CONTIKI BASED IDS FOR HIGHLY SECURED WIRELESS SENSOR NETWORK

A PROJECT REPORT

Submitted by

VENKATRAMAN.N

Register No: 13MCO22

in partial fulfillment for the requirement of award of the degree

of

MASTER OF ENGINEERING

in

COMMUNICATION SYSTEMS

Department of Electronics and Communication Engineering

KUMARAGURU COLLEGE OF TECHNOLOGY
(An autonomous institution affiliated to Anna University, Chennai)

COIMBATORE - 641 049

ANNA UNIVERSITY: CHENNAI 600 025

APRIL 2015

BONAFIDE CERTIFICATE

Certified that this project report titled **“CONTIKI BASED IDS FOR HIGHLY SECURED WIRELESS SENSOR NETWORK”** is the bonafide work of **VENKATRAMAN.N [Reg. No. 13MCO22]** who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE	SIGNATURE
Mr.R.DARWIN	**Dr. RAJESWARI MARIAPPAN**
PROJECT SUPERVISOR	**HEAD OF THE DEPARTMENT**
Department of ECE	Department of ECE
Kumaraguru College of Technology	Kumaraguru College of Technology
Coimbatore-641 049	Coimbatore-641 049

The Candidate with university **Register No. 13MCO22** was examined by us in the project viva –voice examination held on ....

INTERNAL EXAMINER	**EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

First, I would like to express my praise and gratitude to the Lord, who has showered his grace and blessings enabling me to complete this project in an excellent manner.

I express my sincere thanks to the management of Kumaraguru College of Technology and Joint Correspondent **Shri. Shankar Vanavarayar** for the kind support and for providing necessary facilities to carry out the work.

I would like to express my sincere thanks to our beloved Principal **Dr.R.S.Kumar Ph.D.,** Kumaraguru College of Technology, who encouraged me with his valuable thoughts.

I would like to thank **Dr.Rajeswari Mariappan Ph.D.,** Head of the Department, Electronics and Communication Engineering, for her kind support and for providing necessary facilities to carry out the project work.

In particular, I wish to thank with everlasting gratitude to the project coordinator **Ms.R.Hemalatha M.E.,** Associate Professor, Department of Electronics and Communication Engineering ,for her expert counselling and guidance to make this project to a great deal of success.

I am greatly privileged to express my heartfelt thanks to my project guide **Mr.R.Darwin M.E.,** AP, Department of Electronics and Communication Engineering, throughout the course of this project work and I wish to convey my deep sense of gratitude to all teaching and non-teaching staffs of ECE Department for their help and cooperation.

Finally, I thank my parents and my family members for giving me the moral support and abundant blessings in all of my activities and my dear friends who helped me to endure my difficult times with their unfailing support and warm wishes.

ABSTRACT

Analyzing different algorithms in IDS for memory constrained environment and Implementing IDS in 6LoWPAN. 6LoWPAN is still a new and on-going research area. At this time, there are only a few security solutions proposed for the standard. Cryptography solutions focus on choosing a fast, light-weight and secured encryption, and an effective key management method. Even when 6LoWPAN has an ideal cryptography line defence, there is still a need for implementing an IDS for dealing with network performance threats such as DoS and other resource attacks. The IDS will discover and stop most of the attacks that break cryptography protection to make changes on the network operation.

IDS needs to monitor traffic arriving from both sides. The traffic patterns between the two networks are different, so no single traditional solution from IPv6 or WSN can be applied straight away. In this paper the proposed IDS solution has two modules, one to keep track of the sensor network and the other to check the traffic patterns from the IP network. These two units should cooperate for better performance and resource saving.

To implement an effective IDS system, the various rpl control traffic messages (DIO, DIS and DAO) has to be monitored and compared with the distribution pattern. If any huge deviation found, then that will be consider as abnormal behaviour. So, the anomaly node can be detected and isolated from the network. The power consumption of the network also been analysed as a key parameter to implement IDS in 6LoWPAN.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVATIONS

IoT	Internet of Things
LoWPAN	Low power Wireless Personal Area Network
IDS	Intrusion Detection System
QoS	Quality of Service
DIO	DoDag Information Object
DIS	DoDag Information Solicitation
DAO	Destination Advertisement Object
RAM	Random Access Memory
ROM	Read Only Memory
WSN	Wireless Sensor Network
RPL	Routing Protocol for Low Power and Lossy Network
IP	Internet Protocol
DoS	Denial of Service
FSM	Finite State Machine

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW OF INTERNET OF THINGS

The Internet of Things (IoT) is generally thought of as connecting things to the Internet and using that connection to provide some kind of useful remote monitoring or control of those things. IoT emerged with the idea that, even the tiny devices can be connected to the internet.

As of 2014 the vision of the Internet of Things has evolved due to a convergence of multiple technologies, ranging from wireless communication to the Internet and from embedded systems to micro-electromechanical systems (MEMS). This means that traditional fields of embedded systems(ES), wireless sensor networks(WSN), control systems, automation (including home and building automation), and others, all have contributions to enable the Internet of Things (IoT).

The Internet of Things (IoT) is the network of physical objects or "things" embedded with electronics, software, sensors and connectivity to enable it to achieve greater value and service by exchanging data with the manufacturer, operator and/or other connected devices. Each thing is uniquely identifiable through its embedded computing system but is able to interoperate within the existing Internet infrastructure. The Internet of objects would encode 50 to 100 trillion objects, and be able to follow the movement of those objects. Human beings in surveyed urban environments are each surrounded by 1000 to 5000 trackable objects.

In an Internet of Things, the precise geographic location of a thing—and also the precise geographic dimensions of a thing—will be critical. Open Geospatial Consortium, "OGC Abstract Specification" Currently, the Internet has been primarily used to manage information processed by people. Therefore, facts about a thing, such as its location in time and space, have been less critical to track because the person processing the information can decide whether or not that information was important to the action being taken, and if so, add the missing information (or decide to not take the action).

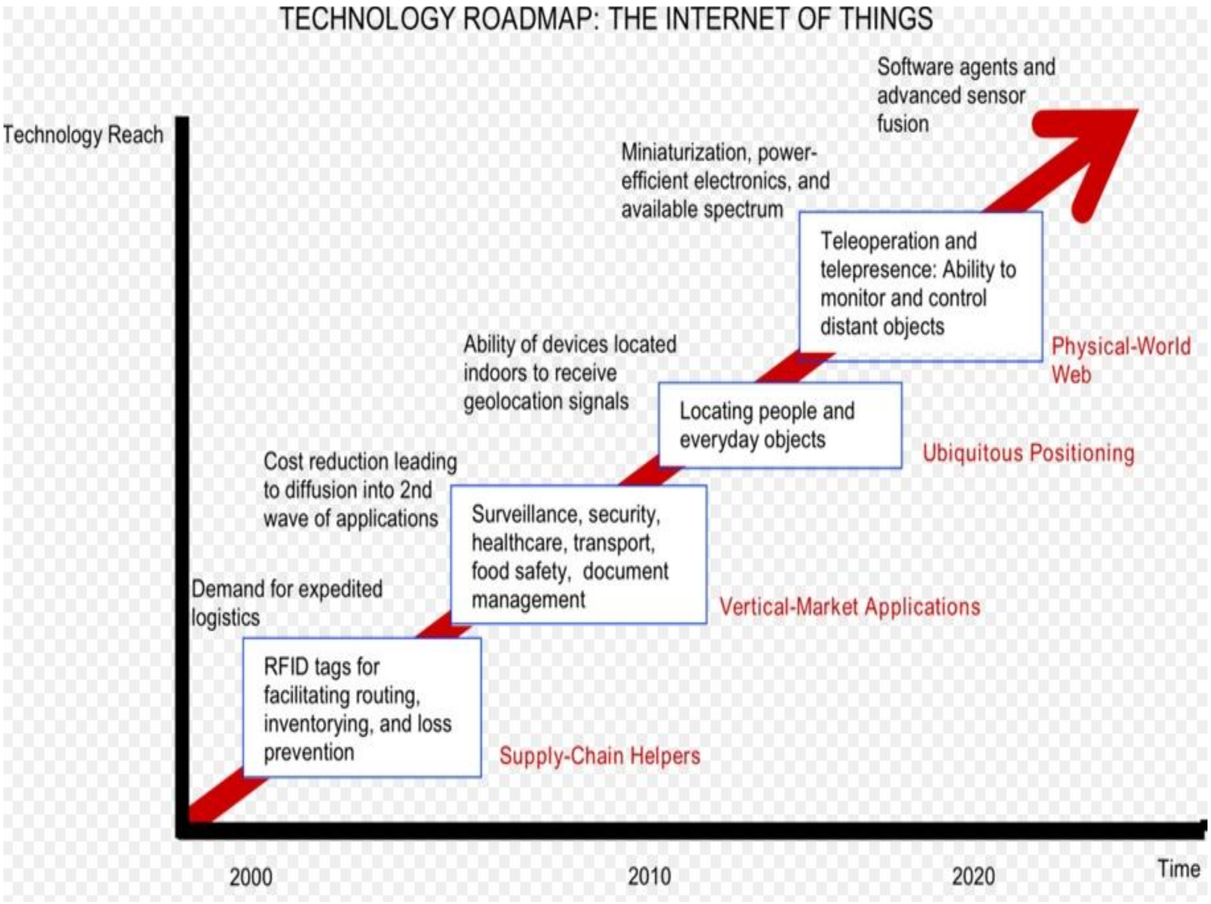

TECHNOLOGY ROADMAP: THE INTERNET OF THINGS
Technology Reach
Software agents and advanced sensor fusion
Miniaturization, power-efficient electronics, and available spectrum
Teleoperation and telepresence: Ability to monitor and control distant objects
Physical-World Web
Ability of devices located indoors to receive geolocation signals
Locating people and everyday objects
Ubiquitous Positioning
Cost reduction leading to diffusion into 2nd wave of applications
Surveillance, security, healthcare, transport, food safety, document management
Vertical-Market Applications
Demand for expedited logistics
RFID tags for facilitating routing, inventorying, and loss prevention
Supply-Chain Helpers
2000
2010
2020
Time

Fig 1.1 Technology road map for internet of things

1.2 CONTIKI

CONTIKI is an open source operating system for networked, memory-constrained systems with a particular focus on low-power wireless Internet of Things devices. Examples of where Contiki is used include street lighting systems, sound monitoring for smart cities, radiation monitoring systems, and alarm systems. Despite providing multitasking and a built-in TCP/IP stack, Contiki only needs about 10 kilobytes of RAM and 30 kilobytes of ROM.

A full system, complete with a graphical user interface, needs about 30 kilobytes of RAM.

Contiki provides three network mechanisms: the uIP TCP/IP stack, which provides IPv4 networking, the uIPv6 stack, which provides IPv6 networking, and the Rime stack, which is a set of custom lightweight networking protocols designed specifically for low-power wireless networks. The IPv6 stack was contributed by Cisco and was, at the time of release, the smallest IPv6 stack to receive the IPv6 Ready certification. The IPv6 stack also contains the RPL routing protocol for low-power lossy IPv6 networks and the 6LoWPAN header compression and adaptation layer for IEEE 802.15.4 links. The Rime stack is an alternative network stack that is intended to be used when the overhead of the IPv4 or

IPv6 stack is prohibitive. The Rime stack provides a set of communication primitives for low-power wireless systems. The default primitives are single-hop unicast, single-hop broadcast, multi-hop unicast, network flooding, and address-free data collection.

Many Contiki systems are severely power-constrained. Battery operated wireless sensors may need to provide years of unattended operation and with little means to recharge or replace its batteries. Contiki provides a set of mechanisms for reducing the power consumption of the system on which it runs. The default mechanism for attaining low-power operation of the radio is called ContikiMAC. With ContikiMAC, nodes can be running in low-power mode and still be able to receive and relay radio messages.

The Contiki system includes a network simulator called Cooja. Cooja simulates networks of Contiki nodes. The nodes may belong to either of three classes: emulated nodes, where the entire hardware of each node is emulated, Cooja nodes, where the Contiki code for the node is compiled for and executed on the simulation host, or Java nodes, where the behavior of the node must be reimplemented as a Java class. A single Cooja simulation may contain a mixture of nodes from any of the three classes.

To run efficiently on memory-constrained systems, the Contiki programming model is based on protothreads. A protothread is a memory-efficient programming abstraction that shares features of both multi-threading and event-driven programming to attain a low memory overhead of each protothread. The kernel invokes the protothread of a process in response to an internal or external event. Examples of internal events are timers that fire or messages being posted from other processes. Examples of external events are sensors that trigger or incoming packets from a radio neighbor. Protothreads are cooperatively scheduled. This means that a Contiki process must always explicitly yield control back to the kernel at regular intervals. Contiki processes may use a special protothread construct to block waiting for events while yielding control to the kernel between each event invocation.

1.3 6LoWPAN

The IoT environment consists of a huge number of devices with resource constraint characteristics such as short radio range, limited processing capability and short battery life. Therefore, the IoT implementation requires a communication protocol that can efficiently manage these conditions. 6LoWPAN is a promising solution with the idea of adding an adaption layer called 6LoWPAN in the network protocol stack for integrating low-power network such as IEEE 802.15.4 into IPv6. This solution can allow the use of the existing infrastructure (Internet Protocol (IP) network) to maximize the utilization of available resources while benefiting from the huge address space of IPv6.

Moreover, the implementation can be accelerated by using tools and mechanisms, such as IPv6 address auto-configuration, to save developers the time and the effort of creating and testing new mechanisms. Because of its open architecture, IoT security problems need to be considered carefully for the standard to be publically deployable. With regard to 6LoWPAN implementation, most of the IoT security threats, coming from 802.15.4, IP network and its adaptation layer, become more detailed and specific. The 802.15.4 part has a weaker secure link than the IP, while its resource-constrained devices are easy to be tampered with and has limited support for security services. Its threats, therefore, can come from both the external and internal attackers and target all the layers. Threats of the IP part, on the other hand, are mostly related to user authentication and data integrity. For example, unauthenticated users can access the information of the LoWPAN part, or falsify the data when sending from the sensor side to the users. Furthermore, the adaptation layer that connects these two parts is also vulnerable to several threats, such as fragmentation attacks, which can make the node run out of resource.

1.3.1 6LoWPAN TOPOLOGY

The 6LoWPAN network consists of one or more local LoWPANs, which are all connected by IPv6 to the Internet through a gateway (or border router). The LoWPAN devices are characterized by short radio range, low data rate, low power and low cost. The network, therefore, deals with small packet size, low bandwidth and requires resource saving for maintaining the life of network nodes.

LoWPAN supports both star and peer-to-peer topology; however, the topology can be changed frequently because of uncertain radio frequency, mobility and battery drain.

In the typical model, IP is the only protocol used to connect different protocols from the data link and physical layer to multiple upper layer protocols. 6LoWPAN, however, utilises the 6LoWPAN stack, a combination of LoWPAN adaptation layer and IPv6, to connect its WSNs to the Internet. The biggest challenge of this combination is to adapt the differences between these two layer packet sizes, which are 1280 octets in IPv6 and 127 octets in LoWPAN. The adaptation layer implemented in the border router is responsible for this mission by fragmentising the packets at the IPv6 layer then reassembling them in 802.15.4 layer. The data link and physical layer of 6LoWPAN using protocols specified for sensor device while the transport layer does not commonly use Transmission Control Protocol because of performance, efficiency and complexity reasons.

In 2008, another IETF working group, Routing over Low-power and Lossy Network (ROLL), was formed to establish a routing solution for such a network. This group proposed RPL (Routing protocol for Low-power and Lossy network), which was later considered the underlying routing protocol for 6LoWPAN. Improving the RPL operation is a critical mission for the network to manage a huge number of nodes with resource constraint characteristic.

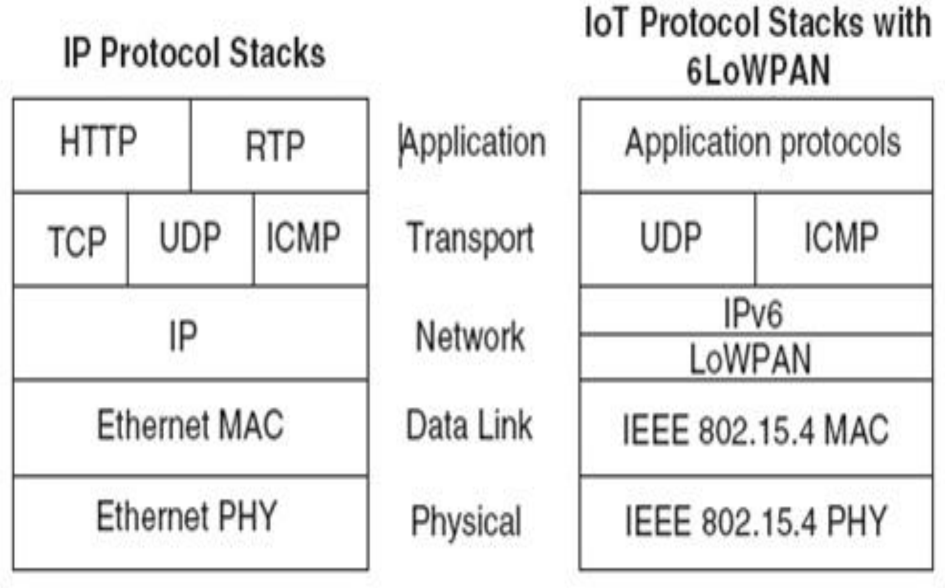


Fig 1.2 Comparison of 6LoWPAN and IP protocol stack

Figure 1.2 shows the difference between the protocol stacks of 6LoWPAN and a typical IP network.

1.4 VULNERABILITIES IN 6LoWPAN SECURITY

6LoWPAN is the combination of the two networks IPv6 and WSN, so security threats from both needs to be examined. There are also threats that aim at the adaptation layer to attack the packet translation process. The operation of 6LoWPAN is represented by the performance of RPL; it is also necessary to analyze the threats towards this protocol.

1.4.1 SECURITY THREATS FROM WIRELESS SENSOR NETWORK SIDE

The security threats of WSN have been extensively studied by the research community. The attacks can be classified by several schemes: outsider–insider adverse source, passive–active, compromising methods, host-based or network-based. From the protecting threat's point of view, detecting the attacks from the outsider and insider requires different protecting systems.

The attackers outside of the network can initiate a passive attack such as unauthorized listening or active attack like denial-of-service (DoS), for example, jamming or power exhaustion. The defense system normally uses cryptography mechanisms to prevent or eliminate outsiders from joining the network. These techniques, however, are not effective when protecting against insider threats. Insider malicious nodes can be created by several ways: attackers physically capture the nodes and reprogram them, attackers use software and devices to breach the cryptography key or inject malicious code. On those cases, the attackers have all the keys, so they can easily overcome any cryptography test.

The insider attacks usually aim at destroying a network operation so it is better to detect them by a well specified monitor system, which can discover early any anomaly network behavior. The outsider and insider attacks are applied on all layers of WSN. Some of these threats are more dangerous because they can easily be deployed and can generate complicated attacks. If the system cannot identify them early, their effects on network operation may be very serious both in short-term and long-term. One example is the Sybil attack, which uses the packet forging mechanism and leads to multiple other attacks like misdirection, exhaustion and unfairness. It will make the WSN unavailable, partitioned or resource exhausted. Another dangerous attack is the Sinkhole, which uses a packet dropping mechanism to attract traffic to a specific node. It generates selective forwarding, black hole attack and combines to partition the network.

1.4.2 SECURITY THREATS FROM INTERNET SIDE

End-users from the Internet can access information from the sensor field once 6LoWPAN is implemented. This raises the threats of authenticating from users and sensor motes, sensor network availability and user accountability. The adversary can access the information illegally if no authentication mechanism is applied in the network. When a communication channel between end-user and sensor network is established, the attacker can also eavesdrop on the sensitive information from the data stream, which breaks the network integrity. Besides that, the accountability of the users accessing the sensor network should be considered for detecting and recreating security incidents. The availability of the communication should be guaranteed by protecting the sensor side and adapting the operation of the Internet side with the resource constraint of the 802.15.4 nature.

Another type of threat is that an attacker from the Internet can get control of the sensor nodes. For example, the botnet attack creates a botnet inside the sensor network for forging the data collection sending to the sink. This attack falsifies the data in the user-end, which leads to wrong alarm or decision. The sensor botnet does not have enough resources for making a successful distributed DoS attack to other networks; however, attackers can make a distributed DoS attack to the botnet itself by flooding to drain the power source. A cryptography line cannot defend against DoS attack from the Internet to the sensor network, so there is a need for implementing the IDS for analysing the IP traffic between the two. Besides that, traditional IDS solutions in the Internet or in the sensor network cannot be simply applied because of the dissimilarity of traffic pattern in these two network designs.

1.4.3 SECURITY THREATS FROM THE ADAPTATION LAYER

The adaptation layer is implemented at the border router for translating the packet between the two networks. The border router is normally a wired node and has strong security protection. However, the packet fragmentation and reassembly progress still have some vulnerability. Kim proposed that fragmentation attack techniques from the IP network can be applied in this layer by modifying or reconstructing the packet fragmentation fields like datagram size, datagram tag or datagram offset. Examples of the threats are Tiny Fragmentation, Ping of Death, Jolt, Teardrop, bank, New Teardrop, or Frag router attack. These attacks can cause critical damage to a sensor node, for instance, reassemble buffer overflow because of packet resequence, exhausting resource because of processing unnecessary fragmentation, or shutting down and rebooting.

1.4.4 SECURITY THREATS FROM ROUTING PROTOCOL FOR LOW POWER & LOSSY NETWORK

The RPL is an underlying and specific routing protocol designed for the purpose of optimizing 6LoWPAN operation. There are security mechanisms proposed for RPL but they only aim at protecting it from external threats by control messages encrypting countermeasures. The drawbacks of 6LoWPAN security, such as weak communication link and nontampering nodes, make RPL weak from internal attack. Once a benign node becomes an internal adversary, it can break the network operation without being detected by cryptography mechanisms. Therefore, analyzing RPL threats in addition to specifying its operation will help to monitor most of the internal malicious behaviors.

Current RPL threats directly attack the routing operation by changing the route, making it longer or even changing the destination address so that the time waiting for a packet goes to indefinite. Threats on other layers that aim at resource consuming such as flooding and overwhelming, or destroying network traffic like jamming or congestion can also be considered indirect attacks to the routing part because they downgrade the node operation. RPL is also vulnerable from passive eavesdropping attacks and active tampering. The passive eavesdropping attacks can be prevented by using a symmetric key to encrypt the packets as proposed in. Tampering active nodes, however, creates compromised nodes, which can cooperate to break the protocol operation rules and easily overcome the cryptography line. Besides that, RPL utilizes some specific rules for optimizing network operation; nevertheless, adversaries can exploit these to create different attacks. Potential attacks of this kind are Rank, local repair and resource depletion attack.

1.5 6LoWPAN SECURITY COUNTERMEASURES

To be publicly accepted, 6LoWPAN needs to have a strong security defense system. However, security techniques from other networks cannot be straightly applied in 6LoWPAN because of many of its specific constraints. WSN is the network that has the nearest nature to 6LoWPAN therefore, WSN security mechanisms are preferred to be utilized in that network. This section summarizes security requirements and examines the most prominent techniques that might be helpful in applying for 6LoWPAN.

1.5.1 6LoWPAN SECURITY REQUIREMENTS

The RFC4919 specifies a list of security requirements for 6LoWPAN, which mainly aim at protecting the communications from the end-users to the sensor network. The requirement list is:

- Confidentiality: only authorized users can access the information
- Authentication: data is only originated from a trusted sources
- Integrity: the received data remains unchanged during transmission
- Freshness: consider for both data and key to ensure no replayed of old messages
- Availability: guarantee the data can be accessible when needed
- Robustness: providing operation despite the abnormal conditions
- Resiliency: provide an acceptable level of security even in the case some nodes are compromised
- Energy efficiency: reduce the control overhead to maximize network lifetime
- Assurance: the ability to disseminate different information

These requirements require the combination of different securing systems. Cryptography is considered the first line for solving the confidentiality, authentication and integrity. This system, however, cannot solve other QoS securing requirements like availability, robustness and resiliency. It therefore needs to cooperate with the IDS, which can monitor and detect malicious sources from the early phase to eliminate further damage of the attacks.

CHAPTER 2

LITERATURE SURVEY

2.1 INTRODUCTION

This chapter presents the literature surveyed in the area of Intrusion Detection System. The merits and demerits of these methods are discussed in terms of complexity, performance, and speed of computation. The purpose of this is to choose the appropriate technique for detecting the anomaly node and alert the cryptographic system to regenerate the keying process.

2.1.1 SPMOS-based Intrusion Detection Architecture

IDS (Instrusion Detection System) has been designed to protect systems from being compromised by network attacks. A lot of researches have been done on it. However, most of them focus on complex and time-consuming detection methods to improve accuracy of the system, with assumption that IDS is running under control of general purpose operating systems (GPOS). In this way, the IDS itself will depress overall performance and cannot be guaranteed secure. In this paper, an embedded architecture of SPMOS-based IDS has been presented. SPMOS, located in SPM, is a little OS running under GPOS. Experiment results show that the architecture is fast. Based on this, we also design a simple IDS and conduct tests by integrating it into SPMOS and GPOS. The former consumes the latter’s 8.3% time only, with less than 6.2% overhead, which verifies the architecture proposed is practical and efficient.

2.1.2 Network intrusion detection using rough sets based parallel genetic algorithm hybrid model

The thesis proposes a hybrid intrusion detection model based on the parallel genetic algorithm and the rough set theory. Due to the difficult for the status of intrusion detection rules. This model, taking the advantage of rough set’s streamline the edge to data and genetic algorithm’s high parallelism, succeeds in introducing the genetic-rough set theory to the intrusion detection. The application of hybrid genetic algorithm in solving the rough set reduction saves computing time. The concludes that model can result in high detection rate and low false detection rate to different types of network via experiments.

2.1.3 IDS Alert Classification Model Construction Using Decision Support Technique

Although many IDSs have been proposed to assist administrators in detecting intrusion, false alarms are still huge and result in the difficulty of analysis. For this reason, we proposed a decision support system for constructing an alert classification model, which consists of three phases: alert preprocessing, model constructing and rule refining. Experimental results show that the proposed method discovers intrusion patterns quickly and precisely, and lightens the load of on-line alert analysis for experts obviously.

2.1.4 The Model - Dynamic and Flexible Intrusion Detection Protocol for High Error Rate Wireless Sensor Networks Based on Data Flow

A new Dynamic Intrusion Detection Protocol model (DYDOG) has been designed based on data flow for High Error Rate Wireless Sensor Networks (WSNs). Here the Dynamic Intrusion Detection nodes are deployed based on the proposed protocol model which will acts as forwarding node as well as Intrusion Monitoring Node with respect to the data flow through the sensor nodes. The Dynamic Intrusion Detection Nodes are selected from the one-hop or two hop neighbor's non-forwarding node list by using Secure Session Key Management approach without deploying separate Intrusion Monitoring Nodes. This makes the network is more flexible and dynamic against various attacks and provide maximum monitoring node's availability with better resiliency in high error rate Wireless Sensor Networks (WSNs).

2.1.5 A New Collaborative Approach for Intrusion Detection System on Wireless Sensor Networks

This paper proposes a new collaborative and decentralized approach for intrusion detection system. Special nodes, called monitors, will be responsible for monitoring the behavior of neighbor nodes. The malicious activities evidences discovered by each monitor will be shared and correlated with the purpose of increasing the accuracy in detection of intruders. Experiment conducted by simulation show that our solution is effective in reducing the false positives.

The monitors nodes will be responsible for watch the entire network, in a distributed fashion. Each monitor, located somewhere in the network, will be in charge for monitoring a sub-part of the network, the nodes neighbors to it. From the traffic of neighbor nodes, the monitor can infer which ones are behaving out of expected. This inference is possible, because the system stores a set of rules that specify the nodes normal behavior.

2.1.6 Group-based intrusion detection system in wireless sensor networks

This paper proposed a distributed group-based intrusion detection scheme that meets all the above requirements by partitioning the sensor networks into many groups in which the sensors in each group are physically close to each other and are equipped with the same sensing capability. This intrusion detection algorithm takes simultaneously into consideration of multiple attributes of the sensor nodes to detect malicious attackers precisely. This algorithm can decrease the false alarm rate and increase the detection accuracy compared with existing intrusion detection schemes while lowering the computation and transmission power consumption.

2.1.7 A Collaborative, Secure and Energy Efficient Intrusion Detection Method for Homogeneous WSN

This paper deals with selection of a set of trusted nodes and does the intrusion detection only with this set of nodes. If the detection is carried out by other types of nodes, this method proposed a multi-detection model in order to secure the process and to analyze the probability of intrusion detection with these set of nodes. This proposed selection algorithm helps in energy efficiency as the information is routed only through these set of nodes. This method is more secure as the detection process is carried out by sensor nodes which are trusted.

2.1.8 Lightweight Intrusion Detection for Wireless Sensor Networks

In our IDS architecture, every node belongs to a single cluster among the clusters which are geographically distributed across the whole network. Our aim is to utilize cluster-based protocols in energy saving, reduced computational resources and data transmission redundancy. In this section, we propose an intrusion framework for information sharing, which utilizes hierarchical architecture to improve intrusion detection capability for all participating nodes. In our scheme, an IDS agent is located in every sensor node. Each sensor node has two intrusion modules, called local IDS agent and global IDS agent. Because of the limited battery life and resources, each agent is only active when it is needed.

CHAPTER 3

INTRUSION DETECTION SYSTEM

3.1 INTRODUCTION

The intrusion detection system is a well-known network security approach. The main idea behind IDS is to collect the network data and analyze any sign of the attack to raise an alarm and discover the adverse resource. The development of technology has changed the communication environment from wired, wireless, ad hoc to sensor network recently. IDS solutions have also changed from data collection and analysis techniques to adaptation to the implemented environment. The nature of WSN is different from other networks in terms of device communication ability and resource available. IDS applied in WSN should optimize the features and computational work for saving network resource. With regard to 6LoWPAN, the optimization ability of the IDS is even more required because of the network scalability.

3.2 IDS APPROACHES

The IDS approaches are often divided by misuse, anomaly-based and specification-based type. A misuse IDS first defines patterns of the known attacks, and when monitoring the network, if it discovers any data that match the pattern, it will raise a security alarm. This method can provide low-false alarm rate, but it needs to store a lot of data to be analyzed, requires the attacks to be well defined and limits in detecting the new attacks. This approach is not favored in WSN or 6LoWPAN because the knowledge about attacks is not well-studied, security resource is constrained, and the network requires the ability to detect novel attacks. Another method, anomaly-based IDS, focuses on classifying the normal network behaviors, then monitors and compares to detect any anomalous activities. The method computes the deviation between the monitored data and the pattern, and if the deviation exceeds a threshold, it will raise an alarm. Anomaly-based IDS has the ability to detect new attacks if these attacks make any change to network operations. It also does not consume many resources. However, the false-alarm rate is still high because the system cannot differentiate between misbehavior and malicious operation. Specification-based IDS specifies the normal operations of the network in detail and monitors any breaking of this description.

The operation patterns are usually created by specialists, so the false detection rate is decreased a lot compared to the anomaly method. Specification-based IDS also has the ability to detect new attacks, if these attacks make the network operations different from the patterns. The disadvantages of this method are that it needs the definition from specialist and it is inflexible in upgrading. The current trend in IDS research is to combine these methods for having more accuracy and more functions.

3.3 APPLICATION OF IDS IN 6LoWPAN

6LoWPAN is still a new and on-going research area. At this time, there are only a few security solutions proposed for the standard. Cryptography solutions focus on choosing a fast, light-weight and secured encryption, and an effective key management method. Even when 6LoWPAN has an ideal cryptography line defense, there is still a need for implementing an IDS for dealing with network performance threats such as DoS and other resource attacks. The IDS will discover and stop most of the attacks that break cryptography protection to make changes on the network operation. However, no IDS solution has been proposed for 6LoWPAN security. This part takes the natural characteristics of 6LoWPAN to analyze the difference to other networks to clarify a 6LoWPAN IDS. 6LoWPAN combines 802.15.4 and IPv6 so its IDS needs to monitor traffic arriving from both sides. The traffic patterns between the two networks are different, so no single traditional solution from IPv6 or WSN can be applied straight away. The IDS solution should have two modules, one to keep track of the sensor network and the other to check the traffic patterns from the IP network. These two units should cooperate for better performance and resource saving.

3.3.1 IDS ISSUES IN WIRELESS SENSOR NETWORK PART

Intrusion detection system solutions in WSN have to be light weight and low work load because of the resources constraint of the nodes. Their main issues are

(i) The feature extraction: the issue in choosing the right features for reducing the monitored data and effectively detects the attacks

(ii) The placement problems: where to put the IDS agent in the network for an optimized operation

(iii) The data analysis techniques: choosing a technique to increase accuracy and decrease the computational work.

To detect WSN attacks, a number of data features were proposed by Da Silva *et al* to monitor the parameters as follows:

(i) The time between two consecutive messages for detecting the negligence (sending message too slowly) or exhaustion (sending message too quickly).

(ii) Payload: for discovering integrity attacks, which makes changes on payload.

(iii) Delay: detect attacks that make high delay in sending the messages such as black hole or selective forwarding.

(iv) Repetition: detect DoS attack.

(v) SenderID: for detecting wormhole, Helloflood attack -this parameter can also be applied in discovering Neighbour Discovery attacks of IPv6 and Sybil, which create a strange SenderID.

(vi) Number of collisions: detect attacks that cause large number of collisions such as jamming attacks.

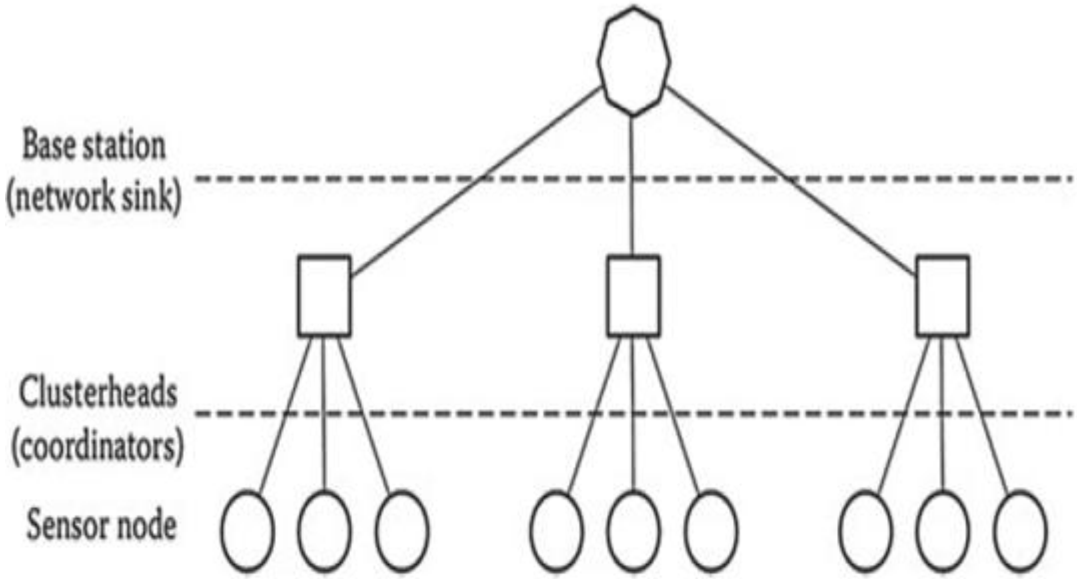


Fig 3.1 Hierarchical approach in putting IDS on WSN

Analyzing threats for choosing parameters to monitor is extremely necessary in 6LoWPAN because different LoWPAN networks connected to the IPv6 will have variant characteristics and be deployed in distinguishable environments so it will have distinct priority security objectives. Where to put the IDS agents in WSN is also an issue that needs to be considered. The network based approach, which puts the agent on the base station to receive and analyze all the monitored data from the nodes, can utilize the strong resource ability of the base station. Another good thing is that it can use the global view to detect cooperation attacks. On the other hand, this architecture creates a lot of communication overhead and is bad at detecting local attacks.

In the host-based approach, IDS agents are implemented in every node. Nodes monitor, analyze the monitoring data and decide themselves. This method can reduce the monitored traffic but put more computational work to consume node resources and shorten its lifetime. The approach can detect local attacks accurately, but it lacks global view for protecting cooperation attacks.

3.3.2 IDS ISSUES IN IPv6 PART

The IDS from IPv6 side is to protect the border router from any threats that send packets from IPv6 to WSN to start a WSN attack. Most of the issues in WSN parts are easy to solve in the IPv6 part because the border router is usually implemented with strong security and nonresource constraint and moreover, the threats that come from the IPv6 network are much less than threats inside the sensor network. For instance, the border router is the most suitable position to put the IDS agent because it is the place where the traffic between the two networks goes. The feature extraction issue is also not restricted like in the WSN part because of the high capacity of the border router. The only issue that needs to be focused on is choosing suitable IDS techniques for detecting threats early and accurately.

Again three types of methods: misuse, anomaly and specification-based can be applied. The misuse direction is still not favourable because no attack signatures are defined.
An IDS can be considered as the combination of anomaly and misuse techniques.
It uses the three techniques: Anderson-Darling Algorithm, Entropy Algorithm and PAT (Predefined Attack Types) calculator for detecting the abnormal behaviors.

The chosen data feature is the discard packets from the congestion avoidance algorithms when the queues are full. To reduce the false alarm rate, they bring the discovered anomaly data to a pattern classifier, which checks the predefined attack type on the stored buffer. A threshold is also chosen for generating a security alert once it is detected to be passed by the classifier. This system requires a lot of computational loads with the three checking modules and another matching part so it will reduce the detection speed. The author did not explain why they chose to analyze only the data, which is discarded from the buffer. By doing that they probably assumed the data that passed to the buffer are attack-free while there is no guarantee in reality. The main architecture of this system, however, can still be applied with different detecting techniques for a better solution.

3.4 PROPOSED SOLUTION

In our view, a normal network performance can only be guaranteed if (i) the network is in its optimized topology; (ii) every node works with its reasonable capability; and (iii) the system has malicious resource trace-back ability to trace and eliminate attack resource after detection. The energy issue because of adding IDS to the network can be solved by, for example, utilizing the hierarchical monitoring architecture to minimize the computation work load in the monitoring nodes and the communication overhead over the network or choosing lightweight IDS techniques. The system that we envisage therefore has three main parts for satisfying these conditions:

(i) the RPL specification-based IDS to monitor 6LoWPAN optimized topology

(ii) the anomaly-based used in cooperation with specification-based to monitor the node performance and

(iii) the statistical-based component to reveal the attacker source.

The system model, which can serve as the baseline for researchers to move on in this field, is shown in Figure.3.2.

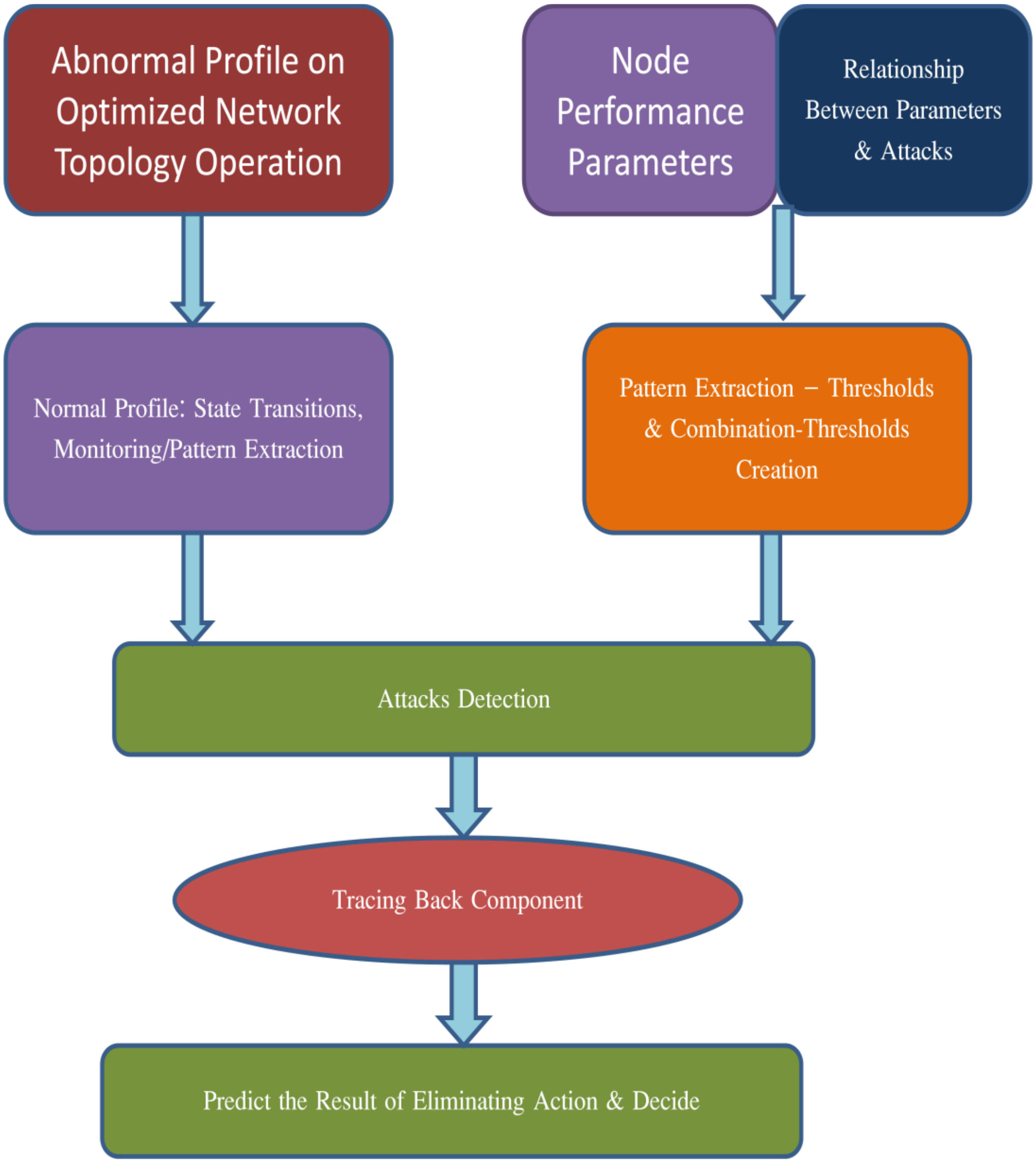

Abnormal Profile on Optimized Network Topology Operation
Node Performance Parameters
Relationship Between Parameters & Attacks
Normal Profile: State Transitions, Monitoring/Pattern Extraction
Pattern Extraction – Thresholds & Combination-Thresholds Creation
Attacks Detection
Tracing Back Component
Predict the Result of Eliminating Action & Decide

Fig 3.2 Proposed IDS for securing 6LoWPAN

3.4.1 RPL SPECIFICATION BASED IDS COMPONENT

RPL is the underlying routing protocol for 6LoWPAN, so building a specification based IDS for RPL is one of the most efficient way to detect fast and accurately any 6LoWPAN attacks that break its optimized topology set up. Initial work on securing RPL is focuses on protecting RPL control messages (DIO, DIS and DAO) and the routing information in IPv6 Hop-by-Hop Option Header and Routing Header.

The suggested RPL security objectives should be,

(i) participants of the DIO, DIS and DAO message exchanges are authenticated

(ii) the received DIO, DIS and DAO messages are not modified during transportation

(iii) the received DIO, DIS and DAO messages are not retransmissions of previous messages and

(iv) the content of the DIO, DIS and DAO messages may be made legible to only authorized entities.

Their solution focuses on adding encryption mechanism for those control messages. However, they lack the ability in detecting internal attackers that break the protocol operation. The IDS will collect the RPL routing information and check the state of control messages transmissions to detect the routing attacks. Each monitor node will observe the communications between the monitored nodes to extract the topology information, mostly the parent–child relationship, to detect anything invalid in the topology. Besides that, any topology change will be remembered, and a threshold is defined so that if a node creates too many changes in its relationships, an alarm will be raised. On the other hand, monitor nodes also listen to the behaviours of their monitored members, so that any dropping, delaying or overusing of control messages can be detected.

A possible way to improve the accuracy of the system is to monitor the FSM (Finite State Machine) states and transitions between them, because they also have patterns when following the normal operation of the network. The Bayesian network technique with the advantage of predicting future transition to classify anomaly transition is a good candidate to be investigated. For example, in normal condition, a state transition happens only several times a minute. However, if that transition is replied again for many times exceeding the threshold suggested by the Bayesian model, although it does not break the specification model, it still can be considered a new threat. Incorporating such technique in the RPL specification-based IDS will help detect unknown attacks that violate RPL’s routing rules. For example, consider the Rank attack, whose nature is to create illegal communication between nodes with lower rank and nodes with higher rank. When the monitor system analyses the transmission between those malicious nodes and checks the validation of the topology, it will easily discover any operation that breaches the rule and thus raise an alarm.

3.4.2 ANOMALY BASED IDS COMPONENT

The RPL specification-based IDS alone is not enough to protect 6LoWPAN operation. This is because the monitoring system will have no clue to detect threats that are not violating the RPL's routing rules. For instance, a benign malicious node initiates data packet dropping; such attacks disrupt the network QoS without violating the RPL's rules. Therefore, anomaly-based IDS should be added to the system, which will include data mining (i.e. network parameters such as delivery rate, end-to-end delay, throughput, etc.) and data fusion techniques to detect anomalous behavior in the network. The anomaly-based IDS will analyze the relationships of those network parameters in the event of attacks as symptoms to diagnose the attacks or to prevent future attacks.

For example, the wormhole attack can decrease the packet delivery ratio and increase the end-to-end delay. Evaluating the two parameters individually may be seen as acceptable in a given range; however, evaluating the parameters in a combined manner may indicate a threat. Therefore, statistical and probability techniques should be used for analysing the relationship between network parameters for potential threats. Bayesian network, strong at inferring the network parameters and their relationships, is one of the best techniques that can be used. The model can be asked to give probability of specific attacks like black hole or wormhole, given the network performance monitored data such as delivery ratio and end-to-end delay. This probability inferring can be used to create a combination threshold for improving the accuracy of the anomaly-based IDS part.

Once implemented, the second part can protect 6LoWPAN from any threats that use malicious nodes to downgrade the node QoS in the network. It is necessary to say, without the first part, that the second part cannot provide good security on network operation, because attackers can downgrade the performance just by setting up bad topology while still letting the nodes work with their best performance. Therefore, these two protection parts, one tracking the optimised topology and one monitoring the node performance, will work in cooperation to provide a robust security for 6LoWPAN operation.

3.4.3 TRACKING BACK COMPONENT

This problem can be analyzed using statistical techniques like the Bayesian operation model. The model will attempt to predict the network nodes that behave maliciously by analyzing statistical data from the monitoring nodes in the network and the detection outcomes from the previous two components. To sum up, our foresight 6LoWPAN security system operates in two-layer cooperative detection as illustrated in Figure.

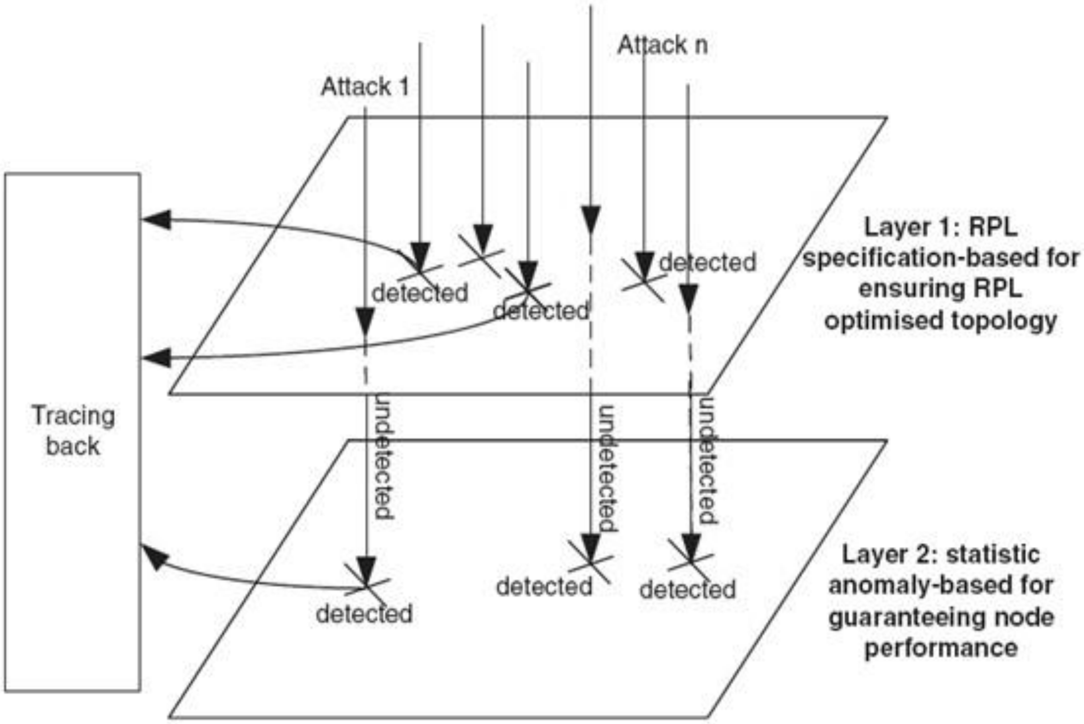

Attack 1
Attack n
detected
detected
detected
Layer 1: RPL specification-based for ensuring RPL optimised topology
Tracing back
undetected
undetected
undetected
detected
detected
detected
Layer 2: statistic anomaly-based for guaranteeing node performance

Fig 3.3 IDS protection layers for securing 6LoWPAN

Layer 1 is the RPL specification-based IDS for detecting all the threats that violate RPL's operation rules for ensuring optimized network topology, while Layer 2 is the anomaly-based IDS that ensure node performance. After detecting the malicious behaviors, the system will attempt to trace back and eliminate the attacker node, to ensure protection of the network QoS. We believe that this system can provide a robust security countermeasure for the 6LowPAN. The system can be expanded more by adding other protection layers, but they should work in cooperation with previous layers. The new protection layer should only aim at the attacks that cannot be detected by previous layers.

3.4.3 WORK FLOW

The network setup will run using COOJA simulator and all the mote outputs are stored in a specific output log file. The figure. 3.4 shows the work flow diagram.

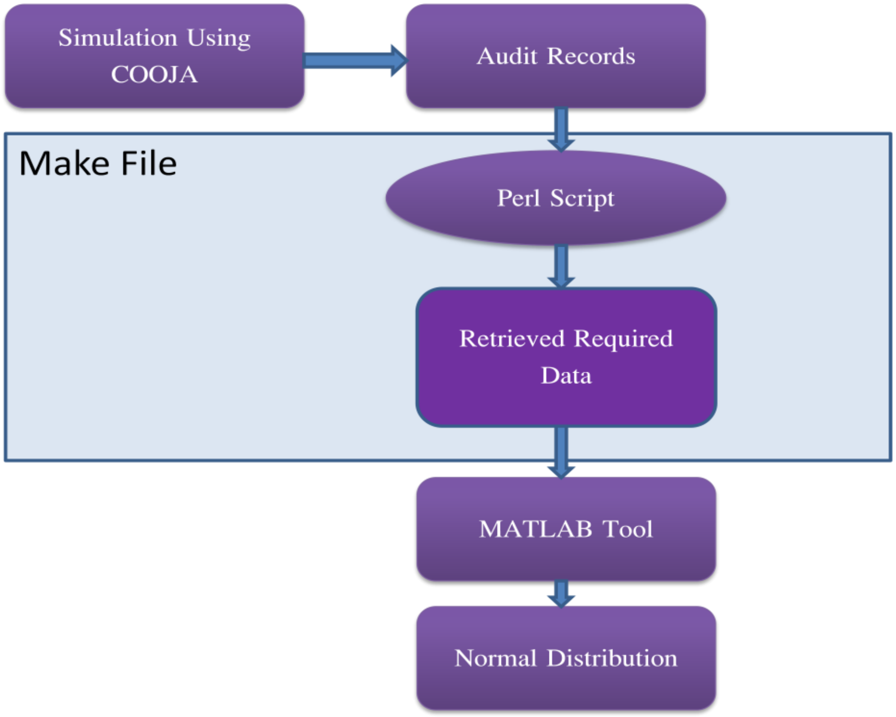


Fig: 3.4 Work flow chart

The log output will contain all the data records of the network. The audit record required to detect the intrusion can be selected out using perl script. This perl script will run through the make file. So the required data can be retrieved from the log output. This audit data is given as input to the MATLAB tool from which the distribution pattern of the audit data has been obtained.

CHAPTER 4

SIMULATION RESULTS

4.1 RESULT AND DISCUSSION

4.1.1 COOJA Setup:

This is the screen shot of the network setup in contiki. The mote-1 is the root node. The green surface surrounding the mote-1 is the coverage area of that mote. The motes outside the coverage area will have to take more than one hop to reach the root node.

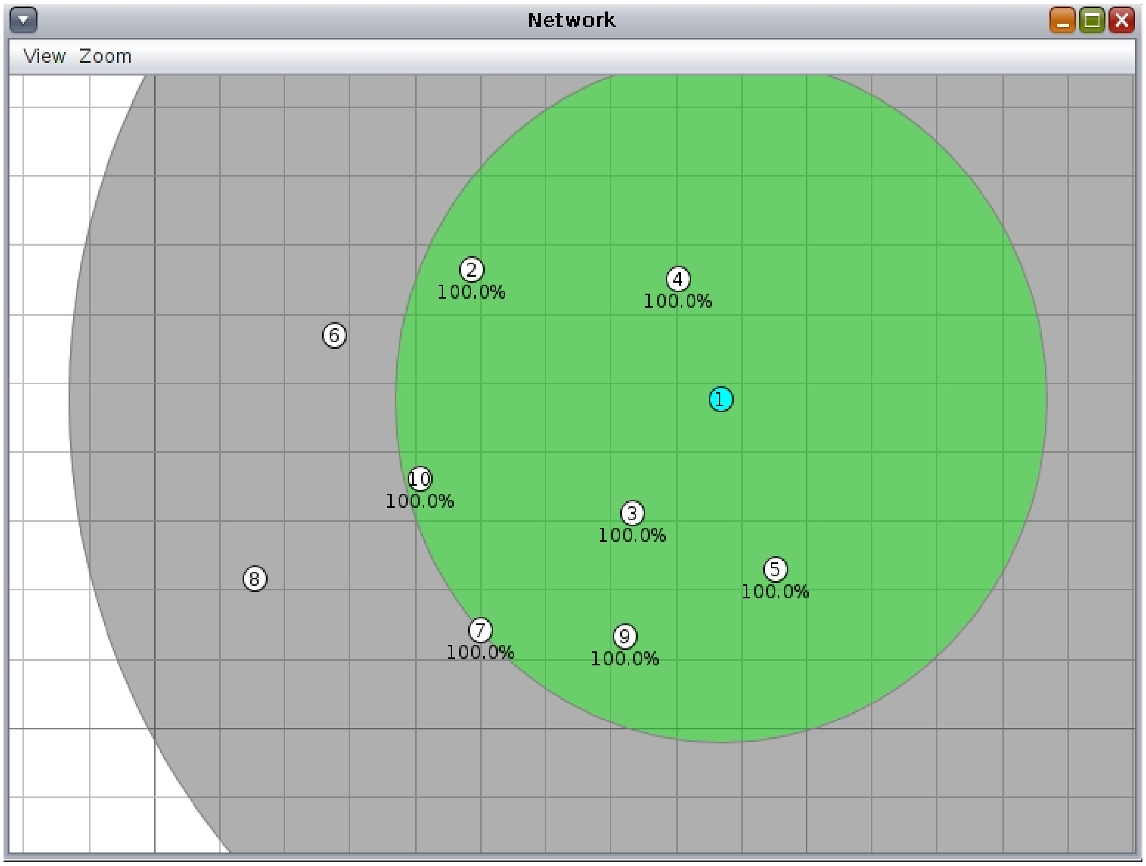
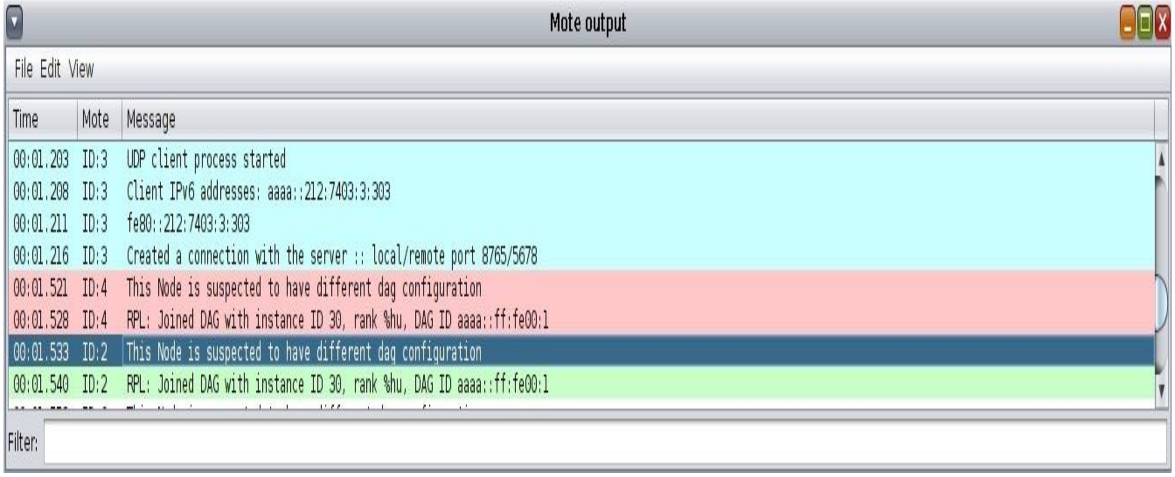


Fig 4.1 Network setup in cooja simulator

The mote output screen displays the output of all the motes in the network setup. This is the screen shot of mote output. The current screen shot shows the detection of the anomaly node by mote-4.



```
Time       Mote  Message
00:01.203  ID:3  UDP client process started
00:01.208  ID:3  Client IPv6 addresses: aaaa::212:7403:3:303
00:01.211  ID:3  fe80::212:7403:3:303
00:01.216  ID:3  Created a connection with the server :: local/remote port 8765/5678
00:01.521  ID:4  This Node is suspected to have different dag configuration
00:01.528  ID:4  RPL: Joined DAG with instance ID 30, rank %hu, DAG ID aaaa::ff:fe00:1
00:01.533  ID:2  This Node is suspected to have different dag configuration
00:01.540  ID:2  RPL: Joined DAG with instance ID 30, rank %hu, DAG ID aaaa::ff:fe00:1
```

Fig 4.2 Mote output in the cooja simulator

4.1.2 DIO Message Distribution:

There are three types of control messages. They are,

- DIO (DODAG Information Object)
- DIS (DODAG Information Solicitation)
- DAO (Destination Advertisement Object)

The DIS is used to solicit DIO from neighboring nodes. Through DIO the node learns the required parameters to join the DAG and then it sends a DAO upwards along the route. In order to implement IDS, the analyses of these control messages are vital. The intruder node if any will try to solicit the information object from the neighboring nodes quite often compared to authentic nodes. The amount of control messages varies according to the configuration in ContikiRPL. Initially we analyze these control messages for the default configurations in ContikiRPL. The analysis will mainly comprise of total number of each control messages and the frequency in which each control messages are sent. Based on the results we define the threshold in the monitoring mote, which can be used to verify the abnormal patterns of the monitored motes.

The frequency of DIO messages are obtained from the audit record. The distribution for the frequency of DIO messages is shown in the figure. 4.3. If DIO messages received fall consistently outside this interval then it is considered as an abnormal behaviour. This figure has been obtained for the DIO traffic messages of 20 motes. The distribution graph has been generated for the data acquired from two hours of simulated results in mote output.

From the graph, the maximum numbers of DIO messages are distributed in the interval time of 600 to 1000 seconds. If any node sends more DIO messages within the interval time of less than 600 seconds and beyond 1000 seconds, then that particular node will be suspected as malicious. This node will be monitored from the monitoring node. if the malicious activity of the node has confirmed, then it will be eliminated from the network.

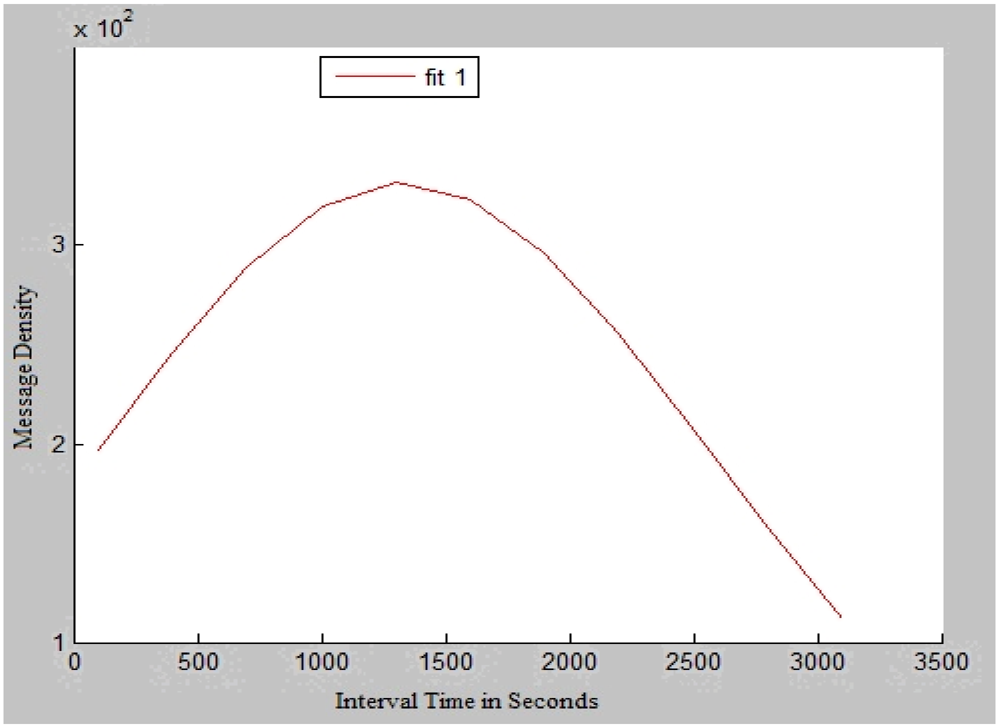


Fig 4.3 Distribution of DIO messages for 20 nodes

The figure. 4.4 has been obtained for the DIO traffic messages of 40 motes. The distribution graph has been generated for the data acquired from two hours of simulated results in mote output.

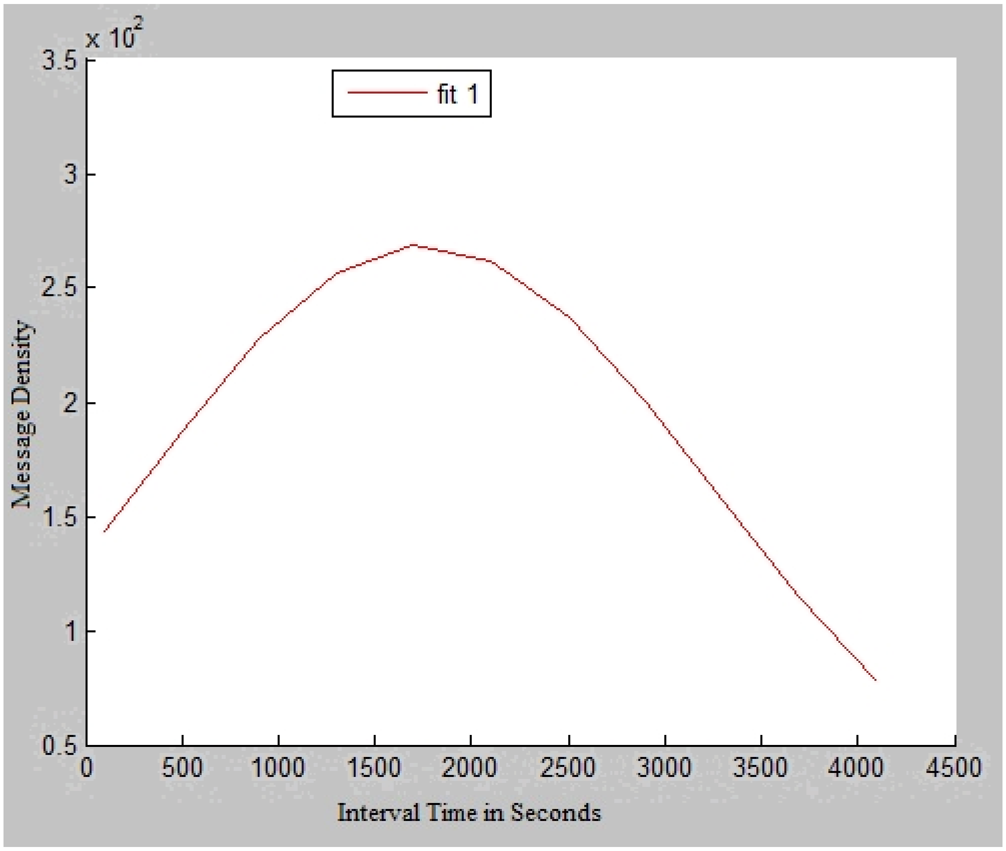


Fig 4.4 Distribution of DIO messages for 40 nodes

From the graph, the maximum numbers of DIO messages are distributed in the interval time of 1000 to 1500 seconds. If any node sends more DIO messages within the interval time of less than 1000 seconds and beyond 1500 seconds, then that particular node will be suspected as malicious. This node will be monitored from the monitoring node. if the malicious activity of the node has confirmed, then it will be eliminated from the network.

The figure. 4.5 has been obtained for the DIO traffic messages of 50 motes. The distribution graph has been generated for the data acquired from two hours of simulated results in mote output.

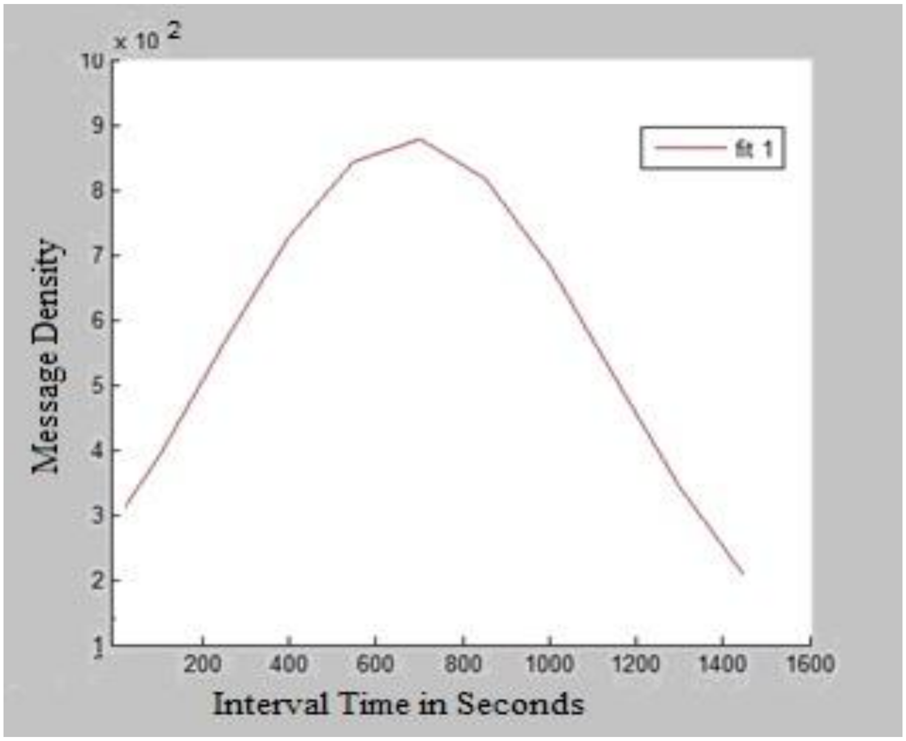


Fig 4.5 Distribution of DIO messages for 50 nodes

From the graph, the maximum numbers of DIO messages are distributed in the interval time of 1000 to 1600 seconds. If any node sends more DIO messages within the interval time of less than 1000 seconds and beyond 1600 seconds, then that particular node will be suspected as malicious. This node will be monitored from the monitoring node. if the malicious activity of the node has confirmed, then it will be eliminated from the network.

4.1.3 DIS Pattern:

Normally the first message a mote sends after startup is a solicitation message (DIS) to get the information object from the neighboring mote (DIO). Since DIO is a broadcast message, the solicitation request from any mote will trigger the receiving mote to broadcast DIO. If a mote receives a broadcasted DIO before sending a solicitation message the mote will not send the solicitation message. So, the result of the analysis is that when a network is setup few motes broadcast DIS which triggers DIO that are received by other motes, even before they solicit for information. So, only a few motes sends DIS and other motes do not send DIS. If the mote is an intruder, in order to gain more information from the network, the solicitation message will be continually broadcasted, which is not a normal behavior. The distribution for DIO message and

DIS message count is shown in the figure. 4.6. From the simulation results, the number of DIO messages received by all the motes is same except the server mote.

The figure. 4.6 is obtained from the simulation results of 20 nodes. From the figure, except the server node, all the client nodes are sent equal number of DIO messages. In this simulation result, the server node sends minimum number of DIO messages. In the view of DIS messages, only a few nodes are sent DIS message.

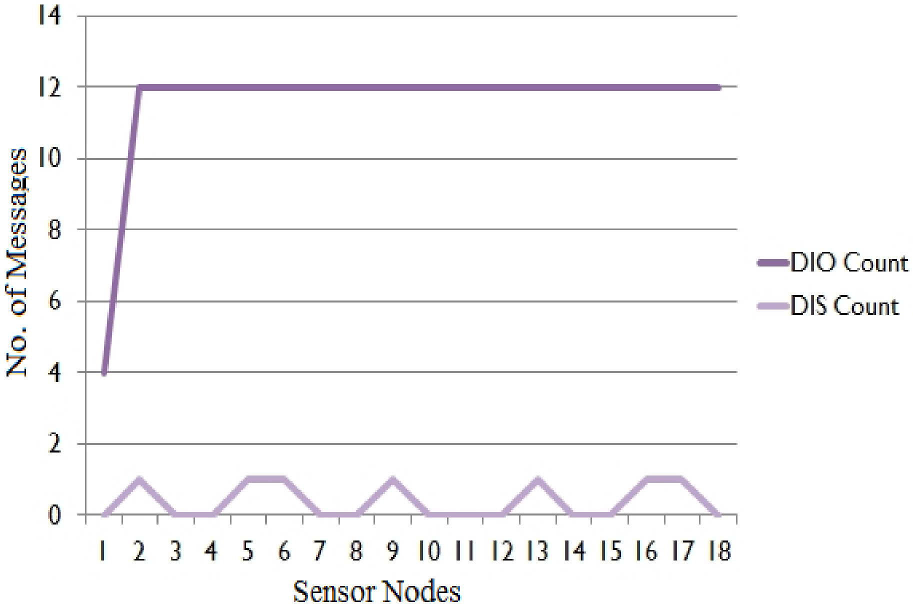


Fig 4.6 Number of DIO & DIS messages sent by 20 nodes

The figure. 4.7 is obtained from the simulation results of 40 nodes. From the figure, except the server node, all the client nodes are sent equal number of DIO messages. In this simulation result, the server node sends minimum number of DIO messages. In the view of DIS messages, only a few nodes are sent DIS message.

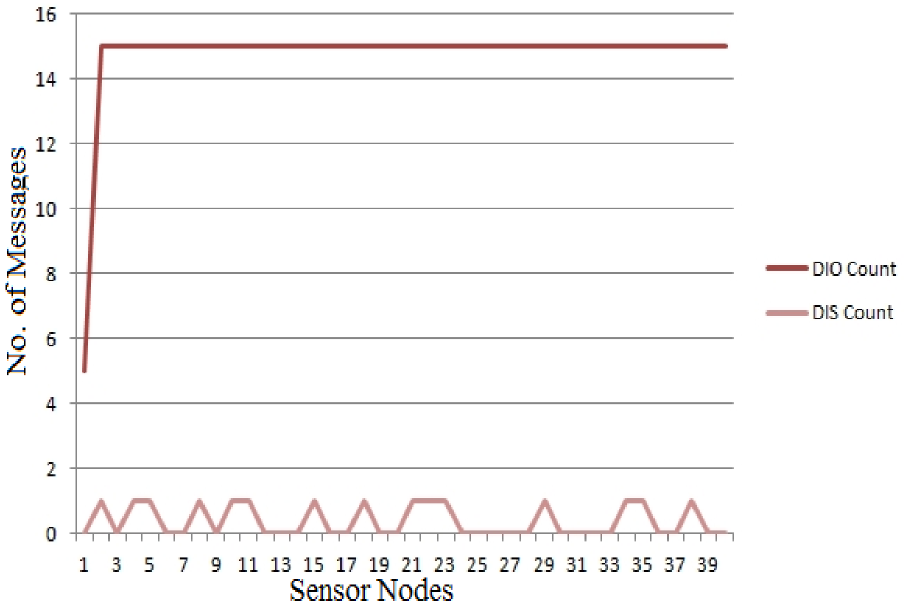


Fig 4.7 Number of DIO & DIS messages sent by 40 nodes

The figure.4.8 is obtained from the simulation results of 50 nodes. From the figure, except the server node, all the client nodes are sent equal number of DIO messages. In this simulation result, the server node sends minimum number of DIO messages. In the view of DIS messages, only a few nodes are sent DIS message.

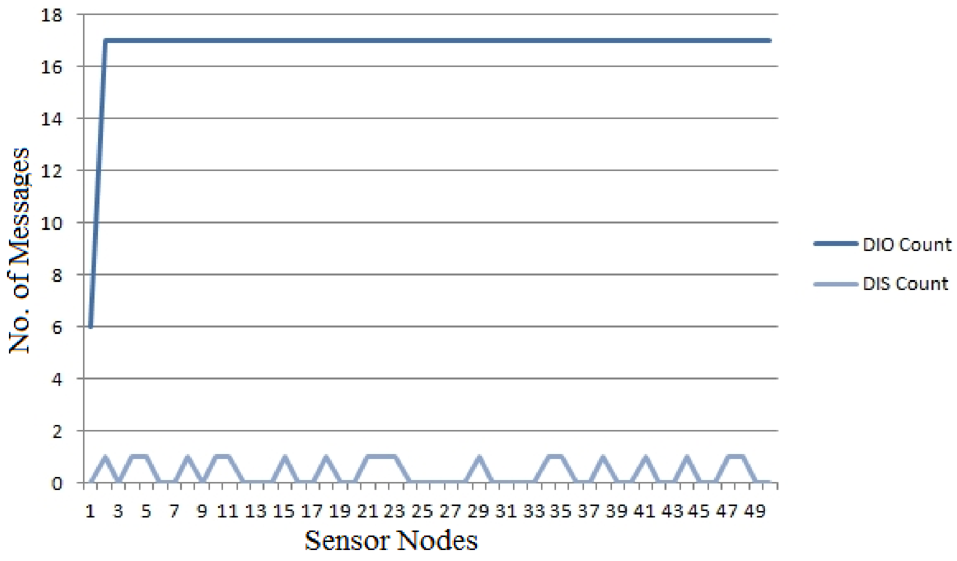
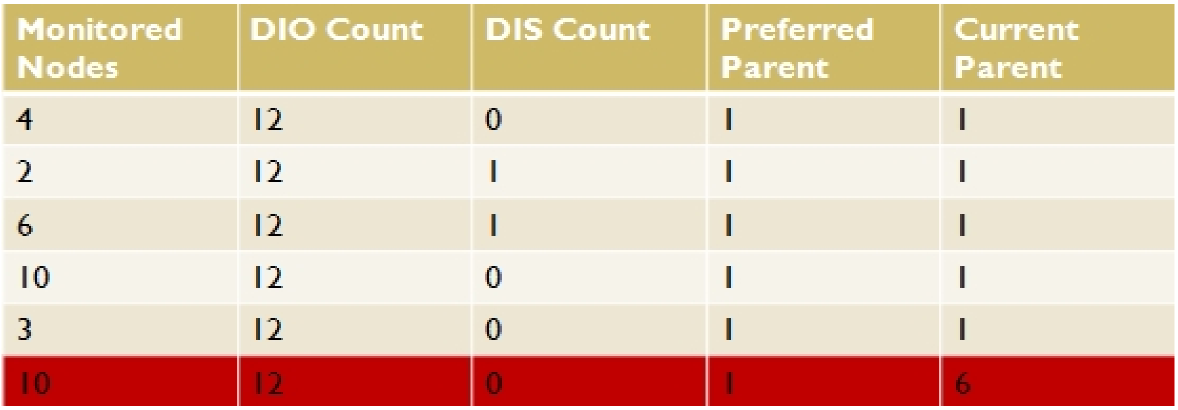


Fig 4.8 Number of DIO & DIS messages sent by 50 nodes

4.1.4 Parent Change Detection:

Since, ContikiRPL is a self organizing network the best route and parent are calculated and assigned in each mote. An intruder will look to find more information by changing the parents and consequently the routes. Frequent change in parents, routes and deviation from optimum parents and routes is considered as an abnormal behavior. When the network is setup, each node calculates and chooses its parents.



Monitored Nodes	DIO Count	DIS Count	Preferred Parent	Current Parent
4	12	0	1	1
2	12	1	1	1
6	12	1	1	1
10	12	0	1	1
3	12	0	1	1
10	12	0	1	6

The above output was observed from the log output. The RED marking denotes the parent change within the network.

4.1.5 Power Consumption:

The normal power consumption of a mote is shown in figure. 4.9. Implementation of real time monitoring will aid in detecting abnormalities in power consumption. From the figure. 4.9 the power consumed by the network before convergence is more. And after the network convergence, it consumes less power.

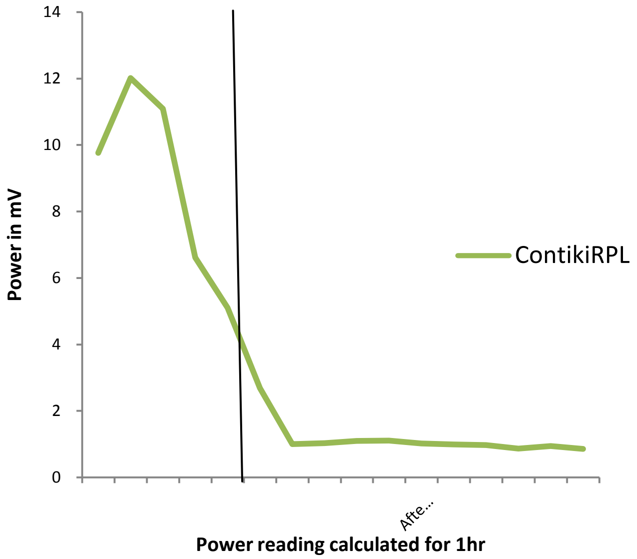


Fig 4.9 Power consumption of the sensor nodes

To convergence the network, the nodes will send rpl control messages. It will send DIS message to get the DIO information from the neighbour nodes. If the node gets the DIO message before sending a DIS message, then it will not send DIS message any more. For this rpl control message transmission, the nodes might consume more power. After the network convergence, there is no need to send these control messages frequently. So the power consumption level is reduced.

CHAPTER 5

GETTING STARTED WITH CONTIKI

5.1 Contiki-OS Structure:

The ContikiOS is written in basic C language. The fig. 5.1 shows the different folders that come with Contiki 2.7. The apps folder consists of application that can be run on Contiki. The core consists of source code for the IP stack. The CPU folder has source codes for processors that can run contiki. The platform consists of readily available motes that can be used to build Contiki. The Makefile.include compiles and builds the entire Contiki system.

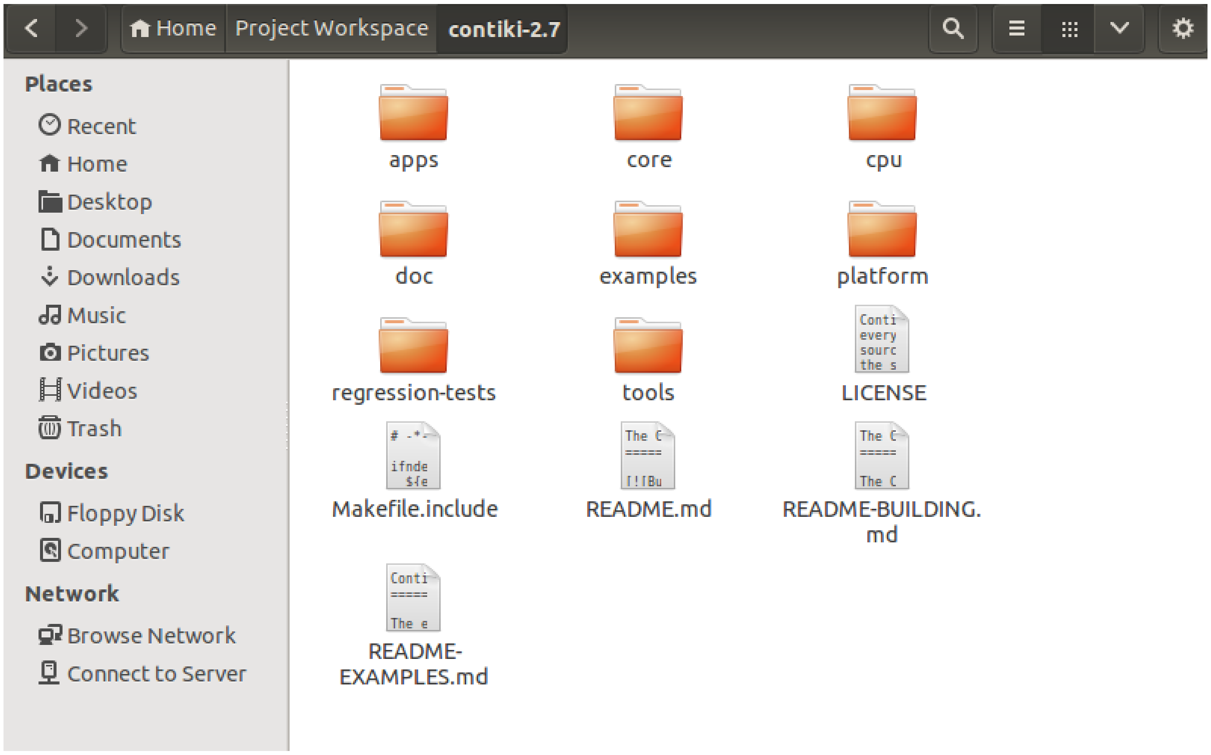


Fig 5.1 Different folders in Contiki 2.7

Inside this contiki folder open the core folder. This one will contain the files which are contributed in the network setup. In particular, the net folder will have the RPL based files. The folder which is having those files is named as rpl. The figure shows the screen shot of this particular folder.

The rpl folder contains the file named uip-icmp6.c. this file has the source code for sending and receiving the control messages. The various actions performed from this code are,

- DIO message sent/received
- DIS message sent/received
- DAO message sent/received
- DAO ACKNOLEDGEMENT sent/received

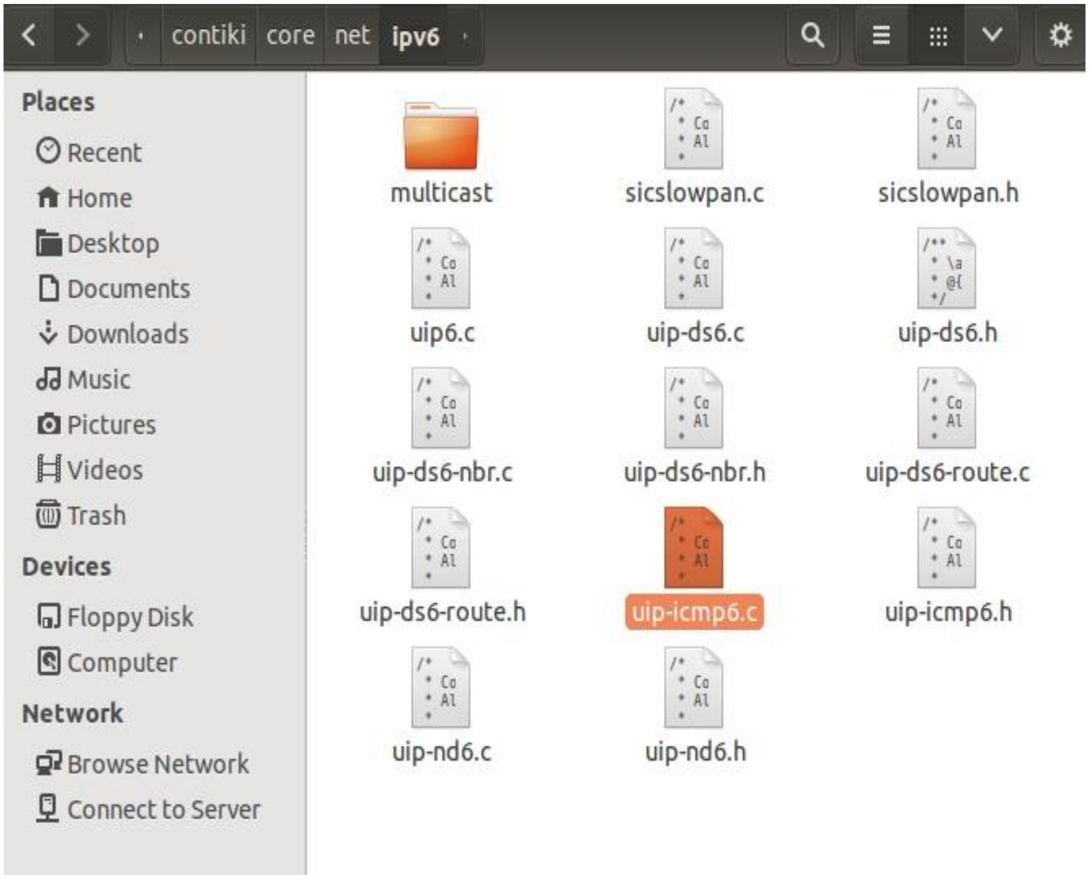
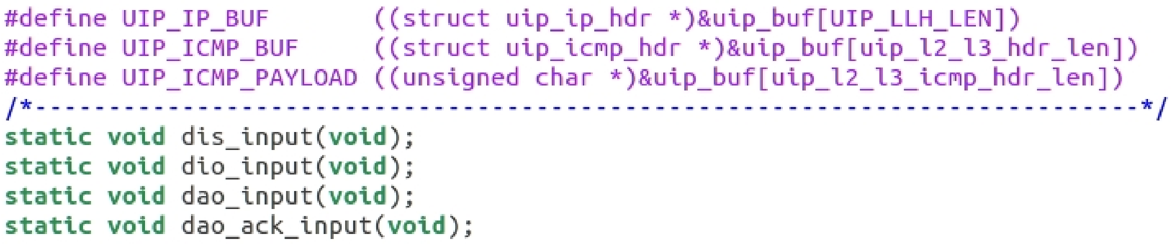


Fig 5.2 Folder path for uip-icmp6 source file

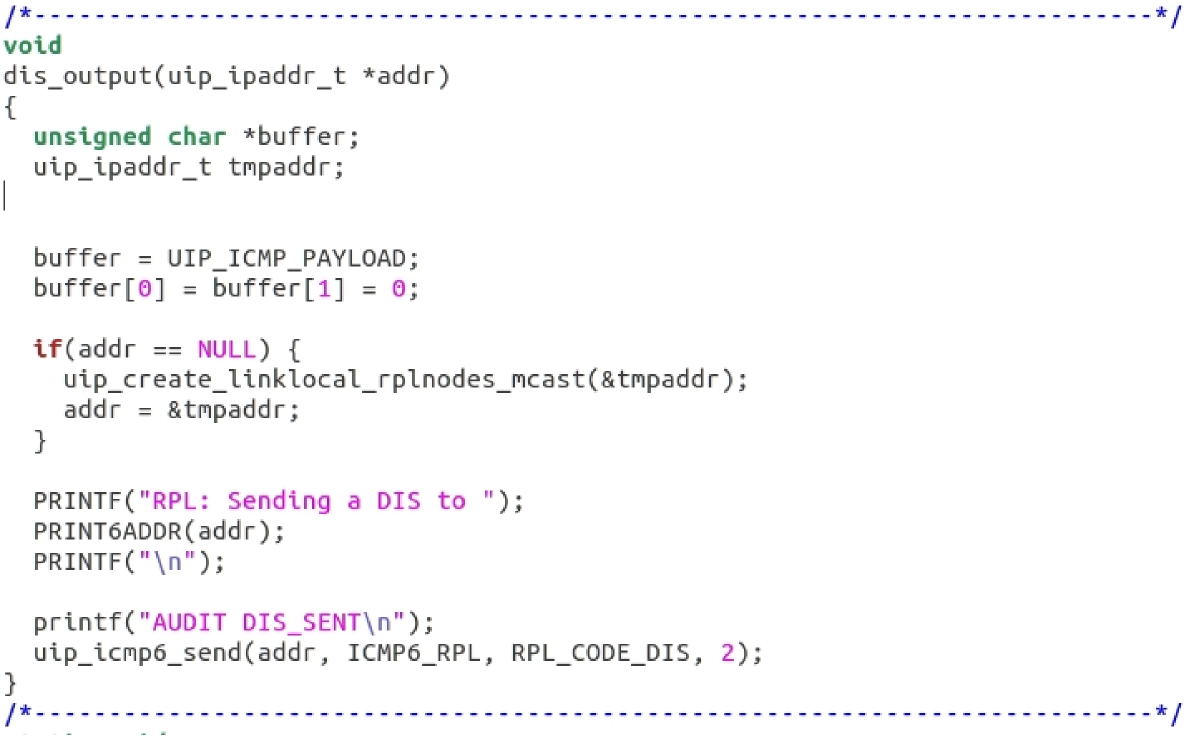
All these DIO, DIS and DAO control message functions are initialized in the script. This has been shown in the figure. The symbol ‘/*’ is used for comments.



```
#define UIP_IP_BUF       ((struct uip_ip_hdr *)&uip_buf[UIP_LLH_LEN])
#define UIP_ICMP_BUF     ((struct uip_icmp_hdr *)&uip_buf[uip_l2_l3_hdr_len])
#define UIP_ICMP_PAYLOAD ((unsigned char *)&uip_buf[uip_l2_l3_icmp_hdr_len])
/*---------------------------------------------------------------------------*/
static void dis_input(void);
static void dio_input(void);
static void dao_input(void);
static void dao_ack_input(void);
```

Fig 5.3 Initialization of control messages

The figure. 5.4 explains how the audit data records are being taken from the mote output. While the simulator runs, the required data for auditing the network is taken out. These records are having information like the time in milliseconds followed by the node number and then the action which is performed by that node.

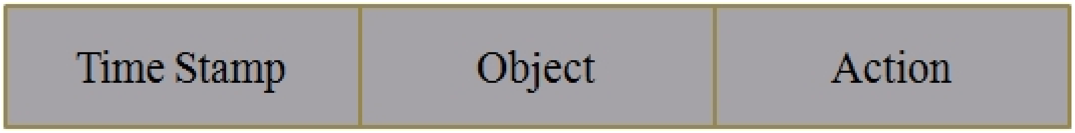


```
/*---------------------------------------------------------------------------*/
void
dis_output(uip_ipaddr_t *addr)
{
  unsigned char *buffer;
  uip_ipaddr_t tmpaddr;


  buffer = UIP_ICMP_PAYLOAD;
  buffer[0] = buffer[1] = 0;

  if(addr == NULL) {
    uip_create_linklocal_rplnodes_mcast(&tmpaddr);
    addr = &tmpaddr;
  }

  PRINTF("RPL: Sending a DIS to ");
  PRINT6ADDR(addr);
  PRINTF("\n");

  printf("AUDIT DIS_SENT\n");
  uip_icmp6_send(addr, ICMP6_RPL, RPL_CODE_DIS, 2);
}
/*---------------------------------------------------------------------------*/
```

Fig 5.4 Sample script for collecting audit records

5.2 Audit Record Structure

The audit records are very vital for designing intrusion detection system. The audit record structure is shown in the figure. 5.5. This audit record structure contains the time stamp in which the time is recorded in milliseconds. After that, the object responsible which is any of the connected node in the network and then the action performed by that node.



Time Stamp	Object	Action

Fig 5.5 Audit record structure

Figure. 5.6 shows the rpl-udp folder in contiki. This folder contains the source files for server nodes and client nodes. The files named .sky are representing the sky motes. While the simulator runs, the script from server.c and client.c will be executed. By running these the audit records could be taken out from the mote output.

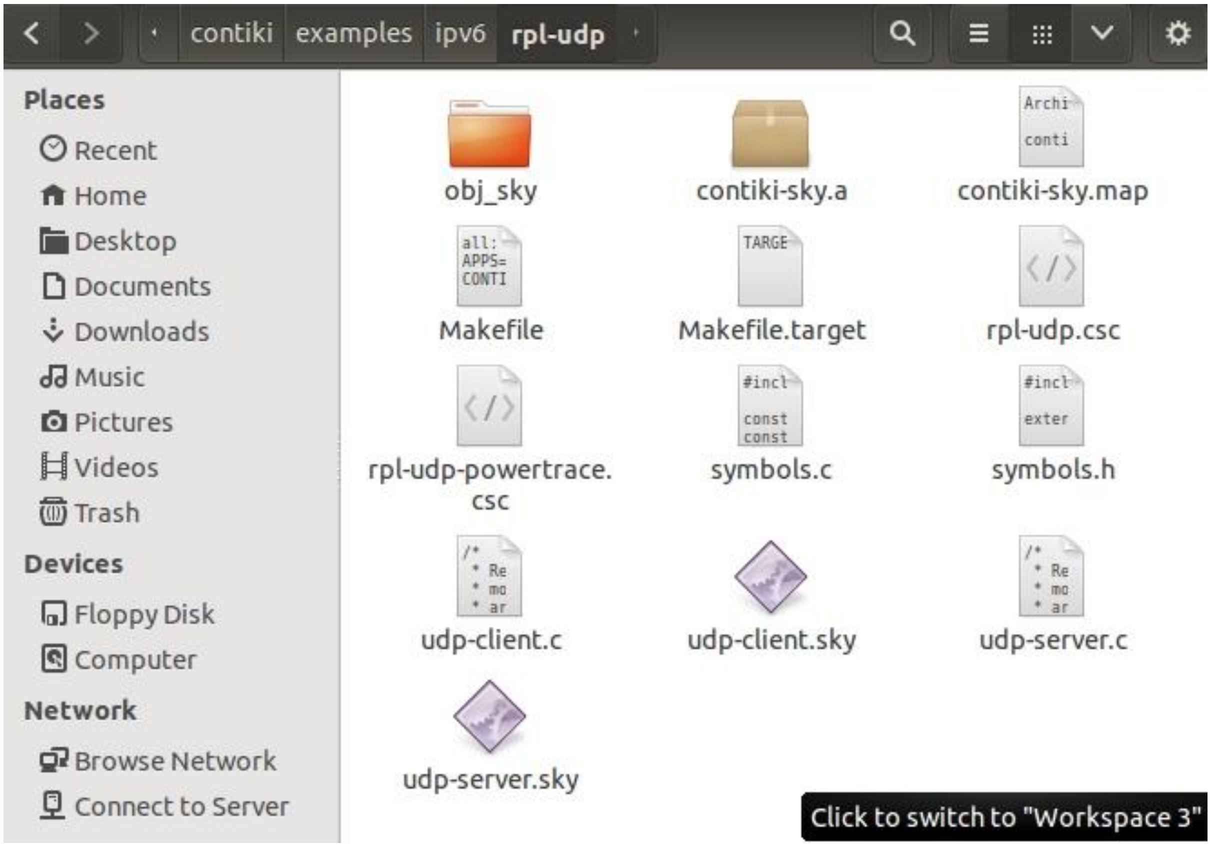
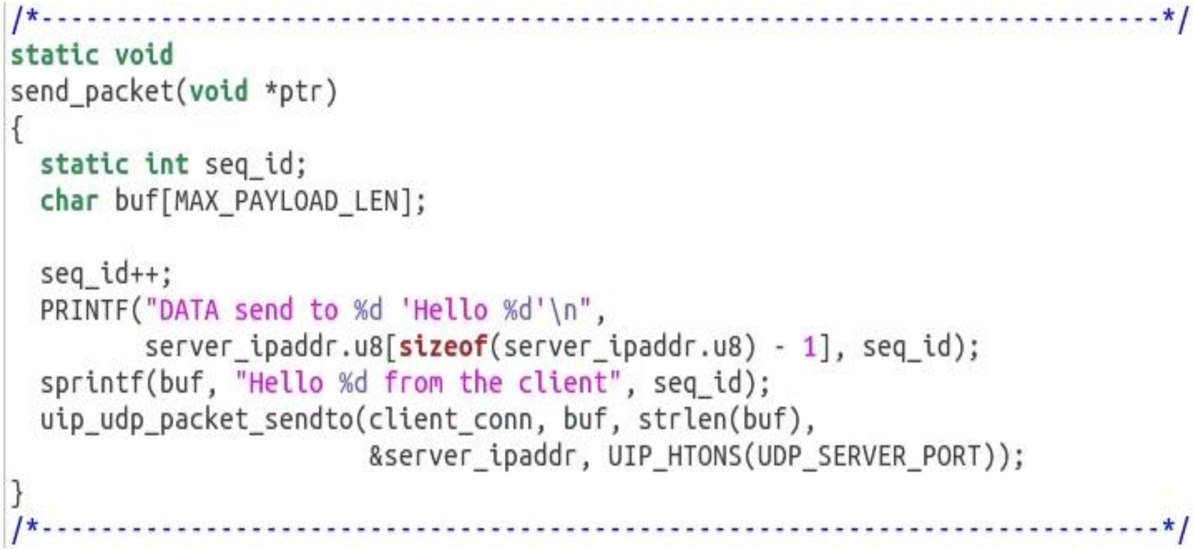


Fig 5.6 Folder for server and client source files

The data sent from a client/server node and data received from a client/server node can be collected using the script. This has been shown in the figure. 5.7.

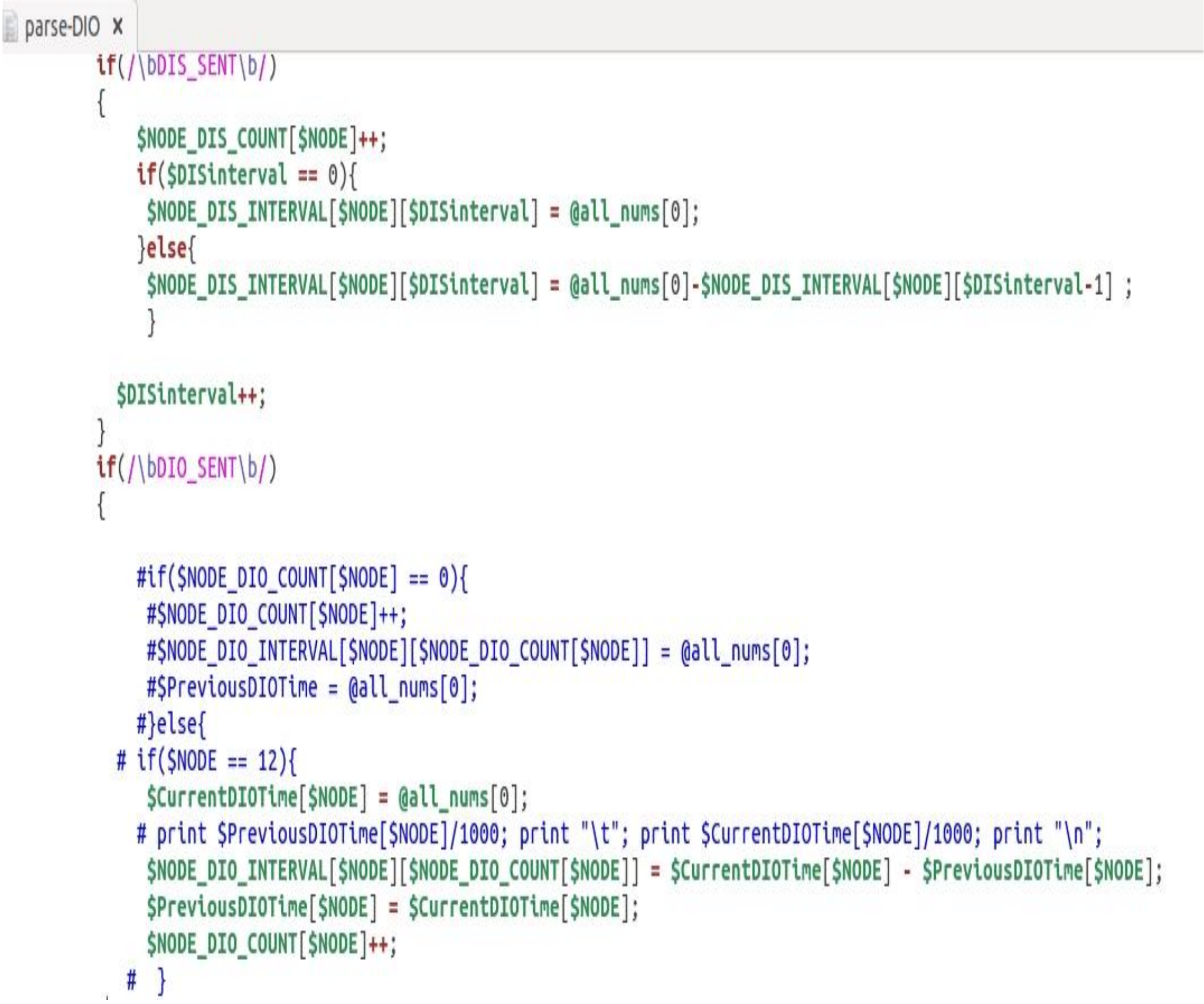


```
/*---------------------------------------------------------------------------*/
static void
send_packet(void *ptr)
{
  static int seq_id;
  char buf[MAX_PAYLOAD_LEN];

  seq_id++;
  PRINTF("DATA send to %d 'Hello %d'\n",
         server_ipaddr.u8[sizeof(server_ipaddr.u8) - 1], seq_id);
  sprintf(buf, "Hello %d from the client", seq_id);
  uip_udp_packet_sendto(client_conn, buf, strlen(buf),
                        &server_ipaddr, UIP_HTONS(UDP_SERVER_PORT));
}
/*---------------------------------------------------------------------------*/
```

Fig 5.7 Sample script for collecting audit data from client node

5.3 Perl Script Sample

Figure. 5.8 shows the parse script which is used to calculate the interval time between the two consecutive DIO, DIS and DAO control traffic messages. From this the total number of control messages sent and receive from a node can be calculated.

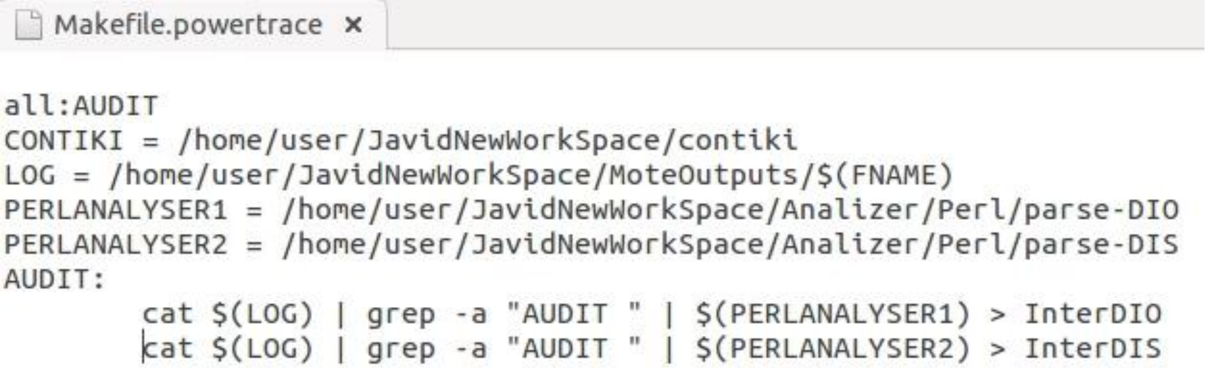


```
if(/\bDIS_SENT\b/)
{
    $NODE_DIS_COUNT[$NODE]++;
    if($DISinterval == 0){
     $NODE_DIS_INTERVAL[$NODE][$DISinterval] = @all_nums[0];
    }else{
     $NODE_DIS_INTERVAL[$NODE][$DISinterval] = @all_nums[0]-$NODE_DIS_INTERVAL[$NODE][$DISinterval-1] ;
     }

  $DISinterval++;
}
if(/\bDIO_SENT\b/)
{

    #if($NODE_DIO_COUNT[$NODE] == 0){
     #$NODE_DIO_COUNT[$NODE]++;
     #$NODE_DIO_INTERVAL[$NODE][$NODE_DIO_COUNT[$NODE]] = @all_nums[0];
     #$PreviousDIOTime = @all_nums[0];
    #}else{
  # if($NODE == 12){
     $CurrentDIOTime[$NODE] = @all_nums[0];
    # print $PreviousDIOTime[$NODE]/1000; print "\t"; print $CurrentDIOTime[$NODE]/1000; print "\n";
     $NODE_DIO_INTERVAL[$NODE][$NODE_DIO_COUNT[$NODE]] = $CurrentDIOTime[$NODE] - $PreviousDIOTime[$NODE];
     $PreviousDIOTime[$NODE] = $CurrentDIOTime[$NODE];
     $NODE_DIO_COUNT[$NODE]++;
   #  }
```

Fig 5.8 Script to calculate the interval time and message count

This parse script will collect the required audit data with time stamp. From this time stamp, the interval time between the audit data could be calculated in seconds. The distribution can be obtained for this interval time and the data density. The monitoring node will keep on analysing the monitored nodes audit record and compare with the original pattern. If any deviation found in the distribution from the pattern then that node will be suspected as anomaly node.

Figure. 5.9 shows the make file to run the perl script. By using this only the required data can be separated from the all collected output data.



```
all:AUDIT
CONTIKI = /home/user/JavidNewWorkSpace/contiki
LOG = /home/user/JavidNewWorkSpace/MoteOutputs/$(FNAME)
PERLANALYSER1 = /home/user/JavidNewWorkSpace/Analizer/Perl/parse-DIO
PERLANALYSER2 = /home/user/JavidNewWorkSpace/Analizer/Perl/parse-DIS
AUDIT:
	cat $(LOG) | grep -a "AUDIT " | $(PERLANALYSER1) > InterDIO
	cat $(LOG) | grep -a "AUDIT " | $(PERLANALYSER2) > InterDIS
```

Fig 5.9 Make file to run the perl script

The ‘grep’ command is used to pick up some specific output records. In the make file shown here will collect only the “AUDIT” data records from the mote output.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

The analyses done are for default ContikiRPL configuration. The network might have different Contiki implementation to improve convergence time to reduce overhead, etc., So, if the configuration changes, the analysis of the control messages will change accordingly. So, the design is in progress to implement a real time analysis and monitoring to facilitate flexibility in the configuration changes in ContikiRPL. The intruder will not only try to acquire the information but also try to distract the normal behavior of the network. The intruder may degrade the network performance by exhausting the node power or by frequently sending the control messages. The proposed method gives the best solution to detect the intruder node. So that it can be eliminated from the network easily and the network will not be vulnerable to such malicious activities.

6.2 FUTURE WORK

Since we try to implement IDS on 6LoWPAN network which is related to IOT, a web server can be launched in the monitoring mote to facilitate the user to see for himself the behavioral details of monitored motes from his browser. The network is connected to IPv6 through border router. So, the user anywhere in the world can be alerted if any malicious node or intruder node found.

REFERENCES

[1] “Towards intrusion detection in wireless sensor networks”., Ioannis, Krontiris, Tassos Dimitriou, and Felix C. Freiling. Proc. of the 13th European Wireless Conference. 2007.

[2] “Anomaly intrusion detection in wireless sensor networks”., Bhuse, Vijay, and Ajay Gupta. Journal of High Speed Networks 15.1 (2006): 33-51.

[3] “Intrusion detection systems for wireless sensor networks: A survey”., Farooqi, Ashfaq Hussain, and Farrukh Aslam Khan. Communication and networking. Springer Berlin Heidelberg, 2009. 234-241.

[4] “ANDES: an anomaly detection system for wireless sensor networks”., Gupta, Sumit, Rong Zheng, and Albert MK Cheng. Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE Internatonal Conference on. IEEE, 2007.

[5] “Abnormal node detection in wireless sensor network by pair based approach using IDS secure routing methodology”., Ahmed, Khandakar Rashed, et al. Int J Comput Sci Netw Sec 8.12 (2008): 339-342.

[6] “Trusting anomaly and intrusion claims for cooperative distributed intrusion detection schemes of wireless sensor networks”., Shaikh, Riaz Ahmed, et al. Young Computer Scientists, 2008. ICYCS 2008. The 9th International Conference for. IEEE, 2008.

[7] “Group-based intrusion detection system in wireless sensor networks”., Li, Guorui, Jingsha He, and Yingfang Fu. Computer Communications 31.18 (2008): 4324-4332.

[8] “A New Collaborative Approach for Intrusion Detection System on Wireless Sensor Networks”., de Sousa Lemos, Marcus Vinícius, Líliam Barroso Leal, and Raimir Holanda Filho. Novel Algorithms and Techniques in Telecommunications and Networking. Springer Netherlands, 2010. 239-244.

[9] “Lightweight Intrusion Detection for Wireless Sensor Networks”., Huh, Eui-Nam, and Tran Hong Hai. Intrusion Detection Systems, Pawel Skrobanek (Ed.), InTech (2011).

[10] “A Collaborative, Secure and Energy Efficient Intrusion Detection Method for Homogeneous WSN”., Mubarak, T. Mohamed, et al. Advances in Computing and Communications. Springer Berlin Heidelberg, 2011. 102-110.

[11] “A Bayesian Network in Intrusion Detection Systems”., M.Mehdi, S.Zair, et al. Journal of Computer Science 3 (5): 259-265, 2007 ISSN 1549-3636.

[12] “6LoWPAN: a study on QoS security threats and countermeasures using intrusion detection system approach”., Anhtuan Le, Jonathan Loo. INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS Int. J. Commun. Syst. (2012).

[13] “IDS Alert Classification Model Construction Using Decision Support Techniques”., Yan Zhang, Shuguang Huang, et al. 2012 International Conference on Computer Science and Electronics Engineering.

[14] “Network intrusion detection using rough sets based parallel genetic algorithm hybrid model”., Fen Zhou, Gaizhen Yang. 2010 International Symposium on Intelligence Information Processing and Trusted Computing.

[15] “A Clustering-Based Method for Intrusion Detection in Web Servers”., Hermano Pereira, Edgard Jamhour. Telecommunications (ICT), 2013 20th International Conference.

[16] “SPMOS-based Intrusion Detection Architecture”., Shi Qingsong, Chen Du, Nan Zhang, Jijun Ma, Tianzhou Chen. Fifth IEEE International Symposium on Embedded Computing.

[17] “A Heuristic Detection Network”., Lei Liu. Operator-Assisted (Wireless Mesh) Community Networks, 2006 1st Workshop.

[18] “A Distributed IDS for Ad Hoc Networks”., Paulo M. Mafra, Joni da Silva Fraga, Altair Olivo Santin. 2012 26th International Conference on Advanced Information Networking and Applications Workshops.