TRUSTWORTHY IN THE DYNAMIC IOT CLOUD

S.Nandhini devi[1], F.A.MONICA SELES [2], S.SHARMELA [3]
MAMSCHOOL OF ENGINEERING, ANNA UNIVERSITY
[1]devinandhini1982@gmail.com
[2] Monicaseles2098@gmail.com
[3] Sharmelasagayaraj17@gmail.com

Abstract— In this paper, we consider the issues of trustworthy computing for the dynamic IoT cloud. First, we introduce the vertical and horizontal computing structures in the extended IoT cloud where IoT devices, edge, fog, and cloud are integrated in a layered infrastructure. Then, we consider the issues and design a framework and accompanying mechanisms for performing trustworthy computing making use of the vertical IoT cloud structure to secure the IoT cloud in vertical and horizontal computation structures. Specifically, we discuss a general trustworthy computing pattern in the IoT cloud and use intrusion detection as an example to illustrate the idea, develop an advanced access control and policy definition model for highly dynamic IoT networks, and introduce an integrated data provenance and information control mechanism to assure the data integrity and secure the information flow for various computation patterns in the IoT cloud.

Keywords—edge computing ,fog computing, role based access control, attribute based access control, resource hierarchy, relative role model, data provenance, information flow control, IoT cloud infrastructure.

I. INTRODUCTION

. Wireless sensor networks (WSNs) comprise of a large number of small sensing and self- powered sensor nodes distributed in a geographical region. The sensor nodes gather communicate in a wireless fashion. Sensing, processing information or detect special events and node is said to be faulty if it is not functioning communication are three key tasks whose combination in one tiny device gives rise to a vast number of remote sensing applications. Although WSNs provide endless opportunities, at the same time pose formidable challenges. Some of these challenges are low battery, less computational capabilities and inefficient use of communication resources. Among these impediments, the most difficult one is the mysterious data sent by an unknown faulty sensor node either to the fusion centre (FC) such as base station (BS) or to the neighbouring sensor node . In WSNs, the accuracy of the observed data sent by a sensor node is important for the overall network's performance. Therefore, detection of faulty sensor nodes is an essential issue in WSNs .A sensor properly . In the literature, the faults in WSNs are broadly classified into two types known as hard fault (permanent or static fault) and soft fault (or dynamic fault) . The hard fault occurs if a sensor node fails to communicate with the rest of the sensor nodes in the network . When the sensor node is able to communicate with the other sensor nodes, but transmits erroneous message, then such type of fault is known as soft fault.

2. Existing System

Open Flow provides an open protocol to program the flow table in different switches and routers. A network administrator can partition traffic into production and research flows of fault detection. Researchers can control their own flows - by choosing the routes their packets follow and the processing they receive. In this way, researchers can try new routing protocols, security models, addressing schemes, and even alternatives to IP. On the same network, the production traffic is isolated and processed in the same way as today. The data path of an Open Flow Switch consists of a Flow Table, and an action associated with each flow entry. The set of actions supported by an Open Flow Switch is extensible, but below we describe a minimum requirement for all switches. For high-performance and low-cost the data path must have a carefully prescribed degree of flexibility. This means forgoing the ability to specify arbitrary handling of each packet and seeking a more limited, but still useful, range of actions.

2.1 Disadvantages

- It needs to be aware of the specific technique supported in order to issue the corresponding commands.
- It is handled within each technology using its own mechanisms.

3. Proposed System

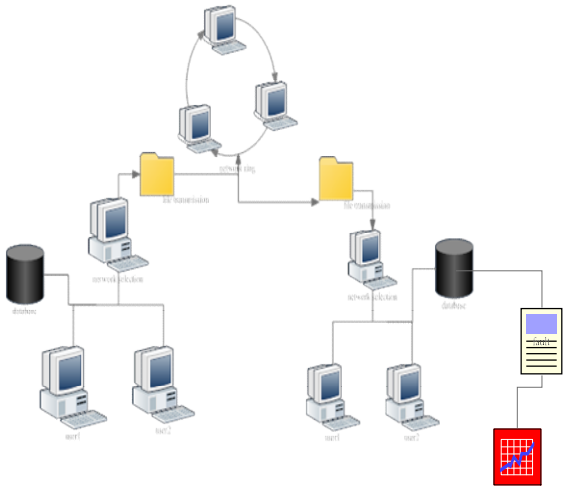
In the proposed work, to minimize the computational complexity and improve the accuracy, each sensor node initially tests the presence of faulty sensor nodes in its neighbour, if found, then predicts the probable fault status of them. For this, the Neyman–Pearson (NP) detection method is used. The sensor nodes shared the predicted probable fault status of the neighbours with them. Then, each sensor node uses a fusion scheme to take the final decision on its fault status. The major contributions of this paper are (i) design and evaluation of an efficient distributed fault diagnosis algorithm for detecting soft faulty sensor nodes in large WSNs, (ii) the Neyman–Pearson (NP) detection method is used to detect the faulty sensor node (iii) the performance is compared with the existing distributed algorithms such as JSA and Jiang , and(iv) the algorithms are implemented in NS3 .The remaining part of the paper is organized as follows. In the related work which provides an exhaustive view about the previous work is discussed. The network model used for the development of the distributed algorithm is provided. The proposed distributed fault detection (DFD) algorithm is described .The analytical model which proves the correctness of the algorithm is also given . The many IOT devices are not powerful enough to perform traffic pattern analysis for intrusion detection also access control and privacy validation conventionally being carried out by centralized certification authority could be too computation intensive for some simple IOT computing infrastructure to perform trustworthy computing for IOT devices.

3.1 Advantages

* It allows different types of communication that have traditionally been handled separately to be integrated at a single decision point.

* It supports the implementation of a variety of mobility protocols.

Figure 2.1 System architecture



In the system architecture the source and the destination can be calculated using the TE algorithm to analysis the path to the source to destination and the different path and the forwarding node with different bandwidth can be selected. Then the frequency can be analysed to transmit the data. Then the data can be transmitted to the different forwarding nodes. Then the request and the response can be calculated. If the frequency changes the different path can be changed to transmit the data to the destination.

3.2 Nodes module

3.2.1 Nodes module

In this module it makes the users to deploy their own nodes in a process to transfer a file or any text document from one user to other users to calculate the time between different nodes from different places. The nodes can be mentioned as a separate system in the network and then the network can be created to make a data transmission in the network. Then the process can be further calculating the node details and then the source and the destination can be verified using the network creation and the node deployment in it.

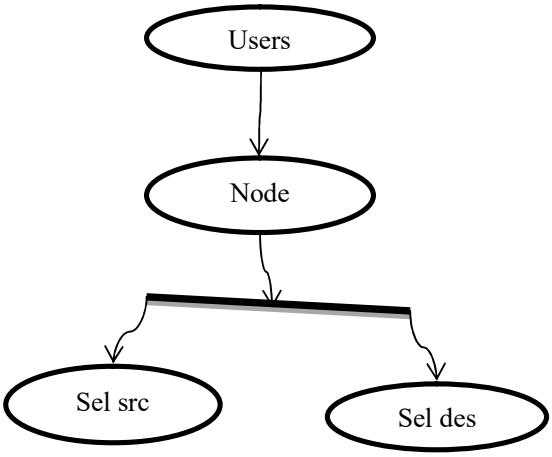
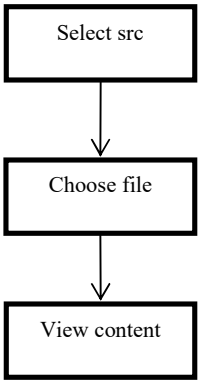


Figure 1.1. Node Module

3.2.2 File module

In this module the user can select any files to choose for transmitting from selected source node to destination nodes using a particular path that can be calculated using a traffic engineering algorithm. Then the data can be verified and then the file splitter makes the file to split the data to transmit between nodes to source to destination. Then the verification can be done after the data reached the destination. Then the merging can be done after verification. All the file in the networks can be send only in packets in it.

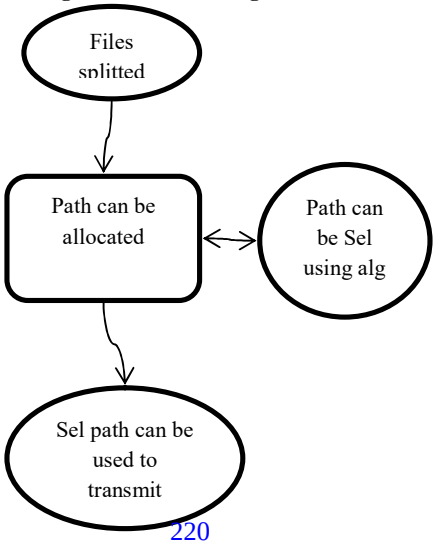
Figure 1.3 File Module



3.2.3 Forwarding module

In this module the path can be selected using a distributed algorithm to transmit a file from source to destination. To receive a file from one to another users. This module can be used as an intermediate to transmit files. Then the node verification can be done before transmitting the packets to the certain forwarding node to make the security process in the network. Then the data can be transmitted to the certain sequence at a certain frequency bandwidth in it.

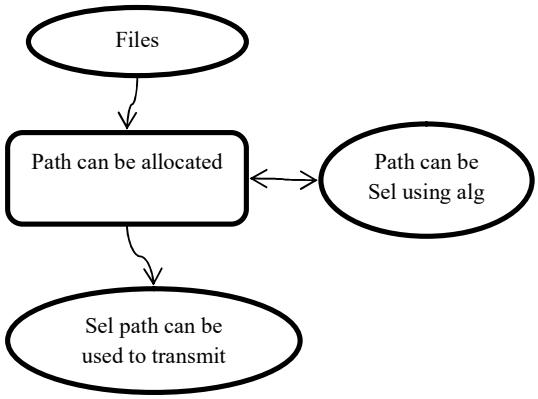
Figure 1.5: forwarding module



3.2.3 Forwarding module

In this module the path can be selected using a distributed algorithm to transmit a file from source to destination. To receive a file from one to another users. This module can be used as an intermediate to transmit files. Then the node verification can be done before transmitting the packets to the certain forwarding node to make the security process in the network. Then the data can be transmitted to the certain sequence at a certain frequency bandwidth in it.

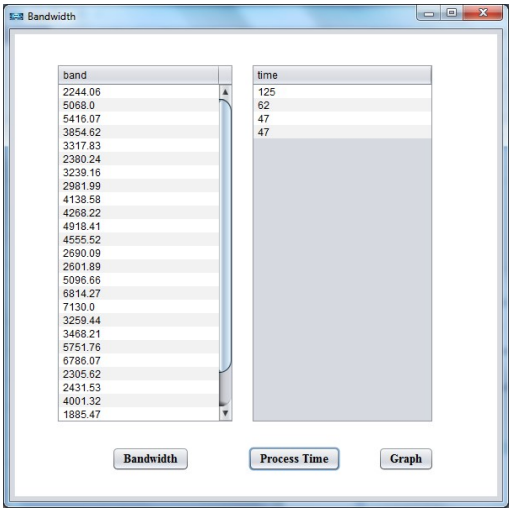
Figure 1.5: forwarding module



3.2.5 Fault module

In this module the fault of transmitting the files from source node to destination node of sensing nodes can be calculated using the distributed algorithm. Then the fault node and process time of the both averages can be calculated as a result data in it. Then the fault can be shown the node delay and the request and the response time of the node. Then the data can be process related to the calculation of node delay in it.

Figure 2.0: fault module



4. Conclusion

This paper proposes a self-detectable distributed fault detection algorithm to detect the faulty sensor nodes such as stuck at zero, stuck at one, stuck at nonzero and random fault in sensor networks. Here, each sensor node collects data from the neighbours and then diagnose itself by using the Neyman–Pearson test. The accuracy and completeness of the algorithm are analyzed by assuming the sensed data is noisy. The algorithm is implemented in NS3 and the performances are compared with the existing algorithms. From the simulation, it is evident that the algorithm detects the faulty sensor nodes with more than 98% detection accuracy for a wide range of fault probabilities and maintain a negligible (at max 6%) false alarm rate. The comparison result shows that the proposed scheme significantly improves the performance parameters for large scale sparse sensor networks as compared to that of existing algorithms. In fact, there is an 8%improvement in detection accuracy and34% improvement in false alarm rate as compared to existing algorithms. The proposed distributed fault detection scheme is efficient in terms of time complexity, message complexity, network life time, detection latency, energy consumption, detection accuracy and false alarm rate.

5. ACKNOWLEDGMENT

This is to acknowledgement that my paper prepared by myself and future enhancements will be implemented .Those are prove the scientific guidance, and Unpublished results.

6. REFERENCES

[1] Yick J, Mukherjee B, Ghosal D. Wireless sensor network survey. Comput Networks2008;52(12):2292–330.

[2] Haeberlen A, Kouznetsov P, Druschel P. The case for byzantine fault detection. In:Proceedings of the 2nd conference on hot topics in systemdependability, vol. 2; 2006. p. 5–5.

[3] Duarte Jr EP, Weber A, Fonseca KVO. Distributed diagnosis of dynamic events in artitionable arbitrary topology networks. IEEE Trans Parallel DistribSyst 2012;23(8):230–45.

[4] Banerjee I, Chanak P, Rahaman H, Samanta T. Effective fault detection and routingscheme for wireless sensor networks. Comput ElectrEng2014;40(2):291–306.

[5] Chessa S, Santi P. Crash faults identification in wireless sensor networks. ComputCommun 2002;25(14):1237–82.

[6] You Z, Zhao X, Wan H, Hung WN, Wang Y, Gu M. A novel fault diagnosis mechanismfor wireless sensor networks. Math Comput Model2011;54(2):330–43.

[7] Barooah P, Chenji H, Stoleru R, Kalmar-Nagy T. Cut detection in wireless sensornetworks. IEEE Trans Parallel Distrib Syst 2012;23(3):483–90.

[8] Guo S, Zhong Z. FIND: faulty node detection for wireless sensor networks. In SenSys'09;2009. p. 1–14.

[9] Lee M-H, Choi Y-H. Fault detection of wireless sensor networks. Comput Commun2008;31(14):3469–75.

[10] Chen J, Kher S, Somani A. Distributed fault detection of wireless sensor networks. In:Proceedings of the 2006 workshop on dependability issues inwireless ad hoc networks andsensor networks. DIWANS '06. New York (NY, USA): ACM; 2006. p. 65–72.

[11] Jiang P. A new method for node fault detection in wireless sensor networks. J Sens2009;9(2):1282–94.

[12] Panda M, Khilar PM. Energy efficient soft fault detection algorithm in wireless sensornetworks. In: IEEE international conference on parallel,distributed and grid computing(PDGC,2012); December 2012. p. 801–5.

[13] Simulator N. <http://www.nsnam.org>.

[14] Preparata F, Metze G, Chien RT. On the connection assignment problem of diagnosable ystems. IEEE Trans Commun Mag 1967;EC-16(6):848–54.

[15] Mallela S, Masson GM. Diagnosable systems for intermittent faults. IEEE TransComput 1978;C-27(6):560–6.

[16] Panda M, Khilar PM. Distributed soft fault detection algorithm in wireless sensornetworks using statistical test. In: IEEE international conference onparallel, distributed andgrid computing (PDGC, 2012); December 2012. p. 195–8.

[17] Luo X, Dong M, Huang Yl. On distributed fault-tolerant detection in wireless sensornetworks. IEEE Trans Comput 2006;55(January):58–70.

[18] Liang GJ, Jun XY, Wei LX. Weighted median based distributed fault detection forwireless sensor networks. J Softw 2007;18(5):1208–17.

[19] Ssu K-F, Chou C-H, Jiau HC, Hu W-T.Detection and diagnosis of data inconsistencyfailures in wireless sensor networks. Comput Network2006;50(June):1247–60.