ISSN: 2229-6948(ONLINE)
DOI: 10.21917/ijct.2020.0312



# ENERGY EFFICIENT AND SECURE DATA TRANSMISSION USING COOPERATIVE ROUTING IN NETWORKS

**M. Saravanan[1], Lakshminarayanan[2] and Allanki Sanyasi Rao[3]**

[1]*Department of Computer Science and Engineering, KPR Institute of Engineering and Technology, India*
[2]*Department of Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, India*
[3]*Department of Computer Science Engineering, Balaji Institute of Technology and Science, India*

***Abstract***

***Wireless Sensor Networks (WSNs) are emerging as a promising technology because of their wide range of applications in industrial, environmental monitoring, military and civilian domains. Due to economic considerations, the nodes are usually simple and low cost. They are often unattended, however, and are hence likely to suffer from different types of novel attack. WSNs are increasingly being deployed in security-critical applications. Due to their inherent resource-constrained characteristics, they are prone to various security attacks, and a black hole attack is a type of attack that seriously affects data collection. In this paper, using the recent advances in uncertain reasoning that has originated from the artificial intelligence community, we propose a trust management scheme named Hybrid and Efficient Intrusion Detection Systems that enhances the security in networks. Here, we have used two frameworks for Trust Calculation and Decision Making process. The trust value is derived using Bayesian Inference, and Decision Making is based on Dempster-Shafer theory, which is a mathematical theory of evidence.***

***Keywords:***

***WSN, Dempster-Shafer Theory, Intrusion Detection System, Artificial Intelligence***

## 1. INTRODUCTION

Wireless sensor networks (WSNs) are increasingly being deployed in security-critical applications. Owing to its inherent resource-constrained characteristics, WSNs are prone to various security attacks like the black hole attack, which seriously affects data collection. The adversary compromises a node and drops all packets that are routed via this node, thereby rendering sensitive data to be discarded or to be disabled from being forwarded to the sink. Consequently, the network will completely fail and, more seriously, make incorrect decisions as the network makes decisions depending on the nodes' sensed data. Therefore, the detection and elimination of BLA is of great significance when it comes to the security in WSNs. Many researches with a key focus on avoiding black holes are happening, and there are also the ones that do not require black hole information in advance.

As per the approach of this paper, the packet is divided into $M$ shares, which are sent to the sink via different routes (multi-path), but the packet can be resumed with $T$ shares ($T \leq M$). However, a limitation is that the sink may receive more than the required $T$ shares, thus leading to high energy consumption. Another preferred strategy that can improve route success probability is the trust route strategy. The main feature of this strategy is to create a route by selecting nodes with high trust, because such nodes have a higher probability of routing successfully. Thus, the routes created in this manner can forward data to the sink with a higher success probability.

## 2. OVERVIEW

WSNs are emerging as a promising technology because of their wide range of applications in industrial, environmental monitoring, military and civilian domains. Due to economic considerations, the nodes are usually simple and low cost. They are often unattended, however, and are hence likely to suffer from different types of novel attack.

WSNs are increasingly being deployed in security-critical applications. Due to their inherent resource-constrained characteristics, they are prone to various security attacks, and a black hole attack is a type of attack that seriously affects data collection. To conquer that challenge, an active detection-based security and trust routing scheme named Active Trust is proposed for WSNs. The most important innovation of Active Trust is that it avoids black holes through the active creation of a number of detection routes to quickly detect and obtain nodal trust and thus improve the data route security. In the proposed process, we add energy efficiency model by using Sleep Wake Scheduling technique in the active trust method.

In this paper, using recent advances in uncertain reasoning originated from artificial intelligence community, we propose a trust management scheme that enhances the security in networks named Hybrid and Efficient Intrusion Detection Systems.

Here, we use two frameworks for Trust Calculation and Decision Making process. The trust value is derived using Bayesian Inference and Decision Making is based on Dempster'-Shafer theory, which is a mathematical theory of evidence.

In the proposed process, we add energy efficiency model by using Sleep Wake Scheduling technique in Trust method. We can achieve more energy consumption and high energy efficiency compared to previous Active Trust model.

- The Trust scheme (Fig.1) is the first routing scheme that uses active detection routing to address BLA. The most significant difference between Active Trust and the previous research is that we create multiple detection routes in regions with residue energy; because the attacker is not aware of detection routes, it will attack these routes and, in so doing, be exposed. In this way, the attacker's behavior and location, as well as nodal trust, can be obtained and used to avoid black holes when processing real data routes. To the best of our knowledge, this is the first proposed active detection mechanism in WSNs.

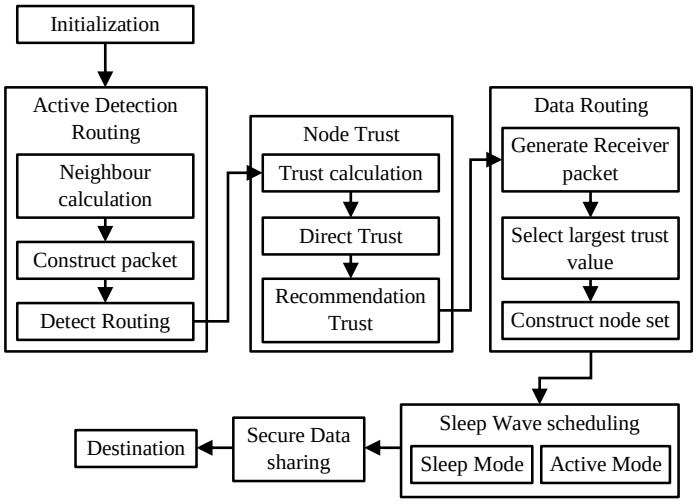


Fig.1. Trust Route Protocol

- The Trust route protocol has better energy efficiency. Energy is very precious in WSNs, and there will be more energy consumption if active detection is processed. Therefore, in the previous research, it was impossible to imagine adopting such high-energy-consumption active detection routes. However, we find it possible after carefully analyzing the energy consumption in WSNs. Research has noted that there is still up to 90% residue energy in WSNs when the network has died due to the energy hole phenomenon. Therefore, the Active Trust scheme takes full advantage of the residue energy to create detection routes and attempts to decrease energy consumption in hotspots (to improve network lifetime). Those detection routes can detect the nodal trust without decreasing lifetime and thus improve the network security. According to theoretical analysis and experimental results, the energy efficiency of the Active Trust scheme is improved more than 2 times compared to previous routing schemes, including shortest routing, multi-path routing.
- The Trust scheme has better security performance. Compared with the previous research, nodal trust can be obtained in Active Trust. The route is created by the principle of initially choosing nodes with high trust to avoid potential attack, and then routing along a successful detection route. Through the above approach, the network security can be improved.
- Through our extensive theoretical analysis and simulation study, the Active Trust routing scheme proposed in this paper can improve the success of routing probability by 1.5 times to 6 times and the energy efficiency by more than 2 times compared with that of the previous researches.

# 3. MODULE DESCRIPTION

## 3.1 ACTIVE DETECTION ROUTING

A detection route refers to a route without data packets whose goal is to convince the adversary to launch an attack so the system can identify the attack behavior and then mark the black hole location. Thus, the system can lower the trust of suspicious nodes and increment the trust of nodes in successful routing routes.

Through active detection routing, nodal trust can be quickly obtained and it can effectively guide the data route in choosing nodes with high trust to avoid black holes. In this scheme, the source node randomly selects an undetected neighbor node to create an active detection route. Considering that the longest detection route length is *w*, the detection route decreases its length by 1 for every hop until the length is decreased to 0, and then the detection route ends. This section details the implementation of the active detection routing protocol.

The content of the detection routing packet can be divided into 6 parts, as shown in Fig.2(a), where it contains (a) packet head; (b) packet type; (c) ID of the source node; (d) maximum detection route length; (e) acknowledge returned to the source for every w hops; and (f) ID of the packet.

The source node selects an undetected node to launch the detection route. Once the detection packet is received by nodes, the maximum route length is decreased by 1. After that, if it is 0, a feedback packet is generated and a feedback route is launched to the source, and then is restored to the initial value. If it is not 0, it proceeds to select the next hop in the same way; or otherwise, ends the route. The structure of a feedback packet (Fig.2(b)) is also composed of 6 parts, namely: (a) packet head; (b) packet type; (c) ID of the source node; (d) destination node; (e) ID of the detection packet; and (f) ID of the packet.

| Header | Type | Source | ϖ | ω | id |
|---|---|---|---|---|---|

(a) Routing Packet

| Header | Type | Source | Destination | S-id | id |
|---|---|---|---|---|---|

(b) Feedback Packet

Fig.2. Packet Structure

The feedback packet is routed back to the data source; because nodes cache the detection route info, the feedback packet is able to return to the source, and the following is the algorithm for the detection route protocol.

## 3.2 NODAL TRUST

During data routing and detection routing, every node will perform a nodal trust calculation to aid in black hole avoidance. When node A performs a detection route for node B, it is verified if the detection data are successfully routed.

*Nodal direction trust*: Consider the trust set of node A to node B during *t* to be: Nodal recommendation trust: Node A is the trust evaluator, node C is the target of evaluation, and node B is a recommender of A. Consider B A C to be the direction trust of A to B and C BC to be the direction trust of B to C; then, the recommendation trust of A to C is. For the trust of multiple recommendations, the calculation of the recommendation trust from A to B, B to C, etc., until D to E.

*Recommendation trust merging*: Consider that the recommender set of node A and that the recommendation trust is up to node K; then, the merged trust is A to K.

*Comprehensive trust*: Comprehensive trust is the total trust, which merges the recommendation trust and direction trust: Comprehensive trust of a node can be computed as follows: After the node launches a detection route, it calculates the direction trust according to Eq. for each received feedback packet. Through

interactions, the node obtains the recommendation trust from its neighbors, and it then calculates the merged trust, for the multiple-recommendation trust. Finally, it calculates the comprehensive trust accordingly.

### 3.3 DATA ROUTING

The data routing refers to the process of nodal data routing to the sink. The routing protocol is similar to common routing protocols in WSNs; the difference is that the route will select a node with high trust for the next hop to avoid black holes and thus improve the success ratio of reaching the sink.

The core idea of data routing is that when any node receives a data packet, it selects one node from the set of candidates nearer the sink whose trust is greater than the preset threshold as the next hop. If the node cannot find any such appropriate next hop node, it will send a feedback failure to the upper node, and the upper node will re-calculate the unselected node set and select the node with the largest trust as the next hop. Similarly, if it cannot find any such appropriate next hop, it sends a feedback failure to its upper node. The upper node, working in the same manner, will re-select a different node from among its neighbors nearer to the sink until the data are routed to the sink or there is conclusively no path to the sink.

### 3.4 SLEEP WAKE SCHEDULING

In this module deals with initializing the nodes in network topology. Here, network topology and topography are used for the network animator window (network window). A syntax is used to create nodes in network animator window. Sensors that are active or asleep are called as surviving sensors and sensors that are malfunctioning or deadlines are called to fail. Sensor modes vary based upon the active sensors that vary each and every time. So, in this work a method is proposed to decide a sleep schedule at each and every key time.

The 1st key time is the initial time at which each sensor works with the initial battery power. Here the sleep schedule is initialized. During the 1st key if some targets are not covered it means that the 2nd key time has started. At the 2nd key time, the sensor's information is updated and the sleep schedule is followed to cover all targets. Similarly, the 3rd key time, 4th key time and so on could be followed. And, a sleep schedule is followed at each key time until survival sensors cannot cover all targets.

Table.1. System and Hardware Requirements

| **Hardware Requirement** | |
|---|---|
| Processor | Intel Pentium IV |
| Processor Speed | 1.4GHz |
| **Software Requirements** | |
| Operating System | Ubuntu 10.04 |
| Simulator Tool | NS 2.34 |
| Language | TCL |
| Protocol Design | C++ |
| Platform | Independent |

We can replace the third switch by a light sensor, when the day light is very low and when it is time to activate the automatic operating mode. However, the switch 1 and switch 2 can be used for emergency situations. The base station can also be connected to a computer to display the details of any serial HyperTerminal, the address numbers of lampposts that have a problem and need the intervention of the maintenance team. This option is very helpful to reduce the maintenance costs.

## 4. IMPLEMENTATION

NS-2 is an open-source simulation tool running on Unix-like operating systems. It is a discreet event simulator targeted at networking research and provides substantial support for simulation of routing, multicast protocols and IP protocols, such as UDP, TCP, RTP and SRM over wired, wireless and satellite networks. It is a useful tool with many advantages like support for multiple protocols and the capability of graphically detailing network traffic. Additionally, NS-2 supports several algorithms in routing and queuing. LAN routing and broadcasts are part of routing algorithms. Queuing algorithm includes fair queuing, deficit round robin and FIFO.

The NS-2 started as a variant of the REAL network simulator. REAL is a network simulator which was originally intended for studying the dynamic behavior of flow and the congestion control schemes in packet-switched data networks. In 1995 NS development was supported by Defense Advanced Research Projects Agency DARPA through the VINT project at LBL, Xerox PARC, UCB, and USC/ISI. The wireless code from UCB Daedalus and CMU Monarch projects and Sun Microsystems enhanced the wireless capabilities of NS-2.

NS-2 is available on several platforms such as FreeBSD, Linux, SunOS and Solaris. It also builds and runs on Windows with Cygwin. Simple scenarios should run on any reasonable machine; however, very large scenarios require the benefit of large memory and fast CPU's.

### 4.1 NS-2 LANGUAGE

NS-2 is basically written in C++, with an OTcl (Object Tool Command Language) interpreter as a front-end. It supports a class hierarchy in C++, called compiled hierarchy and a similar one within the OTcl interpreter, called interpreter hierarchy. Some objects are completely implemented in C++, some others in OTcl and some are implemented in both.

As already mentioned above, NS-2 is an object-oriented, discrete event simulator. There are presently five schedulers available in the simulator each of which is implemented by using a different data structure: a simple linked-list, heap, calendar queue (default) and a special type called real-time.

The scheduler runs by selecting the next earliest event, executing it to completion, and returning to execute the next event. The units of time used by the scheduler are seconds.

An event is handled by calling the appropriate Handler class. The most important Handler is NS Object with Tcl Object as its twin in the OTcl world. They provide all the basic functions allowing objects to interact with one another. For this purpose the receive function group is mainly used. For handling OTcl statements in C++, Ns Objects provide the so-called command function. NsObject is the parent class for some important classes as the Classifier, the Connector and the Trace File class.

In NS-2 network, physical activities are translated to events, and the events are queued and processed in the order of their scheduled occurrences. And the simulation time progresses with the events processed. And also the simulation time may not be the real life time as inputted.

It can model essential network components, traffic models and applications. Typically, it can configure transport layer protocols, routing protocols, interface queues, and also link layer mechanisms.

It can be easily seen that this software tool could, in fact, provide us a whole view of the network construction, while also maintaining the flexibility for us to decide. Thus, just this one software can help us simulate nearly all parts of the network. This definitely will greatly minimize the investment in network infrastructure. The Fig.3 shows a layered structure which NS-2 can simulate for us.

After the simulation is completed, NS-2 presents detailed information on the network layer to the extent of providing a huge trace file recording of all the events in a line by line fashion. The event driven mechanism used in NS-2 could maintain all the happenings as records. These records could be traced to evaluate the performance of special stuffs in the network, such as routing protocol, Mac layer load, and so on.

### 4.2 EVENT SCHEDULER

In the Event scheduler, if multiple data were to be processed at the same time, it processes the same one by one using the FIFO concept. This ensures that there is no congestion while transferring the packets.

### 4.3 PACKETS

It is the collection of data, wherein, whether header is called or not, all header files would be present in the stack registers.

## 5. CREATE NETWORK TOPOLOGY (PHYSICAL LAYER)

The Physical Layer is the first and lowest layer in the seven-layer OSI model of computer networking. The implementation of this layer is often termed PHY. The Physical Layer consists of the basic hardware transmission technologies of a network. It is a fundamental layer underlying the logical data structures of the higher level functions in a network. Due to the plethora of available hardware technologies with widely varying characteristics, this is perhaps the most complex layer in the OSI architecture. The Physical Layer defines the means of transmitting raw bits rather than logical data packets over a physical link connecting networking nodes. The bit stream may be grouped into code words or symbols and converted to a physical that is transmitted over Hardware.

### 5.1 TRANSPORT CONNECTION (TRANSPORT LAYER)

Transport layers are contained in both the TCP/IP which is the foundation of the INTERNET and the OSI model of general networking. The definitions of the Transport Layer are slightly different in these two models. This article primarily refers to the TCP/IP model, in which TCP is largely for a convenient application programming interface to internet hosts, as opposed to the OSI model of definition interface. The most well-known transport protocol is the (TCP). It lent its name to the title of the entire internet protocol suite TCP/IP. It is used for connection-oriented transmissions, whereas the connectionless user datagram suite (UDP) is used for simpler messaging transmissions. TCP is the more complex protocol, due to its state full design incorporating reliable transmission and data stream services.

### 5.2 GENERATE TRAFFIC (APPLICATION LAYER)

In TCP/IP, the Application Layer contains all protocols and methods that fall into the realm of process-to-process communications via an Internet Protocol (IP) network using the Transport layer protocols to establish underlying host-to-host connections. In the OSI model, the definition of its Application Layer is narrower in scope, explicitly distinguishing additional functionality above the Transport Layer at two additional levels: session layer and presentation layer. OSI specifies strict modular separation of functionality at these layers and provides protocol for each of them.

### 5.3 FEASIBILITY ANALYSIS

All projects are feasible, given unlimited resources and infinite time. Before going further into the steps of software development, the system analyst has to analyze whether the proposed system will be feasible for the organization and must identify the customer needs. The main purpose of feasibility study is to determine whether the problem is worth solving. The success of a system also lies in the amount of feasibility study done on it. Many feasibility studies have to be done on any system.

### 5.4 OPERATIONAL FEASIBILITY

During feasibility analysis operational feasibility study is a must. This is because; according to software engineering principles operational feasibility/usability should be very high. A thorough analysis is done and found that the system is operational.

### 5.5 TECHNICAL FEASIBILITY

The system analyst has to check the technical feasibility of the proposed system. Taking account of the hardware that is used for the system development, data storage, processing and output, makes the technical feasibility assessment. The system analyst has to check whether the company or user who is implementing the system has enough resource available for the smooth running of the application. Actually the requirements for this application are very less and thus it is technically feasible.

### 5.6 ECONOMICAL FEASIBILITY

Before going further into the development of the proposed system. The system analyst has to check the economic feasibility of the proposed system. Economic feasibility includes the cost for running the system and the cost benefits that can be reaped by implementing the system. In the case of Crypto Media, the development cost is not high as it does not need any extra hardware and software. Thus, the system is economically feasible.

System design is the process of planning a new system to document or altogether replace the old system. The purpose of the design phase is to plan a solution for the problem. The phase is the first step in moving from the problem domain to the solution domain. The design of the system is the critical aspect that affects the quality of the software. System design is also called top-level design. The design phase translates the logical aspects of the system into physical aspects of the system.

This research work discusses the implementation of network simulator tool and its functions, and also the processes involved with step by step explanations.

First, Ubuntu OS. This operating system is a Linux based operating system, wherein the network simulator tool is used. Ubuntu is a Debian-based Linux operating system, with Unity as its default desktop environment. It is based on free software and named after the Southern African philosophy of Ubuntu (literally, human-ness), which often is translated as humanity towards others or the belief in a universal bond of sharing that connects all humanity.

Second, TCL language. TCL (Tool Command Language) is a scripting language, with which we discuss protocols and applications in network simulator tool like AODV, DSR, DSDV and LINK LAYER.

This research work discusses the architecture of NS2, its features, programming structures, multiple layers, trace files and trace analysis. The aforementioned applications are used in the network simulator tool for the implementation of simulation projects in NS2.

## 6. RESULT AND DISCUSSION PERFROMANCE ANALYSIS

This research work has used NS-2 as the network simulator and has conducted numerous simulations to evaluate the performance. All sensor nodes are randomly scattered with a uniform distribution. The location of the sink is randomly determined. This study evaluates the routing performance under scenarios with different numbers of sensor nodes.

The current data suggests that such might have been the case for detection accuracy research. Early researchers created experimental designs that appear to have excluded important types of information. Subsequent researchers systematically built upon earlier designs such that an entire literature developed around variations in a single basic design. Although this literature has certainly advanced knowledge, this knowledge may be much more limited than it might have been. Had early studies started with descriptive work, the literature may well have progressed differently and more efficiently.

The theoretical implications of the findings are also multifaceted. Most obviously, the current data are inconsistent with the primacy of the source behavior assumption that is central to much deception theory. Theories which make this assumption might need rethinking so that it is either avoided or qualified

- *Packet Delivery ratio*: measures the mean rate of the packet sending and receiving, and then calculates delivery ratio. Packet Delivery Ratio compares the proposed high delivery ratio with the existing frameworks.
- *Energy consumption* reduces the usage of energy level in network, and improves energy efficiency rate.
- *End-to-end Delay*: means the time delay experienced by the source node while transmitting a report message to the sink. End-to-End Delay compares the proposed low delay ratio comparing to existing frameworks.
- *Throughput ratio* measures data sharing and successive ratio rate.

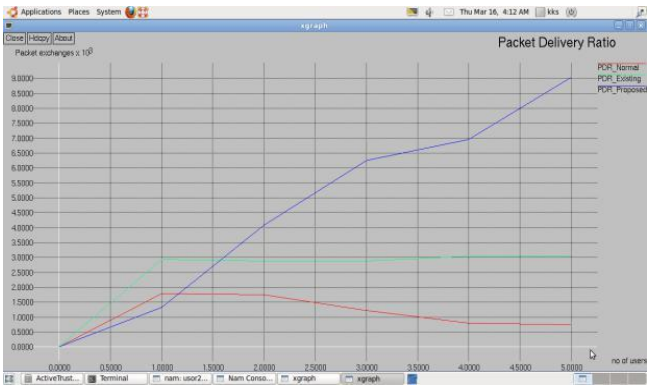


Fig.5. Packet Delivery ratio

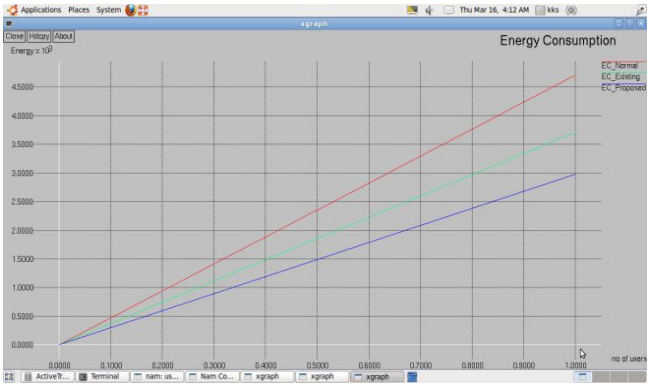


Fig.6. Energy consumption

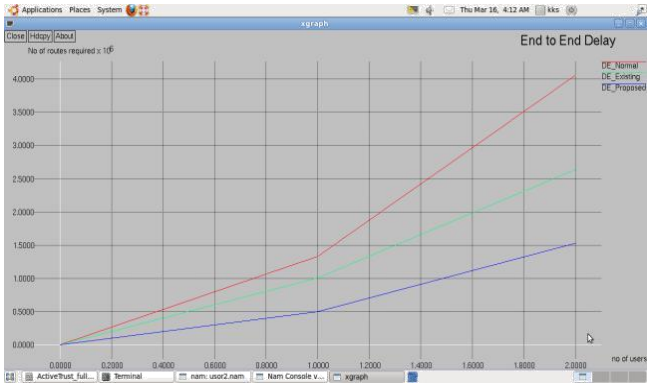


Fig.7. End-to-end Delay

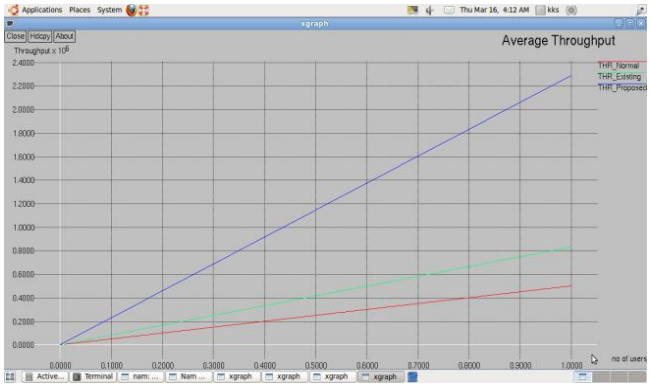


Fig.8. Throughput Ratio

## 7. CONCLUSION

We propose a trust management scheme that enhances the security in networks namely Hybrid and Efficient Intrusion Detection Systems. Here we use two frameworks for Trust Calculation and Decision Making processes. The trust value is derived using Bayesian Inference, and Decision Making is based on Dempster Shafer theory, which is a mathematical theory of evidence. In the proposed process, we add energy efficiency model by using Sleep Wake Scheduling technique in Trust method. We can achieve more energy consumption and high energy efficiency compared to the previous Active Trust model.

The proposed process is a novel security and trust routing scheme based on active detection. It has the following beneficial properties:

1. High successful routing probability, security and scalability. The Trust scheme can quickly detect the nodal trust and then avoid suspicious nodes to quickly achieve a nearly 100% successful routing probability.
2. High energy efficiency. The Trust scheme fully uses residue energy to construct multiple detection routes. The theoretical analysis and experimental results have shown that our scheme improves the successful routing probability by more than 3 times, up to 10 times in some cases. Further, our scheme improves both the energy efficiency and the network security performance. It has important significance for wireless sensor network security.

## REFERENCES

[1] S. Shakkottai, X. Liu and R. Srikant, "The Multicast Capacity of Large Multihop Wireless Networks", *IEEE/ACM Transactions on Networking*, Vol. 18, No. 5, pp. 1691-1700, 2010.

[2] Z. Qian, X. Tian, X. Chen, W. Huang and X. Wang, "Multicast Capacity in MANET with Infrastructure Support", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 25, No. 7, pp. 1808-1818, 2014.

[3] Z. Li, C. Wang, C. Jiang and X. Li, "Multicast Capacity Scaling for Inhomogeneous Mobile Ad Hoc Networks", *Ad Hoc Networks*, Vol. 11, No. 1, pp. 29-38, 2013.

[4] S. Zhou and L. Ying, "On Delay Constrained Multicast Capacity of Largescale Mobile Ad-Hoc Networks", *Proceedings of International Conference on Communications and Networks*, pp. 1-7, 2010.

[5] J.P. Jeong, T. He and D.H.C. Du, "TMA: Trajectory-based Multicast Forwarding for Efficient Multicast Data Delivery in Vehicular Networks", *Computer Networks*, Vol. 57, No. 13, pp. 662-672, 2013.

[6] X. Ge, J. Yang, H. Gharavi and Y. Sun, "Energy Efficiency Challenges of 5G Small Cell Networks", *IEEE Communications Magazine*, Vol. 55, No. 5, pp. 184-191, 2017.

[7] T. Han, X. Ge, L. Wang, K.S. Kwak, Y. Han and X. Liu, "5G Converged Cell-Less Communications in Smart Cities", *IEEE Communications Magazine*, Vol. 55, No. 3, pp. 44-50, 2017.

[8] X. Ge, S. Tu, G. Mao, C. Wang and T. Han, "5G Ultra-Dense Cellular Networks", *IEEE Wireless Communications*, Vol. 23, No. 1, pp. 72-79, 2016.

[9] Z. Su, Q. Xu, Y. Hui, M. Wen and S. Guo, "A Game Theoretic Approach to Parked Vehicle Assisted Content Delivery in Vehicular Ad Hoc Networks", *IEEE Transactions on Vehicular Technology*, Vol. 66, No. 7, pp. 6461-6474, 2016.

[10] B. Yang, Y. Cai, Y. Chen and X. Jiang, "On the Exact Multicast Delay in Mobile Ad Hoc Networks with F-Cast Relay", *Ad Hoc Networks*, Vol. 33, pp. 71-86, 2015.

[11] X. Wang, W. Huang, S. Wang, J. Zhang and C. Hu, "Delay and Capacity Tradeoff Analysis for Motioncast", *IEEE/ACM Transactions on Networking*, Vol. 19, No. 5, pp. 1354-1367, 2011.

[12] C. Hu, X. Wang and F. Wu, "Motioncast: On the Capacity and Delay Tradeoffs", *Proceedings of 10th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 18-21, 2009.