ITFBS: adaptive intrusion-tolerant scheme for body sensor networks in smart space applications

G. Wu  J. Ren  L. Yao  Z. Xu
School of Software, Dalian University of Technology, Dalian 116620, People’s Republic of China
E-mail: yaolin@dlut.edu.cn

Abstract: As an important part of the smart space, body sensor networks (BSNs) provide continuous health monitoring and automation assistance for smart environment residents. A high degree of security and reliability for BSN is extremely required. An adaptive and flexible intrusion-tolerant scheme for BSN, namely ITFBS, is proposed. ITFBS dynamically detects intrusions according to the collected intrusion-related information, and it can provide an adaptive intrusion-tolerant strategy with passive replication by utilising two-step threshold-based intrusion detection and replicas classification. The correctness and effectiveness of ITFBS is theoretically proved, and the experimental results show that ITFBS can effectively tolerate intrusions with low power consumption and high adaptability.

1 Introduction

Recent advances in hardware engineering are now providing us with small, yet powerful networking computing devices. When many of the physical things surrounding us in our own environment are being equipped with these devices, we can exploit them for aiding us in our daily work. Hence our environment will become intelligent. Our homes and offices will become smart spaces. The broad vision of a smart spaces system incorporates ideas found in ubiquitous computing, wearable computing, intelligent environments and cooperative buildings. As an important part of the smart space, body sensor networks (BSNs) can be employed for valuable functions such as remote health monitoring, automation assistance and activity recognition [1–5]. By outfitting patients with various wireless implantable or wearable vital sign sensors, detailed real-time data on physiological status can be continuously sampled. Although BSNs share many of the same challenges and opportunities with general wireless sensor networks (WSNs), many BSN-specific challenges have emerged.

In BSN, high degrees of security and reliability are demanded for most pervasive healthcare applications [6, 7], however, BSN has fewer and smaller nodes compared with the conventional WSN. Smaller nodes imply smaller batteries, creating strict constrains on the energy consumed by processing, storage and communication resources. Usually, BSN is vulnerable to fail when intrusions occur due to the characteristics of energy-sensitiveness and resource-limitedness. In addition, there is almost no underlying security infrastructure for BSN. Therefore it has become urgent to provide an adaptive intrusion-tolerant scheme with low power consumption for BSN in smart space applications.

In this paper, we present an adaptive and flexible intrusion-tolerant scheme for BSN, namely ITFBS. ITFBS dynamically monitors the condition of the sensor in the sensor group. Through two-step threshold-based intrusion detection and replicas classification, ITFBS provides an adaptive intrusion-tolerant strategy with passive replication. The correctness and effectiveness of ITFBS is theoretically proved and simulations have been carried out to evaluate the performance of the proposed scheme.

The remainder of this paper is organised as follows. Section 2 introduces related work. In Section 3, the related variables used in the ITFBS scheme and its application scenario are given. Section 4 presents the ITFBS scheme in detail. The feasibility of ITFBS is proved in Section 5. In Section 6, the performance of ITFBS is evaluated. Finally, we conclude the paper and outline some future work in Section 7.

2 Related work

In the last few years, sensor networks have attracted a lot of attention from researchers in terms of security and reliability of the system. Extensive research has been focusing on the intrusion tolerance. The most common intrusion tolerance solutions are intrusion detection and redundancy.

In the intrusion-tolerant system, intrusion detection is the major phase. There are three different detection policies: misuse or signature-based detection [8], anomaly-based detection [9–12] and specification-based detection [13]. Ahmed et al. [8] propose a novel distributed abnormal node detection technique, which uses both signature and anomaly-based routing methodology to identify the compromise node. Gupta et al. [9] present a centralised anomaly detection mechanism for detecting fail-stop
failures and routing anomalies by the collection of information and detection. Another distributed anomaly detection mechanism [10] uses a clustering algorithm to build a model of normal traffic behaviour, and then uses this model to detect abnormal traffic patterns. An adaptive intrusion detection strategy based on artificial immune is presented in [11]. In [12], the authors propose an anomaly-based distributed centralised detection mechanism to analyse the behaviour of nodes based on cumulative summation (CUSUM) [14]. In [13] the authors propose a specification-based mechanism to identify characteristic signs of sinkhole attack by incorporating inter-agent communication and distributed computing in decision making.

For the recovery techniques with redundancy, there exist mainly two major approaches in replication: active replication [15–17] and passive replication [18, 19]. Active replication means that all requests are processed by all replicas. It ensures a fast response to failures. However, it needs heavy processing and requires the processing of requests to be deterministic. With passive replication, only one replica processes the request, and sends updating messages to the other replicas. Less resource is required in passive replication scheme compared to active replication scheme. However, longer response time is required in case of failure for passive replication scheme. Considering the limited resources and low computation of sensors, passive replication scheme is more suitable for WSN application. LEACH [18] utilises the passive replication scheme to select a node as the new cluster head for their neighbours with a probabilistic algorithm when a cluster head node fails. In [19], a cluster reallocation algorithm is proposed to overcome the failure of the cluster head.

Although the existing schemes stated above play important roles to improve the reliability of BSN, the existing intrusion-tolerant approaches in WSN mainly focus on the detection of intrusion, whereas the intrusion recovery is ignored. In addition, there is a significant gap between WSN and BSN due to the characteristics of BSN and the requirements of medical applications, thus none of these intrusion-tolerant approaches in WSN can be directly applied in BSN. Currently, the reliability research of BSN mostly concerns with the communication links [20–24], but failures caused by the intrusion are not considered. In this paper, an adaptive and flexible intrusion-tolerant scheme for BSN is demonstrated. Major differences between this work and the aforementioned schemes include:

1. In order to deal with intrusions, proactive intrusion-tolerant strategy with passive replication is adopted. Intrusions can be predicted based on pre-collected intrusion-related information. When an intrusion might occur, a new primary replica will be activated to compensate for the intrusion even before it occurs. This method not only improves the intrusion response time, but also prolongs the whole network lifetime through balancing the average energy cost of each replica. What’s more, the intrusion detection thresholds can be dynamically adjusted according to the failure history to improve the accuracy of intrusion detection.

2. Power consumption is reduced by replicas classification. For a sensor group, the replica with high intrusion-tolerance capabilities will be activated only in case of failures. In the other case, the replica with lower energy consumption will be selected as the primary replica. Thus, the energy consumption caused by intrusion tolerance in the intrusion-free state will be reduced.

3 Network model and preliminaries

This section describes the application scenario of BSN, as well as the variables used in the ITFBS scheme. As shown in Fig. 1, the BSN is formed by a collection of biosensors, control nodes and a base station. In general, the biosensors are wireless wearable or implanted vital sign sensors which

![Fig. 1 BSN application scenario](image-url)
Biosensor nodes consist of a processor, memory, transceiver, sensors and a power unit. Each biosensor node is typically capable of sensing, processing, storing and transmitting the data. The control node periodically sends the sampled data to the medical server in the hospital through the base station, where they are stored for further processing [25].

Biosensors (such as sweat sensor, EKG sensor and temperature sensor) cannot afford heavy computation and communication due to limited computing capacity and energy consumption restrictions. Compared with biosensors, the control node (such as a cell phone or PDA) and the base station have comparatively higher transmission rate and computation power. In other words, BSN is a typical asymmetric structure.

Four types of attacks are considered in this paper as follows:

1. **Denial of service (DoS) attack**: DoS attacks can cause serious damage to resource-constrained BSN by preventing communication between the sensor node and the control node or by preventing a single device in BSN from sending data.
2. **Flooding attack**: In BSN, the compromised node or attack node can flood the network with broadcast messages. Flooding is an extreme resource-consuming operation, thus the mishandling of broadcast messages may lead to sensor failure due to the depletion of energy.
3. **Jamming attack**: An adversary can disrupt BSN by sending a more powerful signal over the radio frequencies used by sensor nodes in BSN.
4. **ID duplication attack**: Multiple nodes may be fabricated due to mis-configuration or intrusion, thus the data collection operations will be disturbed from the normal node.

We give the definitions of some variables used in the ITFBS scheme, as listed in Table 1.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_i$</td>
<td>sensor group indicating the specific sensor set with different replicas</td>
</tr>
<tr>
<td>$S_{ij}$</td>
<td>a sensor replica in sensor group $G_i$</td>
</tr>
<tr>
<td>$IN_{ijn}$</td>
<td>the number of incoming packets of sensor $S_{ij}$</td>
</tr>
<tr>
<td>${X_{ijn}}$</td>
<td>a random sequence used by CUSUM to detect abrupt changes</td>
</tr>
<tr>
<td>$R_A$</td>
<td>current primary replica</td>
</tr>
<tr>
<td>$R_B$</td>
<td>current non-primary replica</td>
</tr>
<tr>
<td>$Q_{available}$</td>
<td>the queue containing available replicas in a sensor group</td>
</tr>
<tr>
<td>$\delta$</td>
<td>the time for the primary replica transfer</td>
</tr>
</tbody>
</table>

![Fig. 2 Intrusion-tolerant model](image-url)

4 ITFBS scheme

Fig. 2 presents the ITFBS scheme. ITFBS includes three parts: intrusion-related information collection, intrusion detection and intrusion treatment. Biosensors send some intrusion-related information to the control node and execute commands from the control node. The control node analyses perceived intrusion-related information and makes intrusion treatment decisions to achieve adaptive intrusion tolerance. Obviously, the ITFBS scheme is entirely consistent with the asymmetric structure of BSN. In order to tolerate the intrusion, when the control node perceives sensor failure caused by intrusion, it will select a suitable replica as the new primary replica and activate it according to the feature of each replica in the sensor group. In addition, if the primary replica of a sensor group has been continuously running for a long time in the intrusion-free state, a primary replica transfer may also occur to save energy.

### 4.1 Intrusion-related information collection

The intrusion-related information is collected by the biosensor node and the control node. Based on that information, the control node dynamically determines the activation thresholds of the sensor group and proactively provides intrusion-tolerant solutions. In ITFBS, two kinds of intrusion-related information are collected:

1. **Information collected by the sensor node**: In order to detect the exhaustion intrusion caused by flooding attack, the sensor node should collect a variety of information, such as the number of incoming packets, the number of outgoing packets and the parameters (including buffer usage, the residual battery life, etc.) which reflects the reliability status of the sensor and can be derived from the...
inner hardware memory of the sensor. In addition, the number of packet collisions should be recorded to detect jamming attack and ID duplication attack.

2. Information collected by the control node: The control node collects some information to determine the status of sensor nodes and whether a sensor node is an adversary, such as the number of receiving packets from a sensor node, the transmission delay from a sensor node to the control node and the number of collisions caused by a sensor node.

4.2 Intrusion detection

In order to distinguish the occasional failures from attack instances promoted by intruders, we use the anomaly detection algorithm based on CUSUM [14]. CUSUM is a kind of change point detection algorithm widely used to detect the change of mean value of a random sequence. In ITFBS, a sensor node will be treated as being invaded or an adversary only if its intrusion-related information statistic is greater than an expected threshold. We will give detailed illustration of how to detect intrusion by the use of activation threshold in the following section.

4.2.1 Intrusion-related information statistics: Let \( \{Y_{ij}, n = 0, 1, \ldots\} \) be the number of incoming packets of sensor \( S_j \) in the sensor group \( G_i \) within a sampling period. The average number of incoming packets of \( S_j \) in a sampling period is \( \overline{Y}_{ij} \), and is calculated recursively as follows

\[
\overline{Y}_{ij}(n) = \alpha \overline{Y}_{ij}(n-1) + (1 - \alpha)Y_{ij}(n)
\]

where \( \overline{Y}_{ij}(n) \) is the number of incoming packets of \( S_j \) in the \( n \)th sampling period. \( \alpha (0 < \alpha < 1) \) is a constant parameter representing the estimated memory.

Then we can obtain \( Z_{ij} = \{\overline{Y}_{ij}(n)\} / (\overline{Y}_{ij}(n)) \), and \( \{Z_{ij}\} \) \( \cong 1 \) in normal condition. In order to use CUSUM, we transform \( \{Z_{ij}\} \) to another random sequence \( \{X_{ij}\} \) as follows

\[
X_{ij} = Z_{ij} - \beta
\]

where \( \beta \) is a constant parameter based on the system condition and used to produce \( \{X_{ij}\} \) with a negative mean. Generally, \( \beta \) is selected to be bigger than the mean of \( \{Z_{ij}\} \) under normal conditions. So, we can utilise non-parametric CUSUM with \( \{X_{ij}\} \) to detect changes in the number of incoming packets of sensor \( S_j \) as follows

\[
y_{ij} = 0
\]

\[
y_{ij} = y_{ij-1} + X_{ij}, \quad (n \geq 1)
\]

\[
(x)^+ = \begin{cases} x, & \text{if } x > 0 \\ 0, & \text{otherwise} \end{cases}
\]

4.2.2 Two-step threshold-based intrusion detection: Based on intrusion-related information, ITFBS detects abrupt changes by two-step threshold-based intrusion detection algorithm, where a preparation phase is added ahead of the execution of intrusion treatment.

Under normal circumstances, the mean of \( \{X_{ij}\} \) is negative. In anomaly condition, \( X_{ij} \) will become positive. \( \{y_{ij}\} \) accumulates over time. The large value of \( \{y_{ij}\} \) indicates that an attack occurs. In intrusion detection, the decision function is defined as \( d_{ij}(y_{ij}) \) presented by (6).

\[
d_{ij}(y_{ij}) = \begin{cases} -1, & \text{if } y_{ij} \leq N_{ij1} \\ 0, & \text{if } N_{ij1} < y_{ij} \leq N_{ij2} \\ 1, & \text{if } y_{ij} > N_{ij2} \end{cases}
\]

where \( N_{ij1} \) and \( N_{ij2} \) are two thresholds for the intrusion detection. Different from [12], we use two-step threshold-based scheme to detect the intrusion, so that the control node can trigger the preparation of the intrusion treatment based on the first threshold and trigger the execution of the intrusion treatment based on the second threshold to tolerate the intrusion proactively, that is, if \( d_{ij}(y_{ij}) = -1 \), then the condition of \( S_j \) is normal; if \( d_{ij}(y_{ij}) = 0 \), then the control node will trigger the preparation of the intrusion treatment; if \( d_{ij}(y_{ij}) = 1 \), then the execution of the intrusion treatment will be triggered. \( N_{ij1} \) and \( N_{ij2} \) can be dynamically calculated by the control node using (7) and (8) based on the failure history of sensor \( S_j \), thus the accuracy of intrusion detection can be improved dramatically.

\[
N_{ij2} = N_{ij1} - m_{ij} \epsilon
\]

where \( N_{ij2} \) is the default value of \( N_{ij2} \), \( m_{ij} \) is the number of intrusions within the latest \( \omega \) sampling periods \( (n \geq \omega) \), and \( \epsilon \) is a constant positive parameter.

\[
N_{ij1} = \begin{cases} N_{ij2} - \theta - v_{ij} \Delta t, & \text{if } v_{ij} > 0 \\ N_{ij1}, & \text{otherwise} \end{cases}
\]

where \( v_{ij} \) is the change rate of \( y_{ij} \) within the latest \( \omega \) sampling periods, \( \theta \) and \( \Delta t \) are constant positive parameters.

4.3 Passive replication fault tolerance

Compared with the existing passive replication methods, our methods can achieve much faster response while keeping lower power consumption owing to the following two aspects.

1. Classification of replicas based on their intrusion tolerant level: In the traditional passive replication methods, all replicas are identical. However, in ITFBS, replicas perform the same function, but their intrusion tolerance capability, power consumption and performance are different. According to the intrusion-related information, a suitable replica will be activated as the primary replica based on the feature of each replica in the sensor group at runtime. On the one hand, when there are some intrusions on the primary replica, the control node will select a replica with higher intrusion tolerance capability as the new primary replica. On the other hand, the control node will activate a replica with lower power consumption as the new primary replica if the primary replica with higher intrusion tolerance capability has continuously operated for a long time in the intrusion-free state. With this method, the power consumption of the system in the intrusion-free state can be reduced.

2. Intrusion-aware proactive intrusion tolerance: ITFBS can predict some failures by utilising two-step threshold-based intrusion detection based on the intrusion-related information. When a failure might occur, it will compensate for the failure even before it occurs. In ITFBS, the control node will activate a new primary replica for a sensor group,
if one of the following conditions is met: (i) the current primary replica of the sensor group has failed or will fail due to intrusions; (ii) the current primary replica of the sensor group has been continuously running for a long time in the intrusion-free state. By this means, it can not only improve the failure response time, but prolong the whole network lifetime through balancing the average energy cost of each replica as well.

4.4 Intrusion treatment

The procedure of primary replica transfer is presented in Fig. 3. Intrusion treatments are different according to the effectiveness of the current primary replica. The effective primary replica is the replica that can still work even though the intrusion occurs. Figs. 4 and 5 show the intrusion treatments for the non-effective primary replica and the effective primary replica, respectively.

For the non-effective primary replica, the intrusion treatment of adaptive intrusion tolerance with passive replication mainly includes six steps (refers to Fig. 4):

1. The control node will broadcast status request message STATUS_REQ to the non-primary replicas in the sensor group $G_1$, when it perceives the non-effectiveness of the primary replica $R_A$ in the sensor group $G_1$.
2. Available replicas in the sensor group $G_1$ send available status message STATUS and status request acknowledge message STATUS_ACK to the control node.
3. According to STATUS message, the control node sorts the available replicas based on predefined strategy in Section 4.3 to obtain a replicas queue $Q_{available}$. Thereafter the control node selects the best suitable replica $R_B$ from $Q_{available}$, and sends activation request message WAKEUP_ MSG and data request message POLLING_MSG to $R_B$.
4. If $R_B$ is activated successfully after receiving WAKEUP_MSG and POLLING_MSG message, then it will send the acknowledge messages WAKEUP_ACK and POLLING_ACK to the control node.
5. Replica $R_B$ sends sampled data to the control node according to the data request.
6. If the control node has received the acknowledge messages WAKEUP_ACK and POLLING_ACK, the control node will try the next available replica in $Q_{available}$ according to step (3).

Fig. 5 illustrates the intrusion treatment of adaptive intrusion tolerance with passive replication for the effective primary replica:

1. Current primary replica $R_A$ in the sensor group $G_1$ sends intrusion-related information FT_MSG (e.g. the $\gamma_{ij}$ of $R_A$ for incoming packets has exceeded the first threshold $N_{ij_a}$, see Section 4.2) to the control node.
2. The control node sends FT_ACK acknowledge message to $R_A$.
3. The control node broadcasts status request message STATUS_REQ to non-primary replicas in the sensor group $G_1$.
4. Available replicas in the sensor group $G_1$ send status request acknowledge message STATUS_ACK and available status message STATUS to the control node.
5. According to STATUS, the control node sorts the available replicas based on predefined strategy in Section 4.3 to obtain a replicas queue $Q_{available}$.
6. Current primary replica $R_4$ in the sensor group $G_1$ sends intrusion-related information FT_MSG (e.g. the $y_{ij}$ of $R_4$ for incoming packets has exceeded the second threshold $N_2$, see Section 4.2) to the control node once again.

7. The control node sends FT_ACK acknowledge message to $R_4$.

8. The control node selects the best suitable replica $R_B$ from $Q_{available}$ and sends activation request message WAKEUP_MSG and data request message POLLING_MSG to it.

9. If $R_B$ is activated successfully after receiving activation request message WAKEUP_MSG and data request message POLLING_MSG, then it will send the acknowledge messages WAKEUP_ACK and POLLING_ACK to the control node.

10. $R_B$ sends sampled data to the control node according to the data request.

11. The control node sends the acknowledge message DATA_ACK to $R_B$. If the control node has received the acknowledge messages WAKEUP_ACK and POLLING_ACK from $R_B$, it will send the sleep request PRUNE_MSG to $R_A$. Otherwise, the control node will try the next available replica in the available replicas queue $Q_{available}$ according to step (8).

12. After receiving the sleep request, $R_A$ stops to collect data and switches to sleep mode. The sleep mode acknowledge message PRUNE_ACK is sent to the control node.

13. $R_A$ will send acknowledge message FIN_ACK indicating that it has already sent all the sampled data in the buffer to the control node, when the buffer is empty.

14. In the condition of receiving PRUNE_ACK and FIN_ACK from $R_A$, the control node begins to deal with the sampled data sent by $R_B$, and the intrusion treatment finishes. It is worth noting that the control node will not process the sampled data sent by $R_B$ before it has received FIN_ACK from $R_A$.

5 Feasibility proof of ITFBS

In this section, we present proofs for correctness and effectiveness of ITFBS. Here, the correctness refers to the uniqueness of the primary replica and the consistency for configuration information of all members in the sensor group. The effectiveness includes dynamic characteristics of replicas and the satisfaction of read one write all and reply one (ROWARO) semantics.

**Theorem 1:** ITFBS satisfies the uniqueness of the primary replica corresponding to the control node. At any time, although there are multiple replicas in the sensor group, only one replica as the primary replica provides sampled data for the control node.

**Proof:** Suppose at any time $t$, the primary replica $R_A$ in the sensor group $G_1$ is in normal operation, then there is only one activated sensor replica in the sensor group during this process. Assuming at time $t_1$ ($t_1 > t$), the primary replica transfer is required due to intrusion factors (e.g. intrusion treatment for the non-effective primary replica, see Section 4.4) or non-intrusion factors (e.g. intrusion treatment for the effective primary replica, see Section 4.4), then the original primary replica $R_A$ will enter into sleep mode if it is effective and replica $R_B$ will be activated as the new primary replica at time $t_2$ ($t_2 = t_1 + \delta$, where $\delta$ is the time for primary replica transfer). So there is only one primary replica in active state in the sensor group $G_1$ anytime.

**Theorem 2:** ITFBS satisfies the consistency for configuration information of all members in the sensor group. All replica members in the sensor group have the same configuration information at any time.

**Proof:** Assume there are $j$ replica members $S_{i1}, S_{i2}, \ldots, S_{ij}$ in the sensor group $G_1$, when the control node sends configuration information to the sensor group $G_1$, because the control node broadcasts configuration information to all $j$ replica members in the sensor group $G_1$. The configuration information of replica members $S_{i1}, S_{i2}, \ldots, S_{ij}$ are consistent (e.g. the broadcast of status request message in intrusion treatment, see Section 4.4). When the control node requests sampled data from the sensor group $G_1$, because only one primary replica in the sensor group $G_1$ responds to the data request, it can still guarantee the consistency for configuration information of all replica members.

**Theorem 3:** ITFBS satisfies dynamic characteristics of replicas: the dynamic changes of replica members in the sensor group do not influence the consistency of the sensor group.

**Proof:** The dynamic changes of replica members include the addition and removal of replica members:

1. In the condition of removing replica members, if the replica member to be removed is non-primary replica, then the removal of this replica does not have any effect on the whole system. Otherwise, it will influence the system. The control node detects the removal of the primary replica after a period of time $\tau$ and selects a suitable replica available as the new primary replica to keep the continuity of sampled data (i.e. intrusion treatment for the non-effective primary replica, see Section 4.4).

2. During the addition of the replica member, for example, a new replica $S_{i1}$ is added into the sensor group $G_1$. As long as the configuration information of $S_{i1}$ is consistent with the other replica members in the sensor group $G_1$, it will not affect the whole system.

To sum up, the dynamic changes of group members may lead to certain issues of consistency, but the control node can guarantee the consistency of all replica members in the sensor group.

**Theorem 4:** ITFBS is able to ensure ROWARO semantics. For a sensor group, the sampled data request of the control node is executed only once. The configuration information from the control node must be written for all members in the sensor group. Only one sensor replica in the sensor group responds to the sampled data request of the control node.

**Proof:** During the system operation, only one primary replica member in the sensor group is activated, which executes and responds to the sampled data request of the control node (i.e. Theorem 1). For configuration information, they are sent to replica members in the sensor group in the form of multicast and written in all members (i.e. Theorem 2). In conclusion, ITFBS satisfies ROWARO semantics.
6 Performance analysis

ITFBS has been evaluated on Castalia simulator [26]. Castalia is an open source, discrete event-driven simulator designed specifically for WSN, BAN and general networks of low-power embedded devices based on OMNeT++ [27].

The experiment simulates a typical BSN, in which sensors measure a person’s physiological parameters. We configure the BSN with three types of sensor groups: ECG sensor group, SpO2 sensor group and temperature sensor group. Each sensor group has three different sensor replicas with different intrusion-tolerant levels and power consumptions. All nodes adopt Castalia standard CC2420 IEEE802.15.4 radios. Table 2 describes the detailed simulation parameters. We compare our ITFBS scheme with the scheme based on CUSUM detection algorithm [12] without replicas classification.

To evaluate the ability of ITFBS to copy with different types of intrusion, the simulation includes six time periods and introduces four types of intrusion into the system: DoS attack, flooding attack, jamming attack and ID duplication attack. In the first time period, 0–100 s, the system operates in the intrusion-free state. In the second time period, 100–200 s, the DoS attack is introduced through making a sensor group fail. In the third time period, 200–300 s, a compromised node floods the network with broadcast messages to simulate the flooding attack. In the fourth time period, 300–400 s, the noise floor is increased to introduce the jamming attack. In the fifth time period, 400–500 s, an attack node fabricates the ID used by other sensor node to introduce the ID duplication attack. In the sixth time period, 500–600 s, the system operates in the intrusion-free state once again.

In the whole experiment, we use three metrics to evaluate the performance of ITFBS: intrusion treatment response time, detection rate for a false positive rate of 5% and average energy consumption. The intrusion treatment response time is used to evaluate the performance of timeliness, and the detection rate is used to evaluate the accuracy of intrusion detection, and the average energy consumption is used to estimate the effect of intrusion tolerance in the energy consumption.

The intrusion treatment response time for different types of attacks during the simulation is shown in Fig. 6. It is obviously that the intrusion treatment response time of ITFBS is faster than that of the scheme based on CUSUM, the intrusion treatment is performed only after the happening of failure caused by intrusions. Thus, ITFBS can achieve faster intrusion treatment response time compared to the intrusion-tolerant scheme based on CUSUM.

Fig. 7 presents the intrusion detection rate for a false positive rate of 5%. It can be found that the intrusion detection rate of ITFBS is higher than that of the scheme based on CUSUM. In the system with ITFBS, the threshold for intrusion detection is updated dynamically based on the failure history to obtain high accuracy of intrusion detection, whereas in the system with the CUSUM-based scheme, the threshold is a constant. Therefore the system with ITFBS has better adaptability to intrusion than the system with the scheme based on CUSUM, and thus achieves a higher intrusion detection rate.

The average energy consumption per sensor nodes is presented in Fig. 8. ITFBS achieves lower energy consumption than the intrusion-tolerant scheme without replicas classification in the entire experimental process. In the first time period (i.e. the intrusion-free state), the average energy consumption of ITFBS is lower, because in the system with ITFBS, the high intrusion tolerance and high energy replica member may be activated only when the failure caused by intrusions has occurred or will occur, while in the system without replicas classification, the control node activates a sensor replica as the primary replica randomly. When intrusions occur, the average energy consumption of ITFBS is increased due to the intrusion treatment, but only the sensor groups with failures use the high intrusion tolerance and high energy replicas, thus the average energy consumption of ITFBS is still lower. It is worth noting that, in the sixth time period, there

---

Table 2 Simulation parameters

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>number of sensor types</td>
<td>3 (ECG, SpO2, temperature)</td>
</tr>
<tr>
<td>wireless channel model</td>
<td>log shadowing wireless model</td>
</tr>
<tr>
<td>path loss exponent</td>
<td>2.4</td>
</tr>
<tr>
<td>collision model</td>
<td>additive interference model</td>
</tr>
<tr>
<td>physical and MAC layer</td>
<td>IEEE 802.15.4 standard</td>
</tr>
<tr>
<td>data transmission rate</td>
<td>250 kbps</td>
</tr>
<tr>
<td>buffer size</td>
<td>1024 kbytes</td>
</tr>
<tr>
<td>max physical layer frame size</td>
<td>127 bytes</td>
</tr>
<tr>
<td>physical layer frame overhead</td>
<td>6 bytes</td>
</tr>
<tr>
<td>MAC layer frame overhead</td>
<td>13 bytes</td>
</tr>
<tr>
<td>simulation time</td>
<td>600 s</td>
</tr>
</tbody>
</table>

---

Fig. 6 Intrusion treatment response time

Fig. 7 Intrusion detection rate
In this paper, an adaptive intrusion-tolerant scheme with passive replication for BSN in smart space applications, namely ITFBS, has been proposed. ITFBS can tolerate intrusions through the intrusion-aware proactive intrusion tolerance with passive replication. We theoretically prove the correctness and effectiveness of ITFBS, and evaluate its performance by simulation experiments. Simulation results show that ITFBS achieves high performance while keeping lower energy consumption. The primary contributions of this paper are summarised as follows:

1. An asymmetric architecture is adopted, in which the resource-constrained sensor nodes just do little processing and the control node with abundant resources performs the majority of intrusion-tolerant treatments.
2. Two-step threshold-based scheme and dynamic threshold approach are used to enhance the existing intrusion detection, through which both the intrusion response time and the intrusion detection accuracy are improved.
3. Compared with the existing passive replication fault-tolerant strategy, the average energy cost of each replica in the sensor group is balanced by replicas classification, consequently prolonging the lifetime of the whole network.

In the current implementation, the parameters are configured at system initialisation. As part of our future work, we would like to design a parameter update strategy to configure them dynamically during runtime based on knowledge learned from previous experiences.

8 Acknowledgments

This work was partially supported by the National Natural Science Foundation of China under grant 61173179 and 60903153, and the Fundamental Research Funds for the Central Universities.

9 References

11. Cui, Z., Li, X., Wang, J.: ‘Adaptive intrusion tolerance strategy of the system based on artificial immune’. Int. Conf. on Computational Intelligence and Software Engineering (CSIE 2009), Wuhan, China, December 2009, pp. 1–4
26 Pham, H.N., Pediaditis, D., Boulis, A.: ‘From simulation to real deployments in WSN and back’. IEEE Int. Symp. on World of Wireless, Mobile and Multimedia Networks (WoWMoM 2007), Helsinki, Finland, June 2007, pp. 1–6