ISSN:0975-9646

A Novel Approach Based Wireless Intrusion Detection System

K.Suresh[1], A.Sarala Devi[2] , Jammi Ashok[3]

[1,3]Dept. of CSE,[2]Dept. of IT
Guru Nanak Institute of Technology ,Hyderabad

Abstract-Intrusion detection in Wireless Sensor Network (WSN) is of practical interest in many applications such as detecting an intruder in a battlefield. The intrusion detection is a mechanism for a WSN to detect the existence of inappropriate, incorrect, or anomalous moving attackers. In this paper, we consider this issue according to heterogeneous WSN models. Furthermore, we consider two sensing detection models: single-sensing detection and multiple-sensing detection. Our simulation results show the advantage of multiple sensor heterogeneous WSNs.

1. INTRODUCTION

A Wireless Sensor Network (WSN) is a collection of spatially deployed wireless sensors by which to monitor various changes of environmental conditions (e.g., forest fire, air pollutant concentration, and object moving) in a collaborative manner without relying on any underlying infra-structure support .Recently, a number of research efforts have been made to develop sensor hardware and network architectures in order to effectively deploy WSNs for a variety of applications. Due to a wide diversity of WSN application requirements, however, a general-purpose WSN design cannot fulfill the needs of all applications. Many network parameters such as sensing range, transmission range, and node density have to be carefully considered at the network design stage, according to specific applications. To achieve this, it is critical to capture the impacts of network parameters on network performance with respect to application specifications. Intrusion detection (i.e., object tracking) in a WSN can be regarded as a monitoring system for detecting the intruder that is invading the network domain.

1.1 PROBLEM DEFINITION

The intrusion detection application concerns how fast the intruder can be detected by the WSN. If sensors are deployed with a high density so that the union of all sensing ranges covers the entire network area, the intruder can be immediately detected once it approaches the network area. However, such a high-density deployment policy increases the network investment and may be even unaffordable for a large area. In fact, it is not necessary to deploy so many sensors to cover the entire WSN area in many applications, since a network with small and scattered void areas will also be able to detect a moving intruder within a certain intrusion distance. In this case, the application can specify a required intrusion distance within which the intruder should be detected.

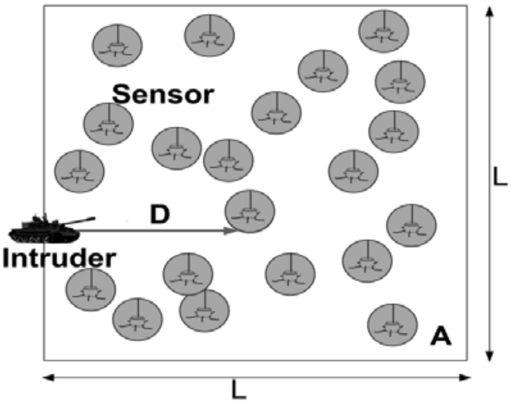


Fig 1: Intrusion detection in WSN

As shown in Fig. 1, the intrusion distance is referred as D and defined as the distance between the points the intruder enters the WSN, and the point the intruder is detected by the WSN system. This distance is of central interest to a WSN used for intrusion detection. In this paper, we derive the expected intrusion distance and evaluate the detection probability in different application scenarios. For example, given an expected detection distance EðDÞ, we can derive the node density with respect to sensors' sensing range, thereby knowing the total number of sensors required for WSN deployment.

In a WSN, there are two ways to detect an object (i.e., an intruder): single-sensing detection and multiple-sensing detection. In the single-sensing detection, the intruder can be successfully detected by a single sensor. On the contrary, in the multiple-sensing detection, the intruder can only be detected by multiple collaborating sensors .In some applications, the sensed information provided by a single sensor might be inadequate for recognizing the intruder. It is because individual sensors can only sense a portion of the intruder. For example, the location of an intruder can only be determined from at least three sensors' sensing.

In view of this, we analyze the intrusion detection problem under two application scenarios: single-sensing detection and multiple-sensing detection. According to the capability of sensors, we consider two network types: homogeneous and heterogeneous WSNs. We define the sensor capability in terms of the sensing range and the transmission range. In a heterogeneousWSN some sensors have a larger sensing range and more power to achieve a longer transmission range. In this paper, we show that the heterogeneous WSN increases the detection probability for a given intrusion detection distance. This motivates us to analyze the network connectivity in this paper.

2. LITERATURE SURVEY

An **Intrusion detection system** (**IDS**) is software and/or hardware designed to detect unwanted attempts at accessing, manipulating, and/or disabling of computer mainly through a network, such as the Internet. These attempts may take the form of attacks, for example, by crackers and/or disgruntled employees. IDS cannot directly detect attacks within properly encrypted traffic.

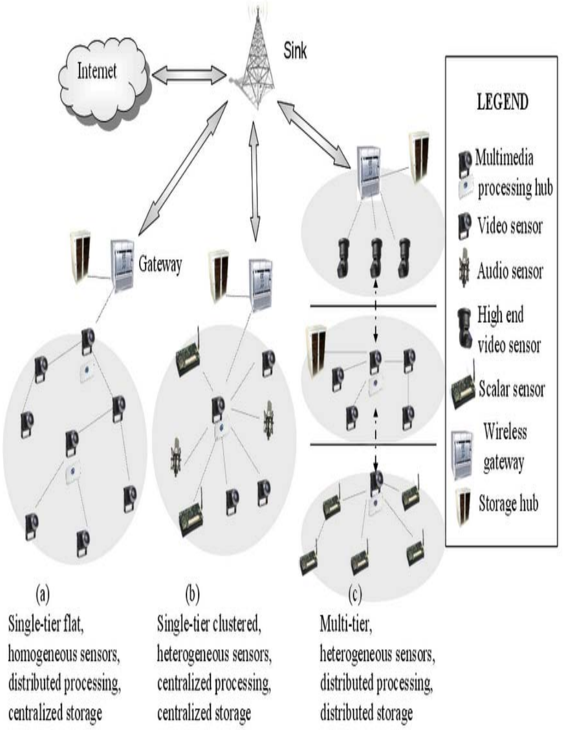
An intrusion detection system is used to detect several types of malicious behaviors that can compromise the security and trust of a computer system. This includes network attacks against vulnerable services, data driven attacks on applications, host based attacks such as privilege escalation, unauthorized logins and access to sensitive files, and viruses IDS can be composed of several components: **Sensors** which generate security events, **a** Console to monitor events and alerts and control the sensors, and a central Engine that records the events logged by the sensors in a database and uses a system of rules to generate alerts from security events received. There are several ways to categorize IDS depending on the type and location of the sensors and the methodology used by the engine to generate alerts. In many simple IDS implementations all three components are combined in a single device or appliance.

2.1 WIRELESS SENSOR NETWORK

A wireless sensor network (WSN) is a wireless network consisting of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants, at different locations. The development of wireless sensor networks was originally motivated by military applications such as battlefield surveillance. However, wireless sensor networks are now used in many civilian application areas, including environment and habitat monitoring, healthcare applications, home automation, and traffic control.

In addition to one or more sensors, each node in a sensor network is typically equipped with a radio transceiver or other wireless communications device, a small microcontroller, and an energy source, usually a battery. The envisaged size of a single sensor node can vary from shoebox-sized nodes down to devices the size of grain of dust although functioning 'motes' of genuine microscopic dimensions have yet to be created. The cost of sensor nodes is similarly variable, ranging from hundreds of dollars to a few cents, depending on the size of the sensor network and the complexity required of individual sensor nodes. Size and cost constraints on sensor nodes result in corresponding constraints on resources such as energy, memory, computational speed and bandwidth.

A sensor network normally constitutes a wireless ad-hoc network, meaning that each sensor supports a multi-hop routing algorithm (several nodes may forward data packets to the base station).



(a) Single-tier flat, homogeneous sensors, distributed processing, centralized storage

(b) Single-tier clustered, heterogeneous sensors, centralized processing, centralized storage

(c) Multi-tier, heterogeneous sensors, distributed processing, distributed storage

Fig 2: Architecture of WSN

3. TESTING AND IMPLEMENTATION

In the testing part, we perform various tests in order to check errors occurred in the project and to rectify those errors and give efficient outputs. Validation is nothing but is the user name and password can be authenticated with all the strings, alphabets and special characters. it need not be like all should be given, it’s up to users interest.

3.1 TESTS PERFORMED IN WIRELESS INTRUSION DETECTION SYSTEM

- **String testing**

All inputs to the applications are in the form of string. The strings were tested for nulls; it was tested for length as well as in data type’s conversions. Exceptions were handled to check if the above validations were performed and errors handled.

- **Unit testing**

Every module was individually tested where each command output was checked to receive appropriate inputs and if it generated appropriate outputs. Every command was individually checked to output correct data. The alignment of the data, scrolling, visibility of text output was all checked for appropriateness.

- **Integrated Testing**

All the modules

1. Environment and user interface module.
2. File system maintenance module.
3. Network management module.
4. Command-line surfing module.
5. Command-line editing and file handling module

Were combined and then executed as a single unit. It was found to be executing in a synchronized manner. The user interface module interacted with the commands module to provide the input, to the command module and receive the output back on the interface module. Similarly, handshaking is visible between other modules.

- **Top down Testing**

The control of flow in the application was tested with that of design phase. The application was top down tested to check if the sequence windows/frames that open every time an event was executed was in sequence or not. The results obtained were found to be that of design phase.

- **Black box Testing**

In the module that handled networking commands sockets are created and queried upon based on the commands .The inputs to the sockets, the expected outputs visible and known. The functional logic that was part of the data being retrieved swells any error resulting were not known as it was a result of incoming data from a remote machine. The user interface module was black box tested to check the control flow .In the other modules in file system related commands this testing was done to check if the output returned were appropriate.

- **White box Testing**

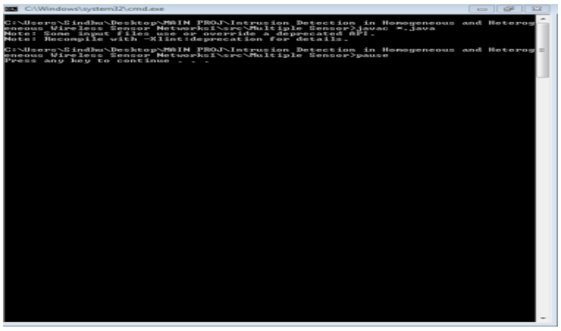
As the application contains code that was mostly in the form of loops and conditions checking every statement was necessary most often only positive conditions are checked using few inputs .This resulted in bypassing all, most of statement which were never being checked.

In the five modules

1 Constructing Sensor Network
2 Packet Creation
3 Find authorized and un authorized port
4 Constructing Inter-Domain Packet Filters
5 Receiving the valid packet

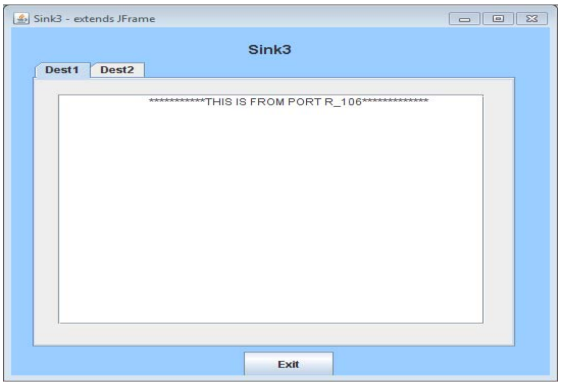
The White box testing is done to remove the unexpected errors.

4. RESULTS



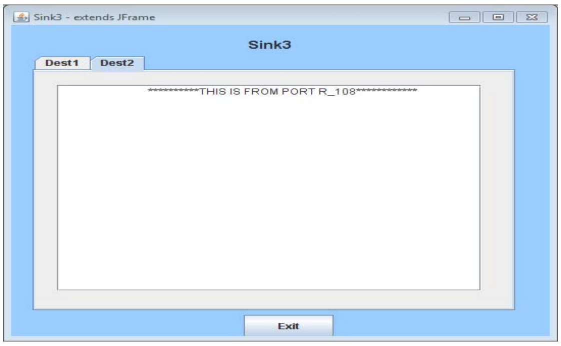
```
C:\Users\Sindhu\Desktop\MAIN PROJ\Intrusion Detection in Homogeneous and Heterog
eneous Wireless Sensor Networks1\src\Multiple Sensor>javac *.java
Note: Some input files use or override a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

C:\Users\Sindhu\Desktop\MAIN PROJ\Intrusion Detection in Homogeneous and Heterog
eneous Wireless Sensor Networks1\src\Multiple Sensor>pause
Press any key to continue . . .
```

Fig 3. Compilation Screen



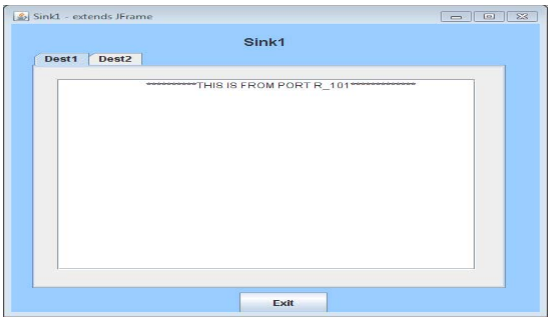
```
***********THIS IS FROM PORT R_106*************
```

Fig 4. One Destination Port of Receiver/Sink3



```
**********THIS IS FROM PORT R_108************
```

Fig 5. Other Destination Port of Receiver/Sink3



```
**********THIS IS FROM PORT R_101*************
```

Fig 6. Other Destination of Receiver/Sink1

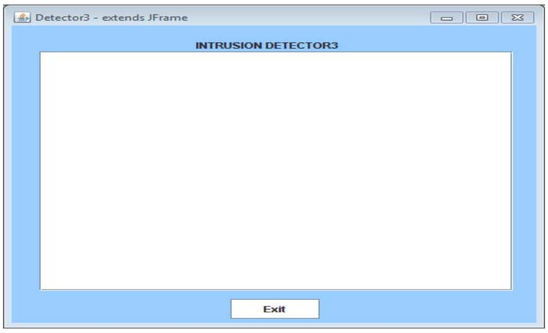
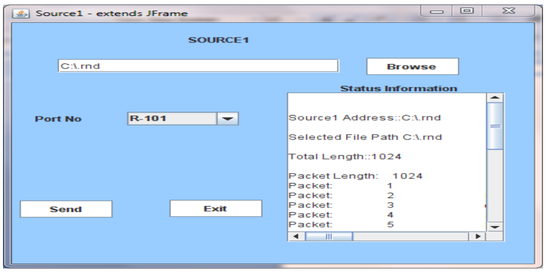


Fig 7. Detector 1



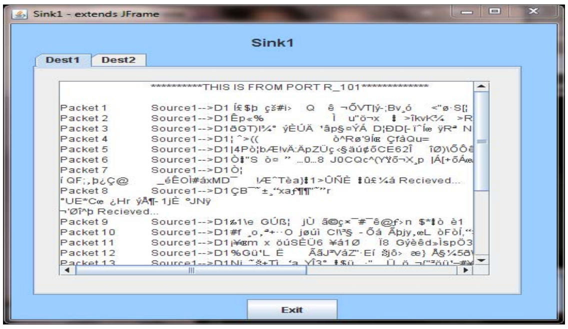
C:\.rnd

Port No R-101

Status Information

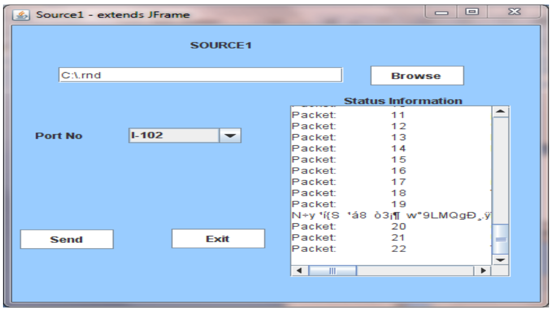
```
Source1 Address::C:\.rnd

Selected File Path C:\.rnd

Total Length::1024

Packet Length:   1024
Packet:          1
Packet:          2
Packet:          3
Packet:          4
Packet:          5
```

Fig 8: Sending File to a Receiver and its details in Status Information



```
          **********THIS IS FROM PORT R_101*************

Packet 1      Source1-->D1 Í£$þ ç፮#i›   Q   ê ¬ÕVT|ý-;Bv¸ó    <"ø·S[¦
Packet 2      Source1-->D1Êp«%            Ì   u"ö¬x  ‡  >îkvK³¼   >R
Packet 3      Source1-->D1ðGT)Ì¾°  ýÈÚÄ  ¹âp§¤ÝÁ  D¦ÐD[-ï¨Íœ  ÿRª  N
Packet 4      Source1-->D1¦ˆ>((                ò^Rø'9Íæ  ÇfâQu=
Packet 5      Source1-->D1|4Pò¦bÆ!vÄ:ÄpZÜç‹§äú¢õCE62Î    ÎØ)\ÕÔê
Packet 6      Source1-->D1Ò‡"S  ò¤  ”  …0…8  J0CQc^(YŸõ¬X¸p  |Á[+õÁæ
Packet 7      Source1-->D1Ò¦
í QF;,þ¿Ç@      _éÊOÌ#áxMD¯    !Æˆ Tèa}‡1>ÛÑÈ  ‡û£¼á Recieved...
Packet 8      Source1-->D1ÇB¯˜±¸“xaƒ¶¶'˜”r
ªUE*Cœ  ¿Hr  ýÅ¶- 1jÈ  ºJNÿ
¬'Øî^þ Recieved...
Packet 9      Source1-->D1‰1\e  GÚß¦   jÙ  ã©ç×¯#¯ê@ƒ›n  $*‡ò  è1
Packet 10     Source1-->D1#f ¸o,ª+··O  jøúì  C!\³§ - Õá  Ãþjy,œL  òFòÍ,“
Packet 11     Source1-->D1¡¥Æm  x  öúSÈÜ6  ¥á1Ø    Ï8  Gýèêd»ÌspÖ3
Packet 12     Source1-->D1%Gü'L  Ë       ÃãJªVáZ"·Eí  Šjô›  æ}  Å§¼5ð
Packet 13     Source1-->D1Ni ˜Š+Ti  ‘a  YÎ3°  ‡$ü   ·"    Ü ö ¬£"³öü'—#¥
```

Fig 9 : File Received by Sink 1



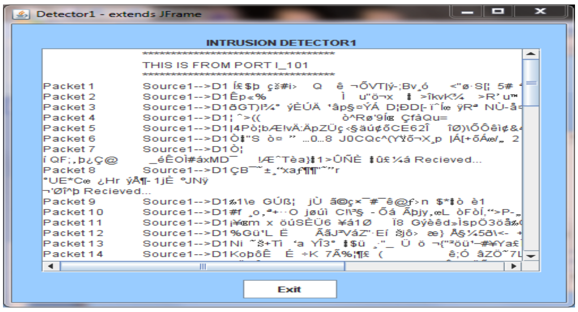
C:\.rnd

I-102

Status Information

```
Packet:         11
Packet:         12
Packet:         13
Packet:         14
Packet:         15
Packet:         16
Packet:         17
Packet:         18
Packet:         19
N÷y ¹í{S  ¹á8  ò3¡¶  w°9LMQgÐ¸.ÿ
Packet:         20
Packet:         21
Packet:         22
```

Fig 10: Data from an Intruder Port and Its Details in Status Information



```
                 **********************************
                 THIS IS FROM PORT I_101
                 **********************************
Packet 1         Source1-->D1 Í£$þ ç§#i›  Q  ê ¬ÕVT|ý-;Bv¸ó   <"ø·S[¦ 5#
Packet 2         Source1-->D1Êp«%          Ì  u"ö¬x  ‡ >îkvK¾  >R’u™
Packet 3         Source1-->D1ðGT)I¾° ýÈÚÄ ¹âp§¤ÝÁ D¦ÐD[-ï˜Íœ ÿR³ NÙ-å
Packet 4         Source1-->D1¦ ˆ>((            ò^Rø'9Íæ ÇfàQu=
Packet 5         Source1-->D1|4Pò¦bÆ!vÄ:ÄpZÜç‹§äú¢õCE62Î   îØ)\ÕÔêì¢&
Packet 6         Source1-->D1Ò‡"S ò¤ ” …0…8 J0CQc^(YŸõ¬X¸p |Á[+õÁœ/„ 2
Packet 7         Source1-->D1Ò¦
í QF;,þ¿Ç@       _éÊOÌ#áxMD¯   !Æˆ Tèa}‡1>ÛÑÈ ‡û£¼á Recieved...
Packet 8         Source1-->D1ÇB¯˜±¸“xaƒ¶¶"˜”r
°UE*Cœ ¿Hr ýÅ¶-1jÈ °JNÿ
¬'Øî^þ Recieved...
Packet 9         Source1-->D1‰1\e GÚß¦  jÙ ã©ç×¯#¯ê@ƒ›n $*‡ò è1
Packet 10        Source1-->D1#f ¸o,ª+··O jøúì C!\³§ - Õá Ãþjy,œL òFòÍ,“>P-„
Packet 11        Source1-->D1¡¥æm x öúSÈÜ6 ¥á1Ø   Ï8 Gýèêd»ÌspÖ3öå‰
Packet 12        Source1-->D1%Gü¹L Ë    ÃãJªVáZ"·Eí Šjô› æ} Å§¼5ð\<-  +
Packet 13        Source1-->D1Ni ˜Š+Tì ‘a YÎ3° ‡$ü ¸·"_ Ü ö ¬{"²öü¹-#¥Ya£
Packet 14        Source1-->D1KoþôÊ  É ÷K 7Ã%¦¶£ (       ê;Ó âZÖ˜7L
```

Fig 11: Intruder Detected by Detector 1

5. CONCLUSION

Intrusion has become a very common problem these days, so this paper analyzed this problem by characterizing the intrusion detection probability with respect to the intrusion distance and the network parameters (i.e., node density, sensing range, and transmission range).

The analytical model for intrusion detection allows us to analytically formulate intrusion detection probability within a certain intrusion distance under various application scenarios. Also it is capable of detecting jamming attacks.

In this paper, we achieved two goals, they are: we detected more attacks and forced the operator to give a decent service. We allow cheaters to come into play, but their impact is self-limiting as a working network is needed for them to play.

Our simulation results verify the correctness of the proposed analytical model. This work provides insights in designing homogeneous and heterogeneous WSNs and helps in selecting critical network parameters so as to meet the application requirements.

Our future work includes the intrusion detection in Internet application and in Parallel computer interconnection network.

REFERENCES

[1] R. Hemenway, R. Grzybowski, C. Minkenberg, and R. Luijten, “Optical-packet-switched interconnect for supercomputer applications,”*OSA J. Opt. Netw.*, vol. 3, no. 12, pp. 900–913, Dec. 2004.

[2] C. Minkenberg, F. Abel, P. Müller, R. Krishnamurthy, M. Gusat, P.Dill, I. Iliadis, R. Luijten, B. R. Hemenway, R. Grzybowski, and E.Schiattarella, “Designing a crossbar scheduler for HPC applications,”*IEEE Micro*, vol. 26, no. 3, pp. 58–71, May/Jun. 2006.

[3] E. Oki, R. Rojas-Cessa, and H. Chao, “A pipeline-based approach formaximal-sized matching scheduling in input-buffered switches,” *IEEE Commun.Lett.*, vol. 5, no. 6, pp. 263–265, Jun. 2001.

[4] C. Minkenberg, I. Iliadis, and F. Abel, “Low-latency pipelined crossbar arbitration,” in *Proc. IEEE GLOBECOM 2004*, Dallas, TX, Dec. 2004, vol. 2, pp. 1174–1179.

[5] C. Minkenberg, R. Luijten, F. Abel, W. Denzel, and M. Gusat, “Current issues in packet switch design,” *ACM Comput.Commun. Rev.*, vol. 33, no. 1, pp. 119–124, Jan. 2003.

[6] C. Minkenberg, F. Abel, P. Müller, R. Krishnamurthy, and M. Gusat,“Control path implementation of a low-latency optical HPC switch,” in*Proc. Hot Interconnects 13*, Stanford, CA, Aug. 2005, pp. 29–35.

[7] C.-S. Chang, D.-S.Lee, and Y.-S.Jou, “Load-balanced Birkhoff-von Neumann switches, part I: One-stage buffering,” *Elsevier Comput.Commun.*, vol. 25, pp. 611–622, 2002.

[8] A. Tanenbaum, *Computer Networks*, 3rd ed. Englewood Cliffs, NJ: Prentice Hall, 1996.

[9] R. Krishnamurthy and P. Müller, “An input queuing implementation for low-latency speculative optical switches,” in *Proc. 2007 Int. Conf.Parallel Processing Techniques and Applications (PDPTA’07)*, Las Vegas, NV, Jun. 2007, vol. 1, pp. 161–167.

[10] H. Takagi, *Queueing Analysis, Volume 3: Discrete-Time Systems*. Amsterdam: North-Holland, 1993.

[11] V.P. Mhatre, C. Rosenberg, D. Kofman, R. Mazumdar, and N. Shroff, “A Minimum Cost Heterogeneous Sensor Network with a Lifetime Constraint,” IEEE Trans. Mobile Computing,vol. 4, no. 1, pp. 4-15, 2005.

[12] M. Yarvis, N. Kushalnagar, H. Singh, A. Rangarajan, Y. Liu, and S. Singh, “Exploiting Heterogeneity in Sensor Networks,” Proc. IEEE INFOCOM, 2005.

[13] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A Survey on Sensor Networks,” IEEE Comm. Magazine, vol. 40, no. 8, pp. 102-114, Aug. 2002.

[14] H. Kung and D. Vlah, “Efficient Location Tracking Using Sensor Networks,” Proc. IEEE Wireless Comm. and Networking Conf., vol. 3, pp. 1954-1961, Mar. 2003.

AUTHORS BIOGRAPHY

Mr. K. Suresh[1]

Kanchi. Suresh, B.Tech(CSE), M.Tech(SE), is currently working as an Associate Professor in Computer Science and Engineering Department at Guru Nanak Institute of Technology Hyderabad, A.P, India. He has completed his B.Tech. (CSE) from VR. Siddhartha Engineering College, Vijayawada – Acharya Nagarjuna University Guntur, A.P. He has completed his M.Tech. (SE) from School of Information Technology-JNTUH Kukatpally. His main interest is around Compilers, Formal Languages and Automata, Computer Networks, Linux Programming and Data Mining. He has attended National and International Conferences.

A.Sarala Devi[2]

Currently working as Assoc.Professor in the Department of Information Technology at Gurunanak Institute of Technology, Hyderabad, A.P, INDIA. She has received her MCA from Kakatiyaa University and M.Tech. with specialization in Computer Science and Engineering from Acharya Nagarjuna University , A..P INDIA.

Her main research interest includes Data Warehousing and Data Mining, Network Security and Computer networks. She has been involved in the organization of conferences and workshops.

Jammi Ashok[3]

Currently working as Professor and Head at Guru Nanak Institute of Technology, Hyderabad, A.P, INDIA. He has received his B.E. Degree from Electronics and Communication Engineering from Osmania University and M.E. with specialization in Computer Technology from SRTMU, Nanded, INDIA

His main research interest includes neural networks, Bioinformatics and Artificial Intelligence. He has been involved in the organization of a number of conferences and workshops. He has been published more than 35 papers in International journals and conferences. He is currently doing his Ph.D from Anna University and is at the end of submission.