Tracking and Positioning Mechanism for Mobile Intrusion Nodes

Kai Guo[1, a],Jia Dong[2, b],Wenzhi Chen[3, c],TianChi Zhao[4, d]

[1]International School, Beijing University of Posts & Telecommunications, Beijing, 102209, China

[2]International School, Beijing University of Posts & Telecommunications, Beijing, 102209, China

[3]International School, Beijing University of Posts & Telecommunications, Beijing, 102209, China

[4]International School, Beijing University of Posts & Telecommunications, Beijing, 102209, China

[a]guokai@bupt.edu.cn, [b]j.dong@se10.qmul.ac.uk,[c]2013212976@bupt.edu.cn,[d]ztc101@bupt.edu.cn

Keywords: TDMA; ETD-MAC; Matlab; Wireless Sensor Network; Target Tracking;

Abstract. Wireless sensor network (WSN), which is usually composed of a large number of wireless communication nodes, is a new area of research in the field of information science. These nodes can be equipped with some sensor modules. Wireless network is an important part of the Internet of Things. With the sustainable development of wireless communication technology, more attention has been paid to the security issues of WSNs. In this paper, we discuss the tracking and positioning mechanism for mobile intrusion nodes. We propose a tracking and positioning mechanism by which the intrusion nodes can be identified and relocated rapidly with high probability and low energy consumption.

Introduction

Time Division Multiple Access (TDMA) protocol is a Medium Access Control (MAC) layer protocol which is based on event-driven wireless sensor networks. TDMA protocol is suitable for data communication and event detection of static nodes. We propose a MAC layer protocol based on event tracking and detection (ETD-MAC) in order to cope with mobile intrusion nodes. Our protocol uses an energy clustering mechanism to assist ETD-MAC, and designs a novel node dormancy mechanism to reduce energy consumption. We verify our idea using NS2 simulation software and Mica2 node experiment platform. The experimental results demonstrate that our new protocol has low power consumption, extensible and low latency characteristics.

ETD-MAC Protocol

Model assumption

ETD-MAC protocol is a clustering TDMA-MAC protocol which is suitable for tracking interest events in an intensive distributed WSN. It is designed with the following features:

(1) Extensible: protocol can adjust automatically according to development direction or tendency of network interest event and generate the relative clustering.

(2) Energy efficiency: this protocol can make network energy distribution homogenization and save tracking energy when satisfying event tracking conditions.

(3) Low latency: by reducing time slot of data, we can reduce the time of data transmission for event tracking. In this way, we decrease event tracking latency.

Symbols we need in this paper are defined below:

(1) Sensor node deployment satisfies uniform distribution and average network convergence rate is Kcov, where Kcov > k (k is the minimal convergence rate for satisfy event tracking and detection).

(2) Our protocol uses the same time synchronization mechanism as S-MAC protocol.

(3) Each node can receive data from its neighbor and detect events within its detection range.

(4) Sensors are divided into 5 levels according to their energy, where 5 refers to highest energy. When we choose cluster-head node, nodes of level 5 would be given top priority, and nodes of level2 and level1 must not be chosen.

(5) There are 3 states when sensor nodes detect events: active state, sleep state and monitoring state.

(6) As is shown in figure 1, protocol data can be embedded into a communication data frame which contains data type, source node ID, destination node ID and data transmission.

Type	ID	Dest	Data

Fig.1. communication data frame

ETD-MAC Protocol

As figure 2 shows, each round the operation of ETD-MAC protocol contains Setup Phrase, Steady-State Phrase and Dissolution Phrase according to the different detection state.

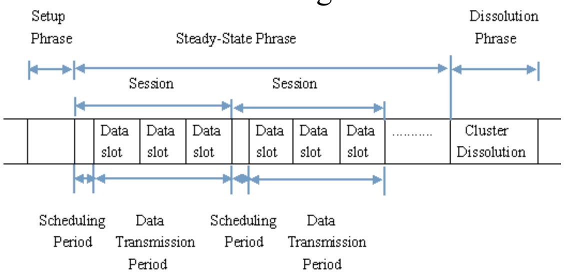

Fig.2. operation descriptions for ETD-MAC protocol

(1) Initial setting: when an interest event occurs (such as the position of intrusion node changing continuously), nodes detected by sensor will turn to monitoring state. If the channel is occupied at this time, sensor nodes will keep on monitoring and receive the demand of cluster-head. If cluster is not chosen, those nodes will detect whether their energy is higher than level 2. If it is true, they will broadcast the frame CCH three times.

CCH	ID	Broadcast	RSS	Energy
8 bits	8bits	8bits	16bits	8bits

Fig.3. CCH broadcast frame structure

Nodes which send CCH broadcast frame will also receive CCH broadcast frames from other competition nodes. Here are rules of the competition: sensor node with highest received signal strength indicator (RSSI) will be given highest priority, since it can generate highest cover for cluster node. If there are several nodes having same or similar RSSI, the node with highest residual energy will become cluster-head node. If there are several nodes which satisfy the condition, sensor with smallest ID will be chosen. Then the cluster-head node will broadcast a HELLO message to establish the cluster. After sensor nodes receive that HELLO message, they will keep monitoring and wait for changing into steady state.

HELLO	ID	Broadcast	CH
8bits	8bits	8bits	8bits

Fig.4. HELLO message structure

(2) Steady state: steady state can be divided into scheduling period and data transmission period. During the scheduling period, sensor nodes in monitor state will send RTS frame to cluster head.

RTS	ID	CH ID	RSS	Energy
8 bits	8bits	8bits	16bits	8bits

Fig.5. RTS frame structure

Cluster head will choose nodes according to the competition rule above, where k is the minimum number of nodes required in covering event area. Next, scheduling rule will be set in accordance with the RSSI value detected. The node with higher RSSI will be assigned prior communication slot because of its low error rate. Then, cluster head will send a CTS broadcast frame. Cluster nodes that receive this frame will turn into waiting state and transmit data in their transmission slot. Other nodes will turn into dormancy state for T seconds, where T is defined by the demand of event detection to reduce energy consumption of sensor nodes.

CTS	ID	Broadcast	Schedule	Start-time
8 bits	8bits	8bits	8kbits	8bits

Fig.6. CTS frame structure

When data is transmitted, sensor in active state will send a data frame as follows.

DATA	ID	CH ID	RSS	Energy	TimeStamp	Start-time
8bits	8bits	8bits	16bits	8bits	32bits	8bits

Fig.7. DATA frame structure

(3) Release statement: after each phase finishes, cluster-head will examine its energy level and whether there is any event loss. If true, it will broadcast a release statement data frame. After that, other nodes will delete record information related with CH and initiate a new cluster-head selection.

Dormancy mechanism based on energy

If a sensor node is not chosen in competition, it will set a dormancy period with time T to reduce the number of active nodes during the competition. As is shown in Figure 8:

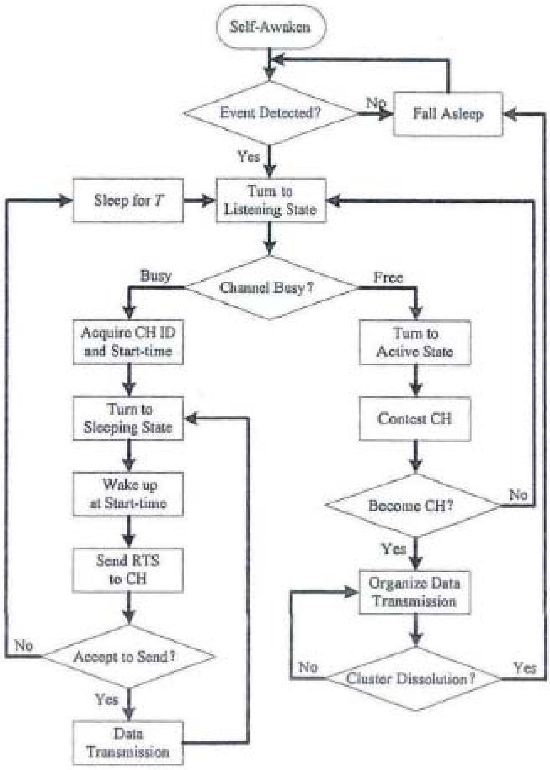

Fig.8. dormancy mechanism

Dormancy period T is an important parameter in this mechanism. Event cannot be traced if this value is too high. However, if the value is too low, energy consumption of the system cannot be reduced effectively. So we need to set T according to the frequency of interest event, detection parameter k and average network coverage ratio kcov.

The energy-based dormancy mechanism will minimize channel monitor time and nodes competition to reduce energy consumption. CTS and DATA frame both contain a timestamp of next phase to identify transition time of different phases. Sensor nodes should get timestamp of new phase when they are awake and send a RTS data frame, so they need not keep monitoring the wireless channel.

We use the number of nodes of interest event detection and loss each round to represent the frequency of interest event. We define them as Ndetect and Nlose, where Ndetect = Nlose =λ (λ is constant number). It satisfies Poisson distribution of parameter kcov, and is lower than k.

$$poisson(Number \geqq k|kcov) \geqq 0.95 \quad (1)$$

We assume the length of event is Δt. Then dormancy period T is:

$$T=n\Delta t(n\geqq 1, n\in N) \quad (2)$$

T is chosen to minimize number of competition nodes. Ncontest(i) represent number of sensor nodes that attend competition data transmission in round i. We assume that there are s rounds in each phase. So the objective function of dormancy period T is:

$$T(n)=\min \sum_{i=1}^{s} N(i)_{contest} \quad (3)$$

According to (1) (2) and (3), dormancy period T can be represented as a nonlinear programming.

$$T(n)=\min \sum_{i=1}^{s} N_{contest}(i) \quad (4)$$

s.t.

$n\geqq 1$; poisson(Number≧k|kcov)≧0.95

where n represents equation number of nonlinear programming constraint. Nodes lost in round i can be divided into two groups: dormancy node and data transmit node in last round.

$$Nlose=NLdormancy(i)+NLtransmit(i) \quad (5)$$

Competition node can be divided into three parts: nodes with interest event exist after finishing data transmission, nodes with interest event detected in the last round and nodes to be awake after executing T seconds dormancy period.

$$Ncontest(i)=NRtransmit(i)+Ndetect+NRawake(i) \quad (6)$$

NRtransmit(i)=k-NLtransmit(i) and Ndetect=Nlose=NLdormancy(i)+NLtransmit(i). So:

$$Ncontest(i)=k+NLdormancy(i)+NRawake(i) \quad (7)$$

The number of dormancy sensor node:

$$Ndormancy(i)=NLdormancy+NRawake(i) \quad (8)$$

Because of the uniform distribution, mathematical expectation of NLdormancy is

$$\overline{NL_{dormancy}(1)} = \lambda \frac{k_{cov}-k}{k_{cov}} \quad (9)$$

Parameter NRawake(i) is determined by the reduced nodes number after n rounds of Ndormancy(i-n) due to the reduced interest event.

In the round i-n+1, reduced node number is:

$$N_{dormancy}(i\text{-}n)\frac{NL_{dormancy}(i\text{-}n\text{+}1)}{K_{cov}-k} \quad (10)$$

In the round i-n+2, reduced node number is:

$$\left[1-\frac{NL_{dormancy}(i\text{-}n\text{+}1)}{K_{cov}-k}\right]N_{dormancy}(i\text{-}n)\frac{NL_{dormancy}(i\text{-}n\text{+}2)}{K_{cov}-k} \quad (11)$$

During the current round, reduced node number is:

$$N_{dormancy}(i\text{-}n)\frac{NL_{dormancy}(i)}{K_{cov}-k}\prod_{j=i-n+1}^{i-1}\left[1-\frac{NL_{dormancy}(j)}{K_{cov}-k}\right] \quad (12)$$

According to (9), mathematical expectation of NRawake(i) is

$$\overline{NR_{awake}(i)}=N_{dormancy}(i\text{-}n)\prod_{j=i-n+1}^{i-1}\left[1-\frac{NL_{dormancy}(j)}{K_{cov}-k}\right]=(1-\frac{\lambda}{K_{cov}})^{n}N_{dormancy}(i\text{-}n) \quad (13)$$

According (9) and (10), mathematical expectation of Ndormancy(i) is

$$\overline{NR_{dormancy}(i)}=(1-\frac{\lambda}{K_{cov}})^{n}\overline{N_{dormancy}(i\text{-}n)}+\lambda\frac{K_{cov}-k}{K_{cov}} \quad (14)$$

From round 1 to round n, the mathematical expectation of Ndormancy (i)can be described as follows: when Kcov –k (except the first round), mathematical expectation of Ndormancy(i) equals to that of NLdormancy(i). When a node turns to dormancy state, they cannot be awakened for T seconds. So we can use equation (15) to represent i, where a and b are integers:

$$i = an + b, a \ge 0, b \in [1, n) \tag{15}$$

If b=1, we can get

$$\overline{N_{dormancy}(i)}=(1-\frac{\lambda}{K_{cov}})^n[\lambda\frac{K_{cov}-k}{K_{cov}}+(1-\frac{\lambda}{K_{cov}})^n]^{\alpha}(K_{cov}-k)+\lambda\frac{K_{cov}-k}{K_{cov}} \tag{16}$$

If b is not 1, we can get

$$\overline{N_{dormancy}(i)}=(1-\frac{\lambda}{K_{cov}})^n[\lambda\frac{K_{cov}-k}{K_{cov}}+(1-\frac{\lambda}{K_{cov}})^n]^{\alpha}(\lambda\frac{K_{cov}-k}{K_{cov}})+\lambda\frac{K_{cov}-k}{K_{cov}} \tag{17}$$

Substitute it into equation (3), we can get

$$T(n)=\min\sum\nolimits_{i=an+1}^{s}(k+N_{dormancy}(i))+\sum\nolimits_{i=n+2,i=an+1}^{s}(k+N_{dormancy}(i))+(K_{cov}-k)+\lambda\frac{K_{cov}-k}{K_{cov}}(n-1) \tag{18}$$

If s=cn, c≧1 and c is a integer, nonlinear function (4) can become:

$$T(n)=\min A_1^n[A_1^n+A_2^{c-2}-1](A_2 n+A_3A_1)+cn(k+A_2)-nk+A_3A_1 \tag{19}$$

s.t.

$n \geqq 1$; poisson(Number≧k|kcov)≧0.95;

$$A_1=1-\frac{\lambda}{K_{cov}},\ A_2=\lambda\frac{K_{cov}-k}{K_{cov}}, A_3=K_{cov}-k$$

We can use Matlab to solve equation (19). If Kcov=12, k=4, λ=4 and c=10, we can get (20)

$$T(n)=\min 0.67^n[0.67^n+1.33^8-1](1.33n+5.33)+49.3n+5.33 \tag{20}$$

s.t.

$n \geqq 1$;   poisson(Number≧k|kcov)≧0.95;

As figure 9 shows, when n=4, nonlinear programming can get a minimum value of 265. The average competition node number is 6.6 which is less than that of TDMA-MAC (value 12).

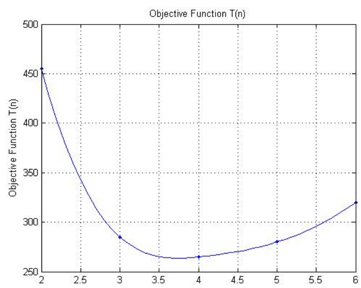

Fig.9. Objective T (n)

Experimental Result

Our experiment is implemented based on energy saving, extensibility, and low latency features and standard energy model sensor nodes is used. We use NS2 simulation software to do the test and compare it with TDMA-MAC protocol. Because NS2 simulation software uses simplified communication channel and RF model, we use practical parameters of Mica2 node during the simulation and compare the performance with S-MAC protocol.

Simulation Energy Consumption

We set 100 sensor nodes and deploy them in a 100m×100m square area with all nodes locate at a fix position. If the cluster-head node is an aggregation node in this cluster and other nodes can communicate directly with the head node, we can set energy consumption parameters of sensor nodes as follows according to observation results of Mica2 nodes: dormancy energy consumption is 0.06mW, receiving energy consumption is 30mW, monitoring energy consumption is 24mW and sending energy consumption is 81mW. Other parameter set as Table 1.

Table 1 simulation parameters

Bandwidth[Kps]		15
Event Detection Requirement[nodes]		4
ETD-MAC Session	Scheduling Period[s]	0.4
	Transmitting Period[s]	0.6
Cluster-based TDMA MAC Session	Scheduling Period[s]	0.4
	Transmitting Period[s]	0.6
	Idle Period[s]	0.5

We can increase RF intensity to enlarge communication range of sensor nodes and average network coverage. If we increase λ, frequency of interest event will increase as well. As we can see in figure 10 and figure 11, signal strength of interest event do not have a strong impact on energy consumption for ETD-MAC protocol. However, when $\lambda \geqq 4$, energy consumption will grow obviously with the increasing network coverage. If the velocity of interest event is too high, dormancy mechanism will have little effect on network energy consumption. In this way, energy consumption of ETD-MAC protocol approximately equals to that of TDMA-MAC protocol.

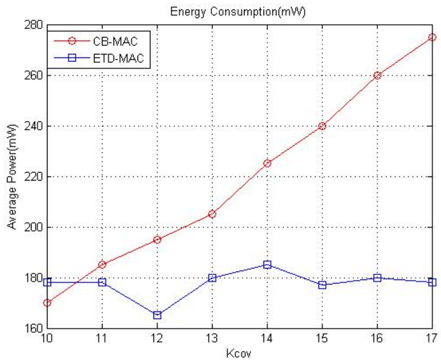

Fig.10. the energy consumption of ETD-MAC protocol and CB-MAC frequency of interest event increases

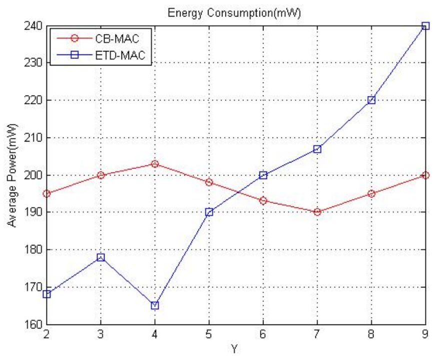

Fig.11. the energy consumption of ETD-MAC and CB-MAC protocol when protocol when Kcov increases

Extensibility and Latency Experiment

In order to do further analysis, we let sensor nodes collect and store perception data which is produced by interest event. Intrusion node would generate a new interest event that will awake network node and execute ETD-MAC protocol. We also execute S-MAC protocol in our experiment to compare their execution latency and extensibility. Dormancy awake event of S-MAC protocol lasts for 2 seconds. Figure 12 shows the latency from event detection to information receiving by cluster-head when k increases from 2 to 9.

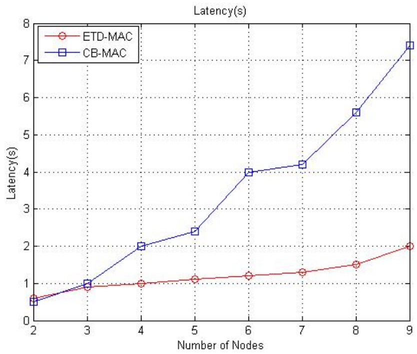

Fig.12. trace latency of ETD-MAC and S-MAC

Summary

In this paper, we propose an ETD-MAC protocol which can apply to trace interest event. We can detect and trace network mobile intrusion node using this protocol. Compared with TDMA-MAC protocol, ETD-MAC protocol is suitable for event tracing and detection in wireless sensor network. It can satisfy the demand of energy consumption, communication latency and extensibility. Moreover, we design a dormancy mechanism to reduce energy consumption of the system. Simulation results and Mica2 nodes experimental results prove that our detection protocol has good performance in terms of node energy consumption, detection efficiency, detection event and extensibility. So we achieve our design goal.

References

[1] Xiaoming Deng, Yan Xiong, and Depin Chen. 2010. Mobility-assisted Detection of the，Replication Attacks in Mobile Wireless Sensor Networks [C],In Proceedings of tiie 6 th International Conference on Wireless and Mobile Computing, 225-232.

[2] X. Meng, K. Lin, and K. Li. 2010. A note-based randomized and distributed protocol for detecting node replication attacks in wireless sensor networks, Algorithms and Architectures for Parallel Processing, 559-570.

[3] Y.X. Hou; H.G. Wang; J.X. Liang; C.X Pei. 2009. A Cross-Layer Protocol for Event-Driven Wireless Sensor Networks [C],1st International Conference on Information Science and Engineering, 3926-3929.

[4] Z.Merhi, M.Elgamel and M.Bayoumi. 2009. EB-MAC: An event based medium access control for wireless sensor networks [C], IEEE International Conference on Pervasive Computing and Communications, 1-6.

[5]Zhijun Li and Guang Gong. Randomly Directed Exploration: An Efficient Node Clone Detection Protocol in Wireless Sensor Networks [C]. 2009. MASS,09,6th International Conference on Mobile Ad-hoc and Sensor Systems, 12-15.

[6] Zhou J,Das T K, Lopez J. 2008. An Asynchronous Node Replication Attack in Wireless Sensor Networks [C]. In IFDP International Federation for Information Processing, Proceedings of the IFIP TC 11 23rd International Information Security Conference, Sushil Jajodia, Pierangela Samarati, Stelvio Cimato, 125 – 139.

[7]Zhu B Addada V,Setia S,et al. 2007. Efficient Distributed Detection of Node Replication Attacks in Sensor Networks [C]. Twenty-Third Annual Computer Security Applications Conference (ACSAC 2007)sComputer Security Applications Conference, Annual, 257-267.