreover, simulation results show that LTMS is more robust against complicated on-off attacks and can significantly reduce the processing overhead.**

***Index Terms*—Wireless Medical Sensor Networks (WMSNs), TMS, internal attacks, on-off attacks.**

I. Introduction

Wireless Medical Sensor Networks (WMSNs) offer a promising technology that has many potential healthcare applications ranging from monitoring the physiological body signals to actuation and drug delivery. Adopting such a revolutionized solution will ease the daily patient life, improve the performance of the overloaded medical staff, allow them to timely intervene, and reduce the expenses of the health systems.

Security concerns are still challenging obstacles to the widespread adoption of WMSNs. Internal security threats, such as packet drop attacks, may have catastrophic consequences. Compromised, selfish, or even faulty Sensor Nodes (SNs) may drop critical messages, such as urgent notifications of abnormal heart rhythms or insulin dose release orders, and consequently endanger the patient's life. This kind of attack can not be prevented by traditional cryptographic measures as malicious SNs are already authenticated and may have a copy of the security keys. Therefore, establishing a trust relationship between SNs within the network is regarded as a complementary security solution to the cryptographic measures to protect the network from malicious activities [1].

Trust Management Systems (TMSs) offer a further level of defense against internal attacks by monitoring other nodes' behavior. Various potential applications emerge from establishing a trust relationship between nodes ranging from routing [2] to defeating threats [3]. Several TMSs have been introduced in the literature for WSNs [3]–[6]; however, a limited number have been proposed to fit WMSNs [1].

In addition to the security concerns inherited from Wireless Sensor Networks (WSNs), WMSNs have unique characteristics that impose further challenges in adopting the existing security measures of WSNs. Therefore, operation requirements such as traffic rates, network topology, resources limitations, and intolerant applications must be taken into account in order to design an effective trust scheme that fits WMSNs. First, some SNs generate low traffic rates around 1 packet/s [7], such as heart rate sensors. Second, the network topology of WMSNs is a two-hop star topology in accordance with IEEE 802.15.6 standard [8]. Third, SNs, especially implanted ones, suffer from strict resources limitations. For instance, The battery is expected to last for years before getting replaced via surgery. Hence, a lightweight trust scheme is a must. Fourth, WMSNs provide very critical applications that can not tolerate any prolonged detection periods.

On the other hand, although TMSs show promising solution to detect packet drop attacks and other misbehaviors, they can be gamed by intelligent adversaries. TMSs are prone to on-off attacks, where adversary changes his behavior between good and bad alternately in order to redeem himself from the burden of bad behavior [9]. Moreover, adversaries can launch more sophisticated on-off attacks by changing the packet drop rates or launching on-off attacks with non-identical periods.

The main contribution of this paper is threefold. First, we clarify the unique requirements of WMSNs. Second, a novel lightweight and effective trust management scheme for in-body, on-body, and off-body SNs is proposed. Third, a comprehensive analysis is offered to show our scheme's merit in defending against complicated on-off attacks. Moreover, the code of our simulation and proposed methods, together with all experiments' data, are made available at (*https://github.com/mshsyr/LTMS*) for reproducibility purposes.

The remainder of this paper is organized into six sections as follows. Related works are given in section II. Section III overviews WMSNs. Our proposed scheme is presented in section IV, followed by the experiments simulation and analysis in section V. Finally, section VI concludes the paper.

II. Related Works

Trust and reputation systems emerge to defend against internal attacks. Various methods to model the trust relationship between nodes ranging from probability-based to fuzzy logic are proposed in the literature for both Mobile Ad hoc Network (MANET) and WSNs [3]–[5], [10]–[12]. However, few research have targeted WMSNs [1], [13].

Many research are put forward based on the Bayesian inference since the future behavior can be inferred based on historical observations. Different kinds of probability distributions are used for modeling in order to evaluate the trust value, such as beta distribution [3], [12], [14], binomial distribution [13], exponential distribution [11] and Gaussian distribution [15]. Although the probability theory offers a robust mathematical basis to model trust and reputation systems, it needs prolonged time to detect malicious activities since the trust value represents a long-term value [16]. To overcome this limitation, different approaches are adopted in the literature. Longevity factor, which gives more weight to recent observations, has been widely used in the literature to reflect the current behavior of the trustee [3], [11]–[14]. A sliding time window is also proposed in the literature to enhance the malicious detection rate [1], [5], [17]. However, the trust value in such schemes represents a short-term value limited to the length of the time window, which does not necessarily reflect the trustee's trustworthiness. Moreover, increasing the length of the time window requires more processing overhead. The punishment factor is another method to overcome the aforementioned issue. It has been widely adopted in the literature to give more weight to the bad behavior [10], [18], [19].

On the other hand, trust schemes proposed for either MANET or WSNs have to be further assessed in terms of WMSNs operating conditions, network topology, and resources limitations. Authors in [1] proposed ReTrust, which is a trust management scheme for WMSNs. According to the authors, ReTrust is a lightweight and attack resistant scheme that fits WMSNs. ReTrust adopts a sliding time window to update the trust value by using a dynamic exponential decreasing longevity factor in order to underweight old observations, which causes a significant processing overhead. Many trust schemes have used ReTrust as a benchmark scheme to contrast with [5], [10], [17], [19], [20]. BDTMS [13] is a trust management scheme for WSNs that targeting healthcare applications. It uses a longevity factor to reflect the recent behavior of the trustee. Neither ReTrust nor BDTMS considers the unique characteristics of WMSNs, such as traffic rates, or evaluates the processing overhead. However, these characteristics have been considered in developing our trust management scheme in order to fit WMSNs.

III. Wireless Medical Sensor Networks

In this section, an overview of WMSNs is introduced. SNs classifications and network topology are presented. Moreover, the WMSNs threat model is discussed.

A. Overview

A single WMSN may comprise up to hundreds of patients' Body Sensor Network (BSN). Each BSN consists of several SNs that sense the physiological signs and the activities of the body [21]. The maximum number of SNs within a single BSN is set to 64 in accordance with IEEE 802.15.6 standard [8]. These bio-sensor nodes are distributed in, on, or off the body. SNs have strict resources constraints, which play a significant role in adopting any security measure. Moreover, in-body SNs have their further power limitation as replacing the battery may require surgery. For instance, pacemakers' batteries, which is lithium iodide cells, are expected to last for around seven years before being replaced via surgery [22]. Fig 1 illustrates an exemplary WMSNs in a hospital. All sensed information is to be sent to the BSN's sink node, which in turn forwards this critical information to the medical server for processing. Authorized physicians are then able to access the patients' medical records and intervene if necessary.

B. Network Model

SNs are classified into three different types based on their role. Sink node is the gateway of the BSN to other BSNs or the internet. End SNs are designed to sense the body signals and exchange the messages with the sink if they are in direct communication or via relay SNs when they are out of the communication range. The topology of the BSN is a two-hop star topology as defined in IEEE 802.15.6 [8]. In this research, we differentiate between two types of SNs based on their resources limitations and traffic rates as follows.

- In-body SNs are end nodes implanted inside the human body to sense the vital signs of the body, such as pacemakers. They have minimal resources, and their power source is expected to last for years. They use low traffic rates around 1 packet per second [7]. These unique features of resources and traffic rates impose further requirements to deploy any proposed TMS.
- On-body and off-body SNs are distributed on the body surface or in the vicinity of the body. They have better resources and processing capabilities. Moreover, replacing nodes' batteries does not require surgical interventions. They usually use higher traffic rates as they are expected to relay messages for implanted SNs, for example.

This differentiation is used to propose two different trust evaluation methods.

C. Threat Model

BSN is prone to different security threats because of the sensitive data it generates and the broadcast nature of the wireless networks. Potential security threats are classified into external and internal. Protecting from external threats is achieved using cryptographic measures [4]. Internal attacks are usually launched by SNs that have passed the authentication process and may have had a copy of the security keys. These SNs are regarded as legitimate SNs from the cryptographic measures perspective. By monitoring the behavior of the SNs within the BSN, TMSs can defend against internal attacks,

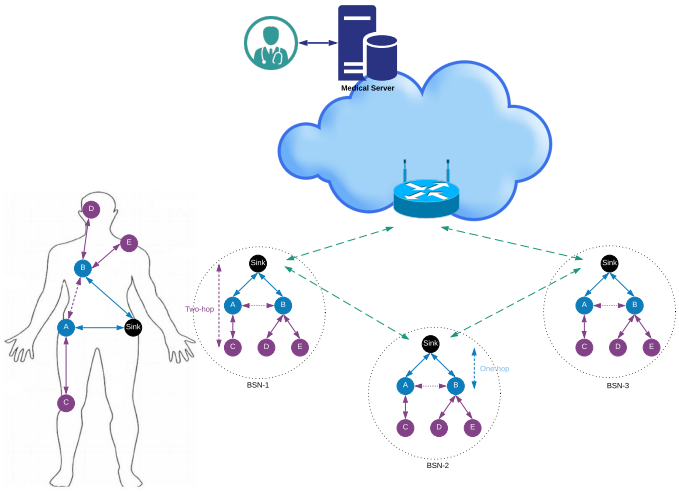


Fig. 1: Network Structure

such as packet drop attacks. Dropping packets is not just able to disrupt the network operation, but it may endanger the patient's life by dropping physician notification messages or drug delivery orders.

On the other hand, the process of evaluating the trust based on direct observations is vulnerable to on-off attacks, where the malicious node changes its behavior between malicious and benign alternately with a view to keep itself undetected [19]. The on-off attack cycle consists of on and off periods. Malicious agent behaves badly during the on period and well during the off period. Therefore, defending against this kind of attack requires a robust design.

IV. LTMS Evaluation Model

In this section, our proposed trust evaluation scheme for WMSNs is presented. Two methods are proposed to evaluate the trust value. The first is introduced for in-body SNs, while the second is for on-body and off-body SNs.

A. Definitions

The process of evaluating the trust value of nodes within the network is done in a distributed manner, where each node has its instance of the trust evaluation engine. As the trust relationship is established between two entities for a specific task, we refer to the party who performs the action as an *agent* and the party who monitors the agent and holds the trust value as a *subject*. The *action* could be any service provided by an agent to a specific subject, which is packet forwarding in our case. *Reputation* is defined as the perception that the agent does not have any intention to change its known behavior. Therefore, reputation value is inferred directly from the observation history. *Trust* is defined as having adequate confidence in the agent's future behavior. It is a subjective value as the subject may consider different factors to evaluate the trust value that are not necessarily related to the agent's honesty. In this context, we assume that the subject overhears the agent to observe forwarded packets, which considered good behavior and dropped packets, which considered bad behavior. These direct observations are used to evaluate direct trust in order to identify malicious agents. Reputation-based trust is defined as follows:

$$T_{ij}(t) = f(Rep_{ij}(t)) \tag{1}$$

where $T_{ij}(t)$ represents the trust value maintained by the subject i for the agent j at the time unit t and $Rep_{ij}(t)$ is the reputation value.

B. Beta Distribution based Trust Model

In this paper, we consider the packet forwarding service to evaluate the trust relationship between the subject and the agent as it is an essential service for multi-hop ad-hoc wireless networks. In this case, the subject maintains two time series $s(t)$ and $u(t)$ for successful and unsuccessful actions, respectively. The observed action has two states to represent if the packet is forwarded successfully or not. These observations are considered a sequence of trials with binary outcomes (Successful, Unsuccessful), which forming a binary space of disjoint elements. Therefore, this binomial Bayesian reputation system can be modeled using a Beta Probability Density Function (PDF) as follows.

$$f(p_x|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p_x^{\alpha-1}(1-p_x)^{\beta-1}$$
$$where \begin{cases} 0 \le p_x \le 1 \\ \alpha > 0 \\ \beta > 0 \end{cases} \tag{2}$$

There are two restrictions for Eq. 2. The first is $p_x \neq 0$ if $\alpha < 1$, and the second is $p_x \neq 1$ if $\beta < 1$. The reputation value is the expected value of Eq. 2 and is defined in Eq. 3.

$$\begin{aligned} Rep_{ij}(t) &= E(p_x) \\ &= \frac{\alpha_t}{\alpha_t + \beta_t} \end{aligned} \tag{3}$$

where $Rep_{ij}(t)$ represents the reputation value maintained by the subject i for the agent j, $E(p_x)$ is the expected value of the beta distribution, x represents the outcome of successful actions, α and β are the probability distribution function shape parameters or the levels.

The reputation value is updated by updating the beta distribution shape parameters α and β. To the best of our knowledge, all the probability distribution based TMSs use the same updating mechanism to update the reputation value by incorporating a longevity factor to give more weight to the current observations as shown in Eq. 4 and Eq. 5

$$\alpha_t = \lambda.\alpha_{t-1} + s(t) \tag{4}$$

$$\beta_t = \lambda.\beta_{t-1} + u(t) \tag{5}$$

where λ is the longevity factor and $0 \le \lambda \le 1$, $s(t)$ and $u(t)$ are the number of observations at the time unit t for

both successful and unsuccessful time series, respectively. The value of λ specifies the exponential decay of the observation history. Smaller values can adopt recent behavior change better than bigger ones; however, the observation history is forgotten quickly. Therefore, the values 0.8 and 0.9 are widely used in the literature for λ [5], [13].

C. The Proposed Method for In-Body SNs

The beta based reputation evaluation model provides a robust basis on the theory of statistics to evaluate the trust relationship between SNs [23]. However, the beta model, in its current form, fails to detect malicious behavior effectively. It needs more time to reflect any behavior change, which does not fit the critical applications of the WMSNs. Authors in [16] compare the effectiveness of beta based reputation model with hidden Markov models and report this drawback. This issue applies to other probability distribution based reputation models as they all use the same updating technique to update the reputation value. The traditional updating mechanism uses a single weight exponential smoothing technique, which fails to reflect any sudden malicious behavior fast because the evaluated reputation value represents the long-term expected value of the probability distribution. Therefore, it needs more time to detect any malicious behavior [9]. This drawback may be exploited by a smart adversary to launch complicated attacks such as on-off attacks.

In [9], we have introduced a novel updating mechanism to allow fast detection of any behavior change. Although the proposed method shows prompt reaction to any sudden behavior change, smart adversaries can take advantage of the model dynamicity to launch complicated on-off attacks. Therefore, our proposed method adopts the asymmetry principle of trust, which considers the trust as a fragile thing that is hard to earn, but easy to lose [24]. Adopting this technique can defend against on-off attacks and make the response to any malicious activity faster. We update the beta levels by incorporating the current slopes b_t and d_t of the successful and unsuccessful time series, respectively. Taking into account that the difference between two subsequent time units is always one, Eq. 6 and Eq. 7 show how slopes are computed.

$$b_t = \omega(\alpha_t - \alpha_{t-1}) + (1-\omega)b_{t-1} \tag{6}$$

$$d_t = \omega(\beta_t - \beta_{t-1}) + (1-\omega)d_{t-1} \tag{7}$$

where b_t and d_t are the slopes at the time unit t, ω is the weighting coefficient and $0 \leq \omega \leq 1$. The smoothing coefficient ω impacts the detection response speed, which is maximized when $\omega = 1$, and this means it just depends on the current change of beta levels. As incorporating the slopes into the updating mechanism can reflect any sudden change in behavior, which makes earning and losing trust identical, algorithm 1 deals with this concept to make the trust value easy to lose and hard to earn.

During the attack, the slope b_t maintains negative values; hence, the level α_t may accumulate negative values depending on the duration of the attack. At the same time, the level β_t, which refers to the malicious activities, develops over the attack duration with a view to make forgetting the bad behavior harder. Therefore, the trust value is evaluated using Eq. 8.

Algorithm 1: Updating mechanism

Input: Observations & beta shape parameters at t and $t-1$
Output: Updated shape parameters
initialization;
while *true* **do**
  if $b_{t-1} \leq 0$ && $d_{t-1} > 0$ **then**
    $\alpha_t = \lambda(\alpha_{t-1} + b_{t-1}) + s(t)$;
    $\beta_t = \lambda(\beta_{t-1} + d_{t-1}) + u(t)$;
    $b_t = \alpha_t - \alpha_{t-1}$;
    $d_t = \beta_t - \beta_{t-1}$;
  else
    $\alpha_t = \lambda.\alpha_{t-1} + s(t)$;
    $\beta_t = \lambda.\beta_{t-1} + u_(t)$;
    $b_t = \alpha_t - \alpha_{t-1}$;
    $d_t = \beta_t - \beta_{t-1}$;
  end
end

$$T_{ij}(t) = \begin{cases} \frac{\alpha_t}{\alpha_t+\beta_t} & for\ \ \alpha_t > 0 \\ 0 & otherwise \end{cases} \tag{8}$$

D. The Proposed Method for On-Body and Off-Body SNs

In this subsection, we propose our method to evaluate the trust value for on-body and off-body SNs. These SNs still have resources limitations; however, the processing capabilities are higher than the implanted ones. More importantly, replacing batteries of on-body and off-body SNs does not require surgical intervention. Therefore, a further level of protection to defend against on-off attacks is introduced to enhance the overall security.

The reputation value evaluated using beta models represents a long-term value. It reflects the accumulated long observation history. Although this feature is useful to assess the trustworthiness of the SNs, it can be exploited by adversaries to launch sophisticated attacks. Many trust management schemes in the literature adopt the sliding time window technique in order to address this security concern [1], [5], [17]. Adopting a sliding time window has some limitations as the trust value reflects only the length of the time window; moreover, it requires more processing each time the trust value is computed.

Our proposed method for on-body and off-body SNs incorporates the short-term and long-term reputation values along with our proposed updating mechanism in order to defend against on-off attacks. This protection module is only triggered when an on-off behavior is detected. The first cycle of the on-off attack is considered as just a malicious activity because the on-off attack is a repeated malicious activity that can only be detected from the second cycle. Therefore, if the same behavior reoccurs, the on-off module is triggered to defend

against on-off attacks. The detailed process is shown in the algorithm 2.

Algorithm 2: Trust evaluation for on-body and off-body SN

Input: Updated beta shape parameters & $Rep_{ij}(t-1)$
Output: Trust value
initialization;
while *true* **do**
  if $\alpha_t \leq 0$ **then**
    $Rep_{ij}(t) = 0$;
  else
    $Rep_{ij}(t) = \frac{\alpha_t}{\alpha_t+\beta_t}$;
  end
  if $Rep_{ij}(t-1) \geq thr_1$ && $Rep_{ij}(t) < thr_1$ **then**
    if $malicious > 0$ **then**
      $cycle = t - malicious$;
      $malicious = 0$;
    else
      $malicious = t$;
    end
  end
  if $cycle > 0$ && $Trust(t-1) < thr_2$ **then**
    $ShRep_{ij}(t) = mean(Rep_{ij}(t - period : t)$;
    $Trust_{ij}(t) = min(ShRep_{ij}(t), Rep_{ij}(t))$;
  else
    $Trust_{ij}(t) = Rep_{ij}(t)$;
    $cycle = 0$;
  end
end

where $thr1$ represents the threshold to differentiate between malicious and benign SNs, which is usually set to 0.5 in the literature [5], [10]–[13], $thr2$ represents the expected trustworthiness that the SNs have in normal operation, $ShRep_{ij}(t)$ represents the short-term reputation value at the time unit t, and $cycle$ and $malicious$ are two variables to differentiate between sudden misbehavior and on-off attacks.

V. Simulation and Analysis

In this section, our proposed trust management scheme for WMSNs is simulated and analyzed. The simulator NS-3.30 [25] is used to run the simulation scenarios. All SNs have the ability to forward packets, while one of them acts as a sink. AODV routing protocol [26] is installed in each SN to relay packets to the sink. However, it has been modified to simulate malicious activities by introducing new attributes to launch packet drop attacks, which will be detailed in the next paragraph. Traffic is generated based on the exponential distribution using the parameterized probability density function shown in Eq. 9

$$p(x;b) = \begin{cases} \mu e^{-\mu x} & x \geq 0 \\ 0 & x < 0 \end{cases} \quad x \in [0,b] \tag{9}$$

where μ is the rate parameter, and b is the bound parameter. As the exponential distribution is theoretically unbounded, the bound b is defined to make the generated values bounded over the interval $[0, b]$.

During the simulation, benign SNs drop received packets from others with a drop ratio of 10%, whereas the packet drop ratio of malicious SNs is 75% unless otherwise indicated. The decision to forward or drop a packet is taken randomly for each received packet individually. The simulation consists of two phases. In the first phase, the network is initialized, and the malicious SNs behave well to increase their trustworthiness. During the second phase, they launch on-off attacks in order to disrupt the network operation and keep themselves undetected. The first phase of the simulation is $50s$, while the second phase is $150s$ unless otherwise indicated. The subject continues to interact with malicious agents even after they are detected in order to study the behavior of the trust schemes under on-off attacks as the trust value is developing over time. The on and off periods are identical during the simulation unless otherwise indicated, and the time unit is set to 1 second.

A. Security and Efficiency Analysis

In this subsection, we present the efficiency analysis of our two methods to evaluate the trust value. We have assessed the robustness of the direct trust evaluation methods of a set of trust management schemes proposed for WSNs [4], [5], [11], [12] and WMSNs [1], [13] under on-off attacks. Both ReTrust [1] and RaRTrust [5] show good performance in defending against on-off attacks. This performance can be attributed to adopting the sliding time window technique, which can only reflect the recent behavior of SNs. Therefore, both ReTrust and RaRTrust have been chosen to contrast our scheme with. Table I shows the parameters of each scheme. It is worth mentioning that we adopt the same parameters values as declared in their publications as they reflect the best performance. Moreover, the longevity factor λ for our scheme is set to equal or higher than other schemes as smaller values enhance the detection performance of trust management schemes and may make the comparison unfair. The exponential slope weight ω is set to 1 as discussed earlier. The expected trustworthiness parameter thr_2 is set to 0.85, indicating that benign nodes have a trust value between 0.85 and 1. On the other hand, table II shows the simulation parameters, which have been chosen with a view to represent the environment of BSN. We use nine SNs to build a BSN as illustrated in Fig. 1, which are sufficient to reflect the behavior of the TMS. Packet size is set to 264B in accordance with IEEE 802.15.6 standard.

TABLE I: Trust Schemes Parameters

Scheme	Parameters
ReTrust [1]	ϕ=0.9, Time Window (TW)=6 time units
RaRTrust [5]	λ=0.8, TW=6 time units
LTMS	λ=0.9, $\omega = 1$, $thr_2 = 0.85$

TABLE II: Simulation Parameters

Parameter	Value
Application	Poisson random traffic
Exponential transmission interval μ	1, 2, 10, 100
Packet size	264B
Routing Protocol	AODV (modified version)
Radio Range	1m
Propagation delay model	Constant speed propagation delay
Propagation loss model	Range propagation loss
Number of SN	9
Time unit	1s
Simulation Time	200s, 400s

1) Trust Evaluation for In-Body SNs: In-body SNs have very tough resources limitations. They are designed to perform a specific function. For instance, a patient who has a heart problem may have an Implantable Cardioverter Defibrillator (ICD) to monitor his/her heart rate and treat any abnormal heart rhythms. Similarly, an implanted insulin pump monitors and controls the blood sugar level. These functions generate low traffic rates. Monitoring the heart rate, for example, generates a traffic rate of around 1 packet/s [7]. Therefore, existing trust management schemes must be assessed under low traffic rates. Fig. 2 illustrates how the trust value is developing under on-off attacks for a low traffic rate. The on-off cycle is set to 10 time units, and the traffic rate μ is set to 1 and generated exponentially. Algorithm 1, referred to as LTMS(1) is contrasted with ReTrust [1] and RaRTrust [5]. Results show that both RaRTrust and ReTrust struggles to work properly under low traffic rates. In RaRTrust, the trust evaluation process fails most of the time as just a few points appear in the figure because the first step of the trust evaluation is calculating the forwarding ratio at the current time unit, which fails due to the lack of observations at certain time units. Although ReTrust is able to evaluate the trust value during the simulation time, it fails to reflect the good behavior during the first phase when no attack is occurring. In the second phase, the trust value fluctuates around the threshold without being able to reflect the bad behavior. On the other hand, our proposed algorithm LTMS(1) can reflect the actual trust value when the agent is behaving well; furthermore, when the attack is running, it shows a quick response, and it can keep the trust value under the threshold most of the time.

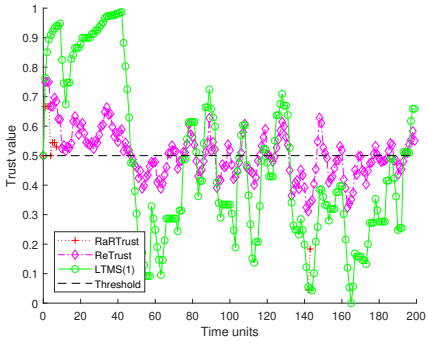


Fig. 2: Trust evaluation for in-body SNs

2) Trust Evaluation for On-Body and Off-Body SNs: In this experiment, we evaluate our proposed method for on-body and off-body SNs. The simulation is run for 400s, where an on-off attack is launched after the first phase. The on and off periods of the attack are set to 30 time units, and the traffic rate μ is set to 100. After three consecutive cycles, the attack is paused to study the behavior of acquiring trust, then the attack resumes. Fig. 3 illustrates how trust value is developing under the attack. Both ReTrust and RaRTrust demonstrate similar behavior. Both lose and earn trust easily, which causes fluctuations around the threshold during the on-off attack. On the other hand, our method LTMS(2) demonstrates an outperforming behavior. It loses the trust quickly when malicious activity is detected, while it makes the trust harder to earn in the off period. The first cycle of the on-off attack is regarded as just malicious behavior. Therefore, from the second cycle on, LTMS(2) makes earning trust during the off period harder as shown between the time units 200 and 250. Moreover, if the attack reoccurs, it maintains the same behavior.

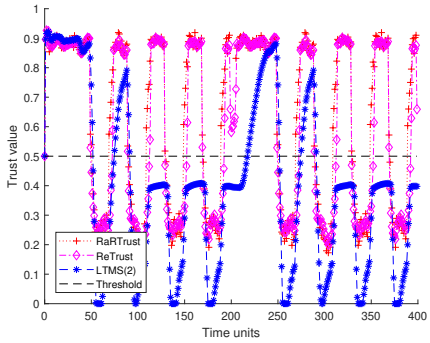


Fig. 3: Trust evaluation for on-body and off-body SNs

B. Performance Analysis

In this subsection, we compare the average processing time consumed by each of the trust management schemes using the MATLAB platform. The test is carried out on Intel Core i5-8500T processor at 2.1GHz and 8GB RAM using the data sets generated by our simulation scenarios. Fig. 4 illustrates the average processing time of ReTrust, RaRTrust, LTMS(1) and LTMS(2). RaRTrust consumes the highest average processing time of $3.2 \times 10^{-4}s$, while ReTrust shows a better processing overhead compared with RaRTrust as it consumes around $2.74 \times 10^{-4}s$. On the other hand, our method LTMS(1) for in-body SNs consumes the lowest processing time among all trust schemes. It consumes $0.85 \times 10^{-4}s$, which saves around 73% and 69% of the processing time of RaRTrust and ReTrust, respectively. Moreover, our method for on-body and off-body SNs consumes around $1.7 \times 10^{-4}s$ average processing time, which saves around 47% and 38% of the processing time of RaRTrust and ReTrust, respectively.

C. On-Off Performance Metric

In order to assess the effectiveness of trust management schemes under on-off attacks, we introduce the on-off Attack

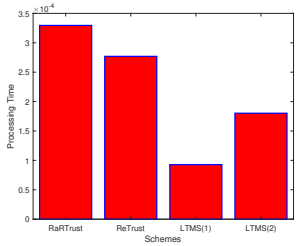


Fig. 4: The average processing time

Detection Metric (ADM). ADM is defined in Eq. 10 as the ratio of the detection time to the on-off attack time.

$$ADM = \frac{|DT|}{|AT|} \tag{10}$$

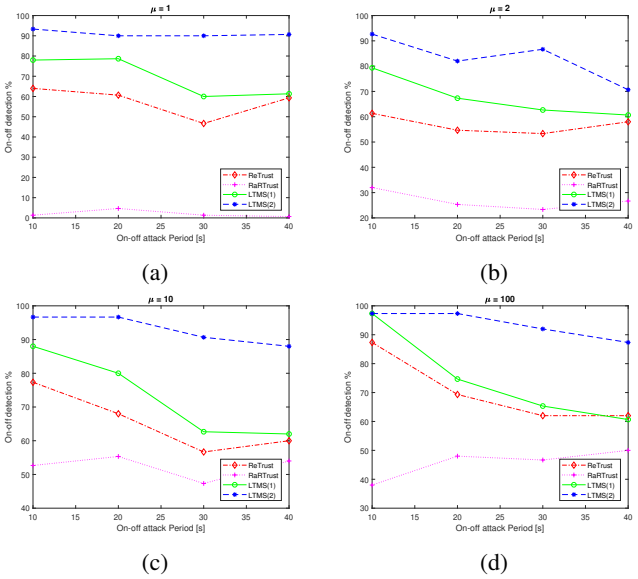
where $|DT|$ denotes to the number of the time units when the attack is detected, and $|AT|$ denotes the total number of time units of the on-off attack.

This metric is able to reflect the robustness of the trust management schemes under on-off attacks. The performance of our proposed methods will be evaluated in the following different scenarios.

1) Variable Traffic Rates: Two sets of traffic rates are chosen. The first contains low traffic rates ($\mu = 1$, $\mu = 2$), which represents the traffic of the in-body SNs, whereas the second set ($\mu = 10$, $\mu = 100$) is chosen for medium and high traffic rates. Fig. 5a-5d show the detection rate performance of the aforementioned schemes for different traffic rates and different on-off attacks cycles.

For low traffic rates, RaRTrust shows the lowest detection rates for all on-off attack cycles. It struggles to detect the on-off attacks with a detection rate of around 0 for $\mu = 1$, while ReTrust shows better performance compared with RaRTrust. It detects around 65% of the on-off attacks when the on-off cycle is 10 time units. By increasing the duration of the on-off attack cycle, the performance of ReTrust decreases to around 60%. On the other hand, our proposed methods demonstrate superior performance compared with ReTrust and RaRTrust with detection rates up to 80% and 93% for LTMS(1) and LTMS(2), respectively.

For medium and high traffic rates, RaRTrust starts to defend against on-off attacks with a detection rate of around 50% for $\mu = 10$ and around 40% for $\mu = 100$. In contrast with RaRTrust, ReTrust shows better performance with a detection rate starts at around 76% and decreases to 60% for $\mu = 10$, and starts at around 86% and decreases to reach 62% for $\mu = 100$. On the other hand, our method LTMS(2) shows the best performance in detecting the on-off attacks. For $\mu = 100$, it starts at just below 98% for the on-off cycle 10 time units and reaches around 88% for the on-off cycle 40 time units. For $\mu = 10$, it starts at just below 97% and reaches 80% for the on-off cycle 40 time units. It is worth mentioning that our lightweight method for in-body SNs LTMS(1) shows better results than both ReTrust and RaRTrust in detecting on-off attacks for medium and high traffic rates with a significantly lower processing overhead.

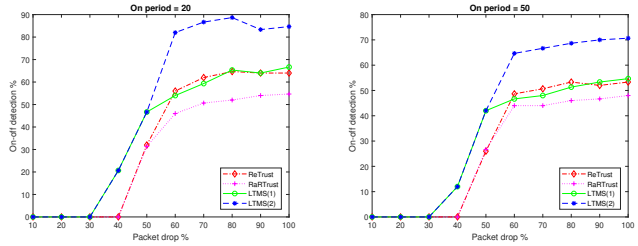


(a) (b) (c) (d)

Fig. 5: The detection performance for variable traffic rates

2) Variable Drop Rates: In this experiment, we evaluate the attack detection performance for different packet drop rates. The drop rate varies from 10% to 100% during the on period instead of the previous fixed drop rate of 75%. Fig. 6a and 6b illustrate the detection rates for two different on periods 20 and 50 time units, where the traffic rate μ is set to 100. ReTrust and RaRTrust detect attacks starting from the drop rate of 40%. Between 40% and 50%, ReTrust and RaRTrust show identical results, then ReTrust overcomes RARTrust in the detection rate.

On the other hand, our proposed methods are able to detect attacks starting from 30% drop rate. LTMS(2) shows superior performance comparatively, while LTMS(1) shows a close performance to ReTrust between 50% and 100%.



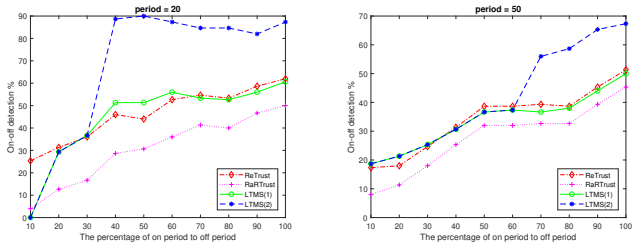
(a) (b)

Fig. 6: The detection performance for variable drop rates

3) Non-Identical Periods: In this experiment, more sophisticated on-off attacks are launched by varying the on and off periods. Obviously, it is harder to detect on-off attacks when the on period is less than the off period. Hence, the on period is set to be a ratio of the off period ranging from 10% to 100%. The traffic rate μ is set to 100. Two on periods 20 and

50 time units are used to evaluate the performance. Fig. 7a and 7b show the detection performance of the aforementioned trust schemes when the on period is 20 and 50 time units. For on period equals to 10% of the off period, both ReTrust and RaRTrust show detection rate greater than 0. However, from our point of view, two sequential time units of bad behavior are not enough to destroy the earned trust. ReTrust and RaRTrust adopt a sliding time window to calculate the trust; meaning, they adopt the most recent changes regardless of the history of the agent as illustrated before in Fig. 2 and 3. By increasing the ratio of the on period, the performance of both ReTrust and RaRTrust is enhanced for both on periods; however, ReTrust shows better performance.

On the other hand, LTMS(1) and LTMS(2) detect attacks starting from 20% and 10% for the on periods 20 and 50, respectively. LTMS(1) shows a close performance to ReTrust. However, LTMS(2) shows prominent performance comparatively after 30% and 60% for the on period 20 and 50, respectively.



(a) (b)

Fig. 7: The detection performance for different on-off ratios

VI. Conclusion and Future Work

Security concerns prevent the widespread adoption of the WMSNs advancements. Trust management provides significant means to reinforce the security of WMSNs. In this paper, we propose a trust evaluation model for WMSNs. Our proposed scheme uses a novel updating and evaluating mechanisms. LTMS is a lightweight and attack-resistant trust scheme for in-body, on-body, and off-body SNs. The experimental results show that LTMS outperforms the state of the art trust management schemes while preserving resources, making it a suitable candidate to meet WMSNs security requirements. In the future, LTMS will be developed to incorporate recommendations from SNs in the vicinity securely.

References

[1] D. He, C. Chen, S. Chan, J. Bu, and A. V. Vasilakos, "Retrust: Attack-resistant and lightweight trust management for medical sensor networks," *IEEE transactions on information technology in biomedicine*, vol. 16, no. 4, pp. 623–632, 2012.

[2] Y. Stelios, N. Papayanoulas, P. Trakadas, S. Maniatis, H. C. Leligou, and T. Zahariadis, "A distributed energy-aware trust management system for secure routing in wireless sensor networks," in *International Conference on Mobile Lightweight Wireless Systems*. Springer, 2009, pp. 85–92.

[3] W. Fang, W. Zhang, Y. Yang, Y. Liu, and W. Chen, "A resilient trust management scheme for defending against reputation time-varying attacks based on beta distribution," *Science China Information Sciences*, vol. 60, no. 4, p. 040305, 2017.

[4] V. B. Reddy, S. Venkataraman, and A. Negi, "Communication and data trust for wireless sensor networks using d–s theory," *IEEE Sensors Journal*, vol. 17, no. 12, pp. 3921–3929, 2017.

[5] N. Labraoui, M. Gueroui, and L. Sekhri, "A risk-aware reputation-based trust management in wireless sensor networks," *Wireless Personal Communications*, vol. 87, no. 3, pp. 1037–1055, 2016.

[6] Z. Yao, D. Kim, and Y. Doh, "Plus: Parameterized and localized trust management scheme for sensor networks security," in *2006 IEEE International Conference on Mobile Ad Hoc and Sensor Systems*. IEEE, 2006, pp. 437–446.

[7] M. N. Islam and M. R. Yuce, "Review of medical implant communication system (mics) band and network," *Ict Express*, vol. 2, no. 4, pp. 188–194, 2016.

[8] IEEE, "Ieee standard for local and metropolitan area networks - part 15.6: Wireless body area networks," *IEEE Std 802.15.6-2012*, pp. 1–271, Feb 2012.

[9] M. S. Hajar, M. O. Al-Kadri, and H. Kalutarage, "Etaree: An effective trend-aware reputation evaluation engine for wireless medical sensor networks," in *2020 IEEE Conference on Communications and Network Security (CNS)*, 2020, pp. 1–9.

[10] N. Labraoui, "A reliable trust management scheme in wireless sensor networks," in *2015 12th International Symposium on Programming and Systems (ISPS)*. IEEE, 2015, pp. 1–6.

[11] J. Zhao, J. Huang, and N. Xiong, "An effective exponential-based trust and reputation evaluation system in wireless sensor networks," *IEEE Access*, vol. 7, pp. 33 859–33 869, 2019.

[12] S. Ganeriwal and M. B. Srivastava, "Reputation-based framework for high integrity sensor networks," in *Proceedings of the 2Nd ACM Workshop on Security of Ad Hoc and Sensor Networks*. ACM, 2004, pp. 66–77.

[13] W. Fang, C. Zhu, W. Chen, W. Zhang, and J. J. Rodrigues, "Bdtms: Binomial distribution-based trust management scheme for healthcare-oriented wireless sensor network," in *2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC)*. IEEE, 2018, pp. 382–387.

[14] W. Fang, C. Zhang, Z. Shi, Q. Zhao, and L. Shan, "Btres: Beta-based trust and reputation evaluation system for wireless sensor networks," *Journal of Nework and Computer Applications*, vol. 59, pp. 88–94, 2016.

[15] W. Fang, W. Zhang, Y. Liu, W. Yang, and Z. Gao, "Btds: Bayesian-based trust decision scheme for intelligent connected vehicles in vanets," *Transactions on Emerging Telecommunications Technologies*, 2020.

[16] M. E. Moe, B. E. Helvik, and S. J. Knapskog, "Comparison of the beta and the hidden markov models of trust in dynamic environments," in *IFIP International Conference on Trust Management*. Springer, 2009, pp. 283–297.

[17] F. Ishmanov, S. W. Kim, and S. Y. Nam, "A robust trust establishment scheme for wireless sensor networks," *Sensors*, vol. 15, no. 3, pp. 7040–7061, 2015.

[18] Z. Wei, H. Tang, F. R. Yu, M. Wang, and P. Mason, "Security enhancements for mobile ad hoc networks with trust management using uncertain reasoning," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, pp. 4647–4658, 2014.

[19] N. Labraoui, M. Gueroui, and L. Sekhri, "On-off attacks mitigation against trust systems in wireless sensor networks," in *IFIP International Conference on Computer Science and its Applications*. Springer, 2015, pp. 406–415.

[20] F. Ishmanov, S. W. Kim, and S. Y. Nam, "A secure trust establishment scheme for wireless sensor networks," *Sensors*, vol. 14, no. 1, pp. 1877–1897, 2014.

[21] M. Chen, S. Gonzalez, A. Vasilakos, H. Cao, and V. C. Leung, "Body area networks: A survey," *Mobile networks and applications*, vol. 16, no. 2, pp. 171–193, 2011.

[22] F. Morady, "Electrophysiologic interventional procedures and surgery," in *Goldman's Cecil Medicine*. Elsevier, 2012, pp. 369–373.

[23] A. Josang and R. Ismail, "The beta reputation system," in *Proceedings of the 15th bled electronic commerce conference*, 2002, pp. 2502–2511.

[24] W. Poortinga and N. F. Pidgeon, "Trust, the asymmetry principle, and the role of prior beliefs," *Risk Analysis: An International Journal*, vol. 24, no. 6, pp. 1475–1486, 2004.

[25] Open source, "Ns-3 a discrete-event network simulator for internet systems," accessed: 01-04-2020. [Online]. Available: https://www.nsnam.org/releases/

[26] S. Das, C. Perkins, and E. Royer, "Ad hoc on demand distance vector (aodv) routing," *IETF RFC3561, July*, 2003.