Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 646781, 7 pages
http://dx.doi.org/10.1155/2013/646781

Research Article

LIRT: A Lightweight Scheme for Indistinguishability, Reachability, and Timeliness in Wireless Sensor Control Networks

Wei Ren,[1] Liangli Ma,[2] and Yi Ren[3]

[1] *School of Computer Science, China University of Geosciences, Lumo Road 388, Wuhan 430074, China*
[2] *Department of Computer Engineering, Naval University of Engineering, Jiefang Road 717, Wuhan 430033, China*
[3] *Department of Computer Science, National Chiao Tung University, Hsinchu 30010, Taiwan*

Correspondence should be addressed to Wei Ren; weirencs@cug.edu.cn

Received 30 June 2013; Accepted 25 September 2013

Academic Editor: Hongli Xu

Wireless sensor control networks (WSCNs) are important scenarios and trends in mobile wireless sensor networks. Compared with traditional wireless sensor networks, WSCNs have two specific characteristics: entities in networks are extended from passive sensors to active sensors (e.g., actuators and actors) and the transmitting messages are extended from data only to data plus (e.g., data and control instructions). Thus new security problems arises. In this paper, we make the first attempt to specify the security requirements for WSCNs, namely, indistinguishability, reachability, and timeliness. In addition, several new attacks in WSCN, distinguishing risks, dropping attacks, and disordering attacks, are pointed out at the first time. A lightweight scheme LIRT is proposed with tailored design to guarantee the indistinguishability, reachability, and timeliness in WSCNs. Extensive and rigorous analysis on LIRT justifies its security strength and performance measures.

1. Introduction

Mobile wireless sensor networks are an extended form of traditional wireless sensor networks where nodes are not static but mobile. Recently, sensors in mobile wireless sensor networks become more and more versatile, for example, underwater sensors, body sensors, and control sensors. The control sensors are usually divided into two classes: actuators and actors. Those actuator and actor sensor networks extend traditional wireless sensor networks from passive networks to active networks, from data networks to control networks, via adding functionalities such as response and action. The actuator and actor sensor networks start to be applied in new applications such as smart grid, smart city, smart building, and factory automation, to name a few [1–3].

Wireless actuator and actor sensor networks can be viewed as wireless sensor control networks (WSCNs) over a group of sensors. WSCNs have two distinctions compared with traditional wireless sensor networks as follows. (1) The entities in networks are extended from sensors only to sensors plus. For example, there exist sensors, actuators, and actors in WSCNs. Actuators may perform actively for controlling, but sensors in traditional wireless sensor networks usually act passively for collecting data. (2) The transmitting messages in networks are extended from data only to data plus, for example, data messages and control messages. Therefore, new security problems arise in WSCNs. If entities in WSCNs can be distinguished by adversaries, adversaries will be able to launch a target attack (that has been explored in our previous paper [4]); if data or control messages are dropped by adversaries, the control loop will be terminated; if data or control messages are disordered, the control status or sequences may be disturbed. We called them indistinguishability, reachability, and timeliness problems in WSCNs. Note that those security problems cannot be solved by previous security schemes for traditional wireless sensor networks due to the specialities of WSCNs. We thus have to explore new methods to solve them, especially, in a tailored design manner.

Concretely, security in wireless control networks starts to attract more and more attention [5–9]. Those work majorally address different contexts from WSCNs, so the solutions may not be able to tackle the aforementioned security requirements. Moreover, the security problems in WSCNs are challengeable due to the inherent properties: wireless lossy channels, jamming-sensitive links, resource-constraint sensor devices, control timing demands, and control sequence ordering requirements.

In this paper, we make the first attempt to clarify and analyze the security requirements in WSCNs and then propose a lightweight scheme called LIRT to guarantee those requirements, namely, indispensability, reachability, and timeliness in WSCNs. We formally prove the achievement of the proposed scheme. Different from other works and previous approaches, all presentations in the paper strictly follow formal expressions for better clarity and rigorous generality.

The contributions of the paper are listed as follows.

(i) We make the first attempt to define formal attacks and security requirements in WSCNs, namely, indistinguishability, reachability, and timeliness in WSCNs.

(ii) We make the first attempt to propose a lightweight scheme to guarantee those security requirements and formally prove the security goals that are achieved.

The rest of the paper is organized as follows. Section 2 gives an overview on relevant prior work. In Section 3 we discuss the basic assumption and models used throughout the paper. Section 4 provides the detailed description and analysis of our proposed scheme. Finally, Section 5 concludes the paper.

2. Related Work

Wireless sensor networks for automation control have attracted more and more attention in recent years [5, 8, 10–12]. Yen et al. [5] proposed packet loss problem in wireless networked control system over IEEE 802.15.4e. They proposed redundant transmission. de Filippi et al. [7] proposed single-sensor control strategies for semiactive steering damper control in two-wheeled vehicles. Thurman et al. [9] explored acoustic sensors in an unmanned underwater vehicle to provide full autonomy control. Au et al. [13] proposed energy-efficient classification algorithms for wearable sensor systems. All the above work focuses on control performance but not control security.

The security problems in WSCNs have not been thoroughly explored in recent work. Target attacks for wireless machine-to-machine control networks are first pointed out by our previous work [4]. We also proposed a scheme called RISE to mitigate target attacks. Stealthy deception attacks in water SCADA systems are first pointed out by Amin et al. [6]. They discuss sensor networks but mainly in wired SCADA networks. Zheng et al. [3] discussed reliable problem in wireless communication networks that support demand and response control. They proposed several methods for deriving reliability performance. Short et al. [2] discussed burst errors in wireless control networks.

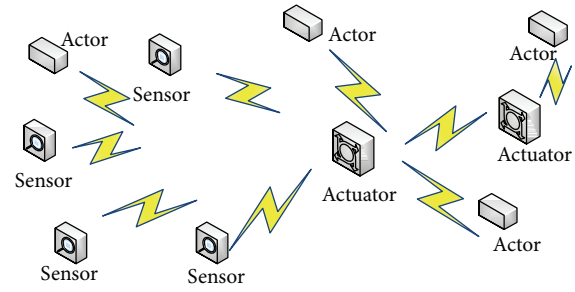

Figure 1: Wireless sensor control networks. Sensors collect sensing data; actuators respond corresponding instructions; actors execute those instructions.

They proposed application-level strategies for ameliorating the effects of packet losses and burst errors in sampled-data control systems. Above related work are independent with our discussion and solution in the following, as our analysis for the security requirements are different from them.

3. Problem Formulation

3.1. Network Model and Attack Model. There exist three major entities (denoted as E) in WSCNs: sensors (denoted as S), actuator (denoted as U), and actors (denoted as C). Usually, sensors send data to actuators. Actuators send instructions to actors. Actors execute instructions. The number of sensors is usually more than that of actuators. The number of actuators is usually less than that of actors. Figure 1 depicts the entities in WSCNs.

We assume that the links among sensors, actuators, and actors are not secure. The adversaries (denoted as A) in the links can launch the following attacks. We assume that the security boundary is out of the entities of WSCNs. That is, we assume entities are trustworthy. The trust models in WSCN scenarios are analyzed in detail in our previous work [14].

Definition 1 (message distinguishing risk (R_m)). Adversaries may distinguish data and instructions in transmitting messages in WSCNs, after viewing the behavior and messages among entities in WSCNs.

The observation is the only advantage of adversaries, as we suppose the links are not secure. It can be formally described as follows:

$$\mathsf{R}_m = \Pr\Big\{\mathsf{A}\ \text{correctly guesses}\ m^* \in D \parallel m^* \in I \mid m^* \Longleftarrow D \cup I, 2^{\mathsf{E}_i \to \mathsf{E}_i:\{m^*\}}, \mathsf{A} \Longleftarrow m^*\Big\}, \tag{1}$$

where $\Pr\{A \mid B\}$ denotes the probability that A happens after event B happens; D means data; I means instructions; m^* is any message in D or I; 2^{ϕ} means the power set of a set ϕ.

Definition 2 (entity distinguishing risk (R_e)). Adversaries may distinguish sensors, actuators, and actors among entities in WSCNs, after viewing the behavior and messages among entities in WSCNs.

It can be formally described as follows:

$$\mathsf{R}_e = \Pr\Big\{\mathsf{A} \text{ correctly guesses } \mathsf{E}_i \in \mathsf{S}\,(\|\mathsf{U}\|\,\mathsf{C}) \mid E_i \Longleftarrow \mathsf{S}\,(\|\mathsf{U}\|\,\mathsf{C}), 2^{\mathsf{E}_i \to \mathsf{E}_j:\{m^*\}}, \mathsf{A} \Longleftarrow m^*\Big\}. \tag{2}$$

Definition 3 (dropping attack (A_d)). Adversaries may drop data that are sent from sensors to actuators and instructions that are sent from actuators to actors.

Definition 4 (disordering attack (A_i)). Adversaries may disturb the arrival time of data at actuators and the arrival time of instructions at actors.

It is natural to see that the prerequisite for dropping attack is message distinguishing risk and entity distinguishing risk.

The disordering attack can be launched without any prerequisite information about message distinguishing risk and entity distinguishing risk. It is thus much easier to be launched via just jamming arbitrary packets into channels, and it is thus more difficult to be defended against.

3.2. Security Definition and Design Goal. The security requirements are defined as follows.

Definition 5 (indistinguishability). The data and instruction cannot be distinguished from messages by adversaries from all their observations. The sensors, actuators, and actors cannot be distinguished from entities by adversaries from their observations.

Indistinguishability is formally described as

$$H\,(\mathsf{S} \mid O) = H\,(\mathsf{U} \mid O) = H\,(\mathsf{C} \mid O) = H\,(\mathsf{S}) = H\,(\mathsf{U}) = H\,(\mathsf{C}), \tag{3}$$

where O is the observation of adversaries; H is the entropy of correctly guessing on entities.

Definition 6 (reachability). The data can arrive at designated actuators finally, and instructions can arrive at designated actors finally.

Definition 7 (timeliness). The data can arrive at actuators timely, and instructions can arrive at actors timely.

Therefore, the design goal is to propose a scheme for guaranteeing indistinguishability, reachability, and timeliness in a lightweight way.

4. Proposed Schemes

We list major notations used in the remainder of the paper in Table 1.

4.1. Indistinguishability. As message distinguishing risk, R_m, and entity distinguishing risk, R_e, are the prerequisite of dropping attack, A_d, we first propose a method to eliminate those risks.

Table 1: Notations.

WSCNs	Wireless sensor control networks
E	Entities
S	Sensors
U	Actuators
C	Actors
A	Adversaries
R_m	Message distinguishing risk
R_e	Entity distinguishing risk
A_d	Dropping attack
A_i	Disordering attack

Proposition 8. *Entity indistinguishability is equivalent to message indistinguishability.*

Proof (straightforward). If entities are distinguishable, messages will be distinguishable via the entities who send; if messages are distinguishable, entities will be distinguishable by analyzing their sending messages. □

Thus, we discuss two risks together. We firstly analyze the information or advantages that can be obtained by adversaries. Adversaries can observe the following behavior and messages in WSCNs.

The messages that can be observed for message distinguishing are as follows:

(M-O1) the length of the messages that are sent among entities,

(M-O2) the format and semantics of the messages that are sent among entities.

The behavior that can be observed for distinguishing entities is as follows.

(B-O1) The sending sequence of the messages and entities, it is a list of messages and entities which send messages in an observing time span. For example, in an observation time span with k minutes, the sending sequences of messages are $\{m_1, m_2, \ldots, m_n\}$. In an observation time span with k minutes, the sending sequences of entities who send messages are $\{E_1, E_2, \ldots, E_n\}$, where $E_i \in \mathsf{E}$, $i = 1, \ldots, n$. The sequences can be observed for distinguishing entities. For example, sensors may always stand at the head of the sequence, and actors may always stand at the rear of the sequence.

(B-O2) The interval of two sequentially sending messages at any two entities; for example, E_i sends m_p, and then E_j sends m_{p+1}. m_p, m_{p+1} are two consecutively sending messages. The time interval between these two messages can be observed for distinguishing entities. For example, actuators may always send a message after sensors send a message, and actors may always send a message after actuator sends a message.

(B-O3) The interval of two consecutively sending messages at one entity, for example, E_i sends m_p and

m_{p+1} sequentially, where $E_i \in \mathsf{E}, m_p, m_{p+1}$. That is, m_p, m_{p+1} are two sequential messages sent from E_i. The time interval between these two messages can be observed for distinguishing entities. For example, sensors may always send messages in a fixed interval.

(B-O4) The interval of k sequentially sending messages at any k entities, it is a generalization of (B-O2). For example, suppose k entities send k messages sequentially. They are $E_1, E_2, \ldots, E_k$. The time interval among them can be observed for distinguishing entities. For example, sensors, actuators, and actors may form a control loop. Observing loops may infer the entity observed.

(B-O5) The interval of k sequential sending messages at one entity, it is a generalization of (B-O3). For example, E_i sends k messages: $m_1, m_2, \ldots, m_k$, where $E_i \in \mathsf{E}$. The time interval among them can be observed for distinguishing entities. For example, intervals for message sending at sensors, at actuators, or at actors may be quite different. Observing those difference may infer the entity observed.

Proposition 9. *If and only if the observation is indistinguishable, the message and entity are indistinguishable.*

Proof. The observation at adversaries is the only knowledge to distinguish message and entity. If and only if the observation is indistinguishable, the message and entity are indistinguishable. □

Therefore, we propose the following strategies via randomization to make observation indistinguishable. Each strategy addresses one observation.

(IND-S1) All messages that are sent among entities have the same length.

(IND-S2) All messages that are sent among entities are encrypted for hiding semantics.

(IND-S3) The sending sequence of the messages among entities is randomized.

(IND-S4) The interval of two sequentially sending messages at any two entities is randomized.

(IND-S5) The interval of two sequentially sending messages at one entity is randomized.

(IND-S6) The intervals of k sequentially sending messages at any k entities are randomized.

(IND-S7) The intervals of k sequentially sending messages at one entity are randomized.

Proposition 10. *Strategy (IND-S6) can be guaranteed by (IND-S4).*

Proof (straightforward). As any interval of two sequentially sending messages at any two entities is randomized, and the intervals of k sequentially sending messages at any k entities in k are also randomized. □

```
Date: M = {D ‖ I ‖ NULL}, W
Result: Sending Packets with Indistinguishability
Initialization;
While T do
    M ⇐ Get Out Queue Buffer();
    //Get packet from Outgoing Queue
    t ⇐ Random() %W;
    //W is the suspended time slot
    Suspend(t);
    if (M <> NULL) then
        TempPkt ⇐ ExtendToFixLen(M);
        //Maintain the same length
    else
        TempPkt ⇐ Create FixLen Dummy();
        //Create dummy packet
    end
    M' ⇐ MaskMsg(TempPkt);
    //Encryption before sending
    SendMsg(M');
end
```

Algorithm 1: Sending algorithm for indistinguishability (SAI algorithm).

Proposition 11. *Strategy (IND-S7) can be guaranteed by (IND-S5).*

Proof (straightforward). The interval of two sequentially sending messages at one entity is randomized, the intervals of k sequentially sending messages at one entity are thus also randomized. □

Thus, the sending algorithm for indistinguishability (called SAI algorithm) at each entity is proposed in Algorithm 1.

4.2. Reachability. As adversaries cannot distinguish messages and entities, they have to drop messages (e.g., by jamming channels) randomly to launch a dropping attack.

To guarantee the reachability of the data and instruction messages, we propose the following strategies via redundancy.

(RCH-S1) Data and instruction are sent for α times.

Proposition 12. *If the dropping probability of a packet is p, strategy (RCH-S1) can guarantee the reachability with probability $1 - p^\alpha$.*

Proof. As the dropping probability of one packet is p, its reachability is $1 - p$. The probability of α packets that are dropped is p^α, and the reachability of at least one in α packets is thus $1 - p^\alpha$. □

The repeat sending for α times can increase the probability of reachability, but it also causes communication overhead. In the following strategies, we will tackle the communication overhead by optimization.

(RCH-S2) Data and instruction are sent for random times in $[\beta_1, \beta_2]$. The repeat times are varied. Usually,

we have $\alpha = \beta_2$. Therefore, it can both increase the probability of reachability and tackle the communication overhead.

Proposition 13. *If the dropping probability of a packet is p, strategy (RCH-S2) can guarantee the reachability with probability at least* $(1/(\beta_2 - \beta_1)) \sum_{\beta_1}^{\beta_2}(1 - p^i)$.

Proof. The reachability of one packet in $[\beta_1, \beta_2]$ is $1 - p^i$, $i \in [\beta_1, \beta_2]$. The expectation of this probability for all i, $i \in [\beta_1, \beta_2]$ is thus $(1/(\beta_2 - \beta_1)) \sum_{\beta_1}^{\beta_2}(1 - p^i)$. □

Proposition 14. *The average communication overhead in (RCH-S2) is less than (RCH-S1) by* $1 - (\beta_1 + \beta_2)/2\alpha$.

Proof (straightforward). The communication overhead in (RCH-S1) is $O(\alpha)$; the communication overhead in (RCH-S2) is $O((\beta_1 + \beta_2)/2)$. Thus, the advantages in (RCH-S2) compared with (RCH-S1) are $(\alpha - (\beta_1 + \beta_2)/2)/\alpha$. That is, $1 - (\beta_1 + \beta_2)/2\alpha$. □

(RCH-S3) The repeating times at originators for data or instruction are γ_1. The repeating times at forwarders for dummy packets are γ_2. Usually, $\gamma_1 \ll \beta_1, \gamma_1 + \gamma_2 \approx \alpha$.

Originators are the entities where data or instruction are originated from. For example, the first sensor who sends the data is the originator for that data. Forwarders are the entities between originators and designated destination entities. That is, forwarders forward the data or instruction to packet destination. The dummy packets at immediate forwarders are not created from meaningless dummy string (NULL) but created from data or instruction received previously by immediate forwarders. That is, before forwarders send dummy packets, they choose the last one in received data or instruction as a dummy packet. This strategy can both improve the reachability of data or instruction and mitigate the communication overhead.

Proposition 15. *If the dropping probability of a packet is p, strategy (RCH-S3) can guarantee the reachability with probability* $1 - p^{\gamma_1+\gamma_2}$. *The communication overhead is* γ_1/α *of that in strategy (RCH-S1).*

Proof (straightforward). The proof is similar to the former proposition. □

The sending algorithm for reachability (SAR algorithm) at each entity is given in Algorithm 2.

```
Date: M = {D ‖ I ‖ NULL}, γ1, γ2, W
Result: Sending Packets for Reachability
//at Originators:
Initialization;
while T then
  M ⇐ Get Out Queue Buffer();
  if (M <> NULL) then
    Count ⇐ 0;
    while Count < γ1 do
      t ⇐ Random() %W;
      //W is suspended time slot
      Suspend(t);
      TempPkt ⇐ ExtendToFixLen(M);
      M′ ⇐ MaskMsg(TempPkt);
      SendMsg(M′);
      Count + +;
    end
  else
    t ⇐ Random() %W;
    Suspend(t);
    TempPkt ⇐ CreateFixLenDummy();
    M′ ⇐ MaskMsg(TempPkt);
    SendMsg(M′);
  end
end
//at Forwarders:
Initialization;
while T do
  M ⇐ Get In Queue Buffer();
  //Get packet from ingress queue
  if (M <> NULL) then
    Count ⇐ 0;
    while (Count < γ2) do
      t ⇐ Random() %W;
      Suspend(t);
      TempPkt ⇐ Extend To FixLen(M);
      M′ ⇐ MaskMsg(TempPkt);
      SendMsg(M′);
      Count + +;
    end
  else
    t ⇐ Random() %W;
    Suspend(t);
    TempPkt ⇐ Create FixLen Dummy();
    M′ ⇐ MaskMsg(TempPkt);
    SendMsg(M′);
  end
end
```

Algorithm 2: Sending algorithm for reachability (SAR algorithm).

4.3. Timeliness. Adversaries cannot distinguish messages and entities. The dropping attack cannot aim at designated messages or entities. The dropping is thus randomly dropping, for example, by jamming channels. The jamming subsequently results in disordering attack. The timeliness of the control operations is damaged. To guarantee the timeliness of the data and instruction messages, we propose following strategies.

(TML-S1) The suspended time is randomly chosen from a time slot that is shortened exponentially. That is, $W_s \overset{r}{\Leftarrow} 1/2^n * W$, where W is the maximal suspended time slot at the first time; n is the suspending times; $\overset{r}{\Leftarrow}$ means "is randomly chosen from";

W_s is the actual suspended time. The timeliness can be improved with the exponentially shortening of suspended time. This strategy is corresponding to (RCH-S1).

Proposition 16. *If the suspended time slot W in α times is shortened to $1/2^{\alpha}W$, strategy (TML-S1) can guarantee the timeliness with total suspended time $\sum_{i=1}^{\alpha}(1/2^i) * W$.*

Proof. Suppose the suspended time slot is W, the suspended time in expectation is $1/2W$. If the suspended time slot is shortened to $1/2^n * W$, the suspended time in expectation is $1/2^{n+1} * W$. If first $\alpha - 1$ are all dropped by adversaries, the worst suspended time is $\sum_{i=1}^{\alpha}(1/2^i) * W$. □

Similarly, (TML-S2) can be proposed corresponding to (RCH-S2). That is, when data and instruction are sent for random times in $[\beta_1, \beta_2]$, the suspending time between two consecutively sending is randomly chosen from a time slot that is shortened exponentially. That is, $W_s \overset{r}{\Leftarrow} 1/2^n * W$.

(TML-S3) We propose to shorten the suspended time slot exponentially at forwarders. The minimum is lower bounded by a threshold value, denoted as Th.

The sending algorithm for timeliness (SAT algorithm) at each entity is given in Algorithm 3.

Proposition 17. *Strategy (TML-S3) does not damage indistinguishability.*

Proof. Everyone in WSCNs may be originators or forwarders. It depends on messages to forwarder or originator. Originators or forwarders both shorten suspended time slot exponentially. Thus, strategy (TML-S3) does not damage indistinguishability. □

Proposition 18. *If the suspended time slot W in γ_1 times is shortened to $1/2^{\gamma_1}W$ at originators and in γ_2 times is shortened to $1/2^{\gamma_2}W$, strategy (TML-S1) can guarantee the timeliness with time $(\sum_{i=1}^{\gamma_1}(1/2^i) + \sum_{i=1}^{\gamma_2}(1/2^i)) * W$.*

Proof (straightforward). The proof is similar to the proof of the former proposition. □

The proposed scheme—LIRT—is the combination of strategies for indistinguishability, reachability, and timeliness. As the strategy (TML-S3) includes SAI, SAR, and SAT, it can be viewed as an appropriate version of LIRT. As the scheme is described intentionally in an incremental manner in this section, the advantages of LIRT are clear to follow for its advantages due to the improvements step by step.

```
Date: M = {D ‖ I ‖ NULL}, γ1, γ2, W
Result: Sending Packets for Timeliness
//at Originators: Initialization;
While T do
  M ⇐ Get Out Queue Buffer();
  if (M <> NULL) then
    Count ⇐ 0;
    W' ⇐ W;
    While (Count < γ1) do
      t ⇐ Random()% W';
      W' ⇐ max(1/2 * W', Th);
      //Exponentially suspending
      Suspend(t);
      TempPkt ⇐ Extend To FixLen(M);
      M' ⇐ MaskMsg(TempPkt);
      SendMsg(M');
      Count + +;
    end
  else
    t ⇐ Random() %W;
    Suspend(t);
    TempPkt ⇐ Create FixLen Dummy();
    M' ⇐ MaskMsg(TempPkt);
    SendMsg(M');
  end
end
//at Forwarders: Initialization;
while T do
  M ⇐ Get In Queue Buffer();
  if (M <> NULL) then
    Count ⇐ 0;
    W' ⇐ W;
    while (Count < γ2) do
      t ⇐ Random() %W';
      W' ⇐ max(1/2 * W', Th);
      Suspend(t);
      TempPkt ⇐ Extend To FixLen(M);
      M' ⇐ MaskMsg(TempPkt);
      SendMsg(M');
      Count + +;
    end
  else
      t ⇐ Random() %W;
      Suspend(t);
      TempPkt ⇐ Create FixLen Dummy();
      M' ⇐ MaskMsg(TempPkt);
      SendMsg(M');
  end
end
```

Algorithm 3: Sending algorithm for timeliness (SAT algorithm).

5. Discussion

In former discussion, feedback information such as networking status and receiver's acknowledgement is not used for simplicity. If feedback information is available, it can be used to enhance previous strategies by achieving adaptive and optimal overall performance. The feedback information that can be gathered by senders is as follows.

(i) The network feedback on network status, it is sent by intermediate forwarders or detectors, and it reports congestion, risks, and dropping rate of messages.

(ii) The feedback from receivers, it is sent by designated destination of messages, and it reports message arrival, delay, jitter, and timeliness.

If the feedback information is available in WSCNs, the strategies can be enhanced by intelligent method for adaptivity and optimization.

6. Conclusions

WSCNs are important types in mobile wireless sensor networks and present their own characteristics compared with traditional wireless sensor networks. In this paper, we made the first attempt to specify the security requirements for WSCNs, in which of the upmost importance are indistinguishability, reachability, and timeliness. To clarify and illustrate the security requirements, several new attacks in WSCNs were pointed out at the first time, for example, distinguishing risks, dropping attacks, and disordering attacks. To defend against those attacks, a lightweight scheme LIRT was proposed. LIST can guarantee the indistinguishability, reachability, and timeliness in WSCNs, which is justified by extensive and rigorous analysis on security strength. The performance of LIRT is also measured by communication overhead, to confirm its applicability in realistic scenarios.

Acknowledgments

Wei Ren's research was financially supported by the National Natural Science Foundation of China (61170217), the Open Research Fund from the Shandong Provincial Key Laboratory of Computer Network (SDKLCN-2011-01), Fundamental Research Funds for the Central Universities (CUG120109), and Wuhan Planning Project of Science and Technology (2013010501010144). Yi Ren's research was sponsored in part by the "Aim for the Top University Project" of the National Chiao Tung University and the Ministry of Education, Taiwan.

References

[1] M. A. S. Masoum, P. S. Moses, and K. M. Smedley, "Distribution transformer losses and performance in smart grids with residential plug-in electric vehicles," in *Proceedings of the IEEE PES Innovative Smart Grid Technologies (ISGT '11)*, pp. 1–7, January 2011.

[2] M. Short, U. Abrar, and F. Abugchem, "Application level compensation for burst errors in wireless control networks," in *Proceedings of the 17th IEEE Conference on Emerging Technologies Factory Automation (ETFA '12)*, pp. 1–8, 2012.

[3] L. Zheng, N. Lu, and L. Cai, "Reliable wireless communication networks for demand response control," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 133–140, 2013.

[4] W. Ren, L. Yu, L. Ma, and Y. Ren, "RISE: a reliable and secure scheme for wireless machine to machine communications," *Tsinghua Science & Technology*, vol. 18, no. 1, pp. 100–117, 2013.

[5] B. Yen, D. Hop, and M. Yoo, "Redundant transmission in wireless networked control system over IEEE 802.15.4e," in *Proceedings of the International Conference on Information Networking (ICOIN '13)*, pp. 628–631, 2013.

[6] S. Amin, X. Litrico, S. Sastry, and A. M. Bayen, "Cyber security of water scada systems—part I: analysis and experimentation of stealthy deception attacks," *IEEE Transactions on Control Systems and Technology*, vol. 21, no. 1963, p. 1970, 2012.

[7] P. de Filippi, M. Corno, M. Tanelli, and S. M. Savaresi, "Single-sensor control strategies for semi-active steering damper control in two-wheeled vehicles," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 2, pp. 813–820, 2012.

[8] J. Ploennigs, V. Vasyutynskyy, and K. Kabitzsch, "Comparative study of energy-efficient sampling approaches for wireless control networks," *IEEE Transactions on Industrial Informatics*, vol. 6, no. 3, pp. 416–424, 2010.

[9] E. Thurman, J. Riordan, and D. Toal, "Real-time adaptive control of multiple colocated acoustic sensors for an unmanned underwater vehicle," *IEEE Journal of Oceanic Engineering*, vol. 38, no. 3, pp. 419–432, 2013.

[10] G. Lee, J. Lee, E. Lee, and D. Kim, "Synchronization algorithm for hybrid control networks: can and wireless control networks," in *Proceedings of the IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS '11)*, pp. 1–4, 2011.

[11] N. Son, D. Tan, and D. Kim, "Backoff algorithm for time critical sporadic data in industrial wireless sensor networks," in *Proceedings of the International Conference on Advanced Technologies for Communications (ATC '12)*, pp. 255–258, 2012.

[12] R. A. Swartz, J. P. Lynch, and C.-H. Loh, "Near real-time system identification in a wireless sensor network for adaptive feedback control," in *Proceedings of the American Control Conference (ACC '09)*, pp. 3914–3919, June 2009.

[13] L. K. Au, A. A. T. Bui, M. A. Batalin, and W. J. Kaiser, "Energy-efficient context classification with dynamic sensor control," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 6, no. 2, pp. 167–178, 2012.

[14] W. Ren, L. Yu, L. Ma, and Y. Ren, "How to authenticate a device? Formal authentication models for M2M communications defending against ghost compromising attack," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 679450, 9 pages, 2013.