A New Approach For Trust Calculation In Internet Of Things

ZAID HANOOSH [1]* , AHMED AL GHAZALI [2]

[1]university of kufa ,college of dentistry, Iraq.

[2]university of kufa, college of physical planning ,Iraq.

Abstract

IoT (internet of things) is defined as a worldwide framework for information world, which provides advanced services by connecting physical and virtual things through information technology and adoptable communication available in the evolution process. IoT is just applied to overcome the concerns related to safety. Sharing information among different devices can affect the private information of users. Therefore, a proper approach mechanism is required to prevent the risk of malicious and vulnerable failures. Multi-services IoT is vulnerable to many types of malicious attacks. Trust management provides a potential solution for safety issues of distributed networks. In this article, an algorithm is designed in order to calculate trust, which does not calculate the amount of trust for all neighbors and just calculates it for suspicious neighbors located in the list of suspicious nodes. Subsequently, energy consumption in comparison to the approach of basis article has so much decrease.

Keywords: internet of things, trust management, safety, multi-services, energy.

1. Introduction

IoT is a network, which is built by heterogeneous devices with different process capabilities that can communicate and cooperate in a smart environment. We expect that a million devices will connect to IoT in the future. This is a worldwide network in which smart things such as computers, smartphones, sensors, drivers and other everyday devices communicate and provide information in real time (Mendoza & Kleinschmidt, 2018). New samples of IoT have introduced more applications that are new and useful, for example, smart cities, smart networks and more importantly electronic health. All of these applications are with the aim of life quality improvement. However, achieving all of these devices depends on a strong safe tool for protecting these billion IoT devices (Awan, 2019). Malicious
partners can make serious threat to proper network performance by jeopardizing the reliability of this tool through fake services, denial of cooperation and other malicious behaviors. Hence, things that operate in such open and risky

environment have to make correct decisions about the degree of their trust to a specific partner; this is a vital job, but has challenges yet (Kravari & Bassiliades, 2017). Trust management is an important subject in IoT. This issue allows numerous devices and things share their opinion about trust of their partners. This is a rule to prevent the malicious effects of services, which are provided by selfish or incompatible nodes. In order to make sure that data is transferred and minimized the uncertainty to services that are available in IoT applications, devices should trust each other. In IoT framework, achieving reliability among heterogonous groups that manage multi-services is so difficult (Altaf, Abbas, Iqbal, & Derhab, 2019). Trust management with the aim of solving distributed issues related to safety has become the point of research in recent years (Wang, Bin, Yu, & Niu, 2013). There are rare researches about trust management in field of in IoT.

2. The Importance of Trust

Among the aspects of services, IoT has considered as a service provider (SP). The aim of trust management is to provide a helpful service to present qualified services for service requester (SR) with cooperation of IoT. This relation has been shown in figure 1. This is a mutual relation because trust mechanism affects both RS (for protecting the privacy) and PS (Wang et al., 2013).

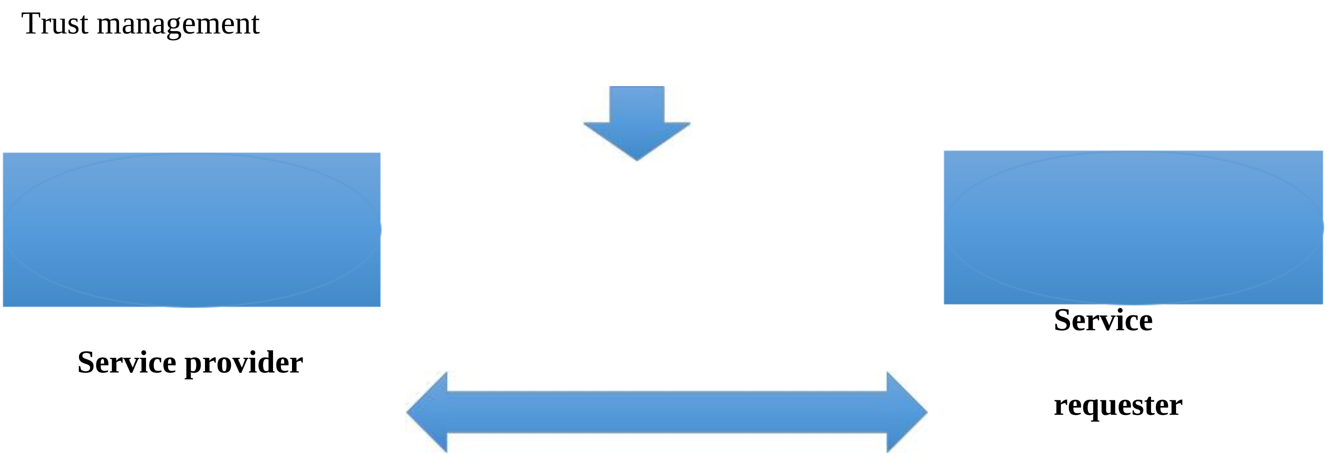

Figure 1. The services model

Traditional scales is for protecting information of Iot, coding and availability control. However, coding cannot solely solve this problem in complex and heterogonous IoT because internal vulnerable can produce fake information and yet system by using reliable coding confirms that. In the field of availabilty control, the term for entering the network is that its identity will be provided in the list of availabilty control. Anyhow, the mechanism of availibilty control to changeable behaviors is not secure specially for malicious behaviors and since traditional availibity control is centralized, so it is not appropriate for distributed environment. Nevertheles, trust management can overcome those issues. This management is based on unified mechanism, which is flexible for showing all trust relations, local control of trust relations and sepration of mechanism from policy. All componnets of network are the result of trust manageemt problem. The most popular trust management software includes trust evaluation and decision based on trust that can severly connect to IoT. Some elements and features of trust management can be extracted from reasearched solutions about trust:

Services: This feature assigns the role of trust management. The main idea of trust management is that safe decision have to trust the extra immunity information from third reliable party.

Trust as a third party easily provides services for service requesters and service provider in a network system.

Decision-making – the aim of trust management. Trust is judged because of nodes` reliability, which cooperate with each other and based on which decision it takes for providing a service, it will choose a reliable navigation and data transfer.

Self-organization. This feature shows the trust management. Based on trust management, some nodes or even sub networks can be chosen and self-organized for a special job (means sending packages, data evaluation) in terms of cooperation with each other in the network scene (means IoT). Services, decision-making and self-organization are three essential elements. Based on these features, we offer some definitions of trust mechanism.

Definition 1 (trust mechanism T). Trust management is a service mechanism that self-organized set of items based on their trust condition for conscious decision-making.

3. Related works

Many researchers has investigated trust in the field of IoT. In (Otebolaku & Lee, 2018), writers have proposed trust-centered personal services based on text information. This solution explains how text information for a person can be used for feedback process. They provide architecture, model and calculations for presenting their personal services proposed by themselves. In (Li, Song, & Zeng, 2017), writers have proposed a reliable internet and policy-based that has solutions for smart cities in which writers has described a policy-based system and has evaluated and provided the performance of their solutions.

In (Sharma, Pilli, Mazumdar, & Govil, 2016), writers have proposed a framework for trust management. Writers have provided different calculable trust models: simple statistical model, probabilistic model, and fuzzy model, model based on learning machine, flow model, graphical conceptual model and distance model. In (Mendoza & Kleinschmidt, 2018), writers have proposed a distributed trust model for multi-services IoT by using direct and indirect observations. MT design assigns positive scores for authentic nodes and negative scores for malicious nodes by using direct interactions among groups (services` request) and neighbors advices (by exchanging trust tables). Writers performed malicious nodes that have bad attacks in order to be able analyze the model effectiveness. The obtained results shows that proposed MT model determines malicious behaviors of network, while focusing on area that have %10 to %30 malicious nodes. This model may use for recognizing other common attacks in IoT, such as On-Off and assigned attacks. In (Maddar, Kammoun, & Youssef, 2018), writers have proposed a new model of intrusion detection for IoT especially for WSNs. This model observes geographical location of nodes to make sure they make connection with proper nodes for each transaction. Then, writers have proposed rules for recognizing attacks. A mathematical model for trust calculation proposed with the aim of updating trust nodes and omitting malicious nodes.

In (Mendoza & Kleinschmidt, 2018), trust is calculated in such a way that have large amounts of energy consumption and because energy consumption of nodes in IoT networks is so important, so the proposed method should consider this point. Some studies that can be used for reducing energy consumption and traffic because of trust calculation are computing trust only for suspicious nodes. Suspicious nodes can be determined by many methods also if node has abnormal traffic. So proposed method is changing the techniques of trust calculation in such a way that energy consumption will be minimized.

4. Proposed method

Our proposed method is to examine the degree of trust of all network nodes which is performed by neighbor nodes in semi-centralized and semi-distributed manner and by cooperating with trust management server, malicious nodes are determined and placed on the blacklist and serving or blocking them. Also in the method presented in the final stage, for the first time a mechanism has designed to remove nodes from the blacklist if the node is incorrectly placed on the blacklist or modify its behavior, and receive the necessary services from its neighbor and serve them (block out). Hence, we have designed 5 different phases for our proposed method, which we will describe the performance of each stage separately in below.

First phase: initialization of trust values for the rest of the neighboring nodes in each node and definition of the blacklist on the server.

Setting the value of zero as the initial value of trust: at the beginning of the algorithm, each nodes does not have sufficient knowledge of its neighboring nodes, thus it considers zero for them to indicate that they do not have any cognition of the rest of their neighbors. Because the proposed method uses a trust server for centralized trust management, a list will be defined on this server, called blacklist. This list includes sabotage nodes that have reported various suspicions of neighboring nodes. At this level, this black list is defined and there is no node in it.

Second phase: calculation of trust values for neighboring nodes in each node

In this level, all nodes calculate the trust values for neighboring nodes (the nodes that they connect to and the data sent to them through these nodes). The method of calculating trust for each node is performed through algorithm 1 (trust calculation algorithm in the basis paper).

Third phase: send a malicious node detection report

At this stage, each node after calculating the trust value will compare these values with threshold values that are defined (the threshold is equal to (– 0.5) because trust values vary between 1 and -1 and zero value means not recognizing from each node). If it was less than given value, it sends the neighboring node as the suspicious node to the trust management server.

Fourth phase: providing service management and receiving service based on trust values by trust management server

In this stage, trust management server receives reports that some nodes are suspected of their neighboring nodes. Now. If the numbers of these reports is greater than the threshold (n/2 of the number of neighbors). This node will be placed in the list of sabotage nodes, which in this list will reduce providing service to the node, will be tried to block this node, and will be minimized receiving service from this node.

Fifth phase: retrieve non-malicious nodes from the blacklist

In this level, the server will recalculate the trust for each node entering the blacklist after a specified time interval (t). This operation is because the node may be mistaken (a blunder, computational or network error such as latency and attacks, etc.) that the error should be corrected. On the other hand, it may be possible for a node to display a malicious behavior for a variety of reasons (such as unplanned mistake or programming problems or any other reason), but after that interval returns to its common and normal behavior. So sue to this, it is required to remove the blacklist. To this end, the server sends a trust recalculation message to neighbors of malicious nodes, asking them to calculate new trust values for the malicious node and send it to the server. The server will also keep the node in the list if these values are less than the threshold value, and if it was more, it will remove the node from the black list.

Our method with the method of article (Mendoza & Kleinschmidt, 2018) has some special differences that we briefly describe each of them:

1. In our proposed method, instead of calculating trust continually by neighbors and sending these trust values tables for other neighbors and updating data for each other, these trust values are calculated only at cases and specific times, which is very useful for network nodes, which are usually devices with a high energy and memory and processing power.

2. In our method, instead of sending tables of trust values for neighbors that create a large volume of traffic over the networks, only the identifier of the neighboring nodes that is

suspected is sent to the trust management server, which remarkably it reduces traffic and energy consumption in the network.

3. In the proposed method, a semi-centralized semi-distributed structure has been used to optimize the benefits of using both methods. The behavior of the nodes by their neighboring nodes, which are in close proximity with each other, is reviewed and a high-power server that perform monitoring process by considering all aspects performs final decision.

4. In the proposed method, the final step, which is to restore nodes from the black list, is used for the first time among similar articles that performs the mentioned benefits as well. Because nodes may be placed in the blacklist in wrong form or because of temporary maladministration, the corresponding behavior to these nodes should be re-evaluated after some time, and if they will be modified, the should be removed from blacklist and and return to their normal providing service and receiving service.

Algorithm 1 shows how to calculate the trust of each node by the neighbors of that node. In this algorithm, all neighbors of a node compute the trust value of that node based on the algorithm 1 in the base article (line 3), and if this value is lower than the trust threshold (line 4), this node as a the malicious node will be sent to the server (line 5).

Algorithm 1- Neighbor trust computing algorithm in each node.

```
1- T neighbors=0;
2-for (Node n: Neighbors)
3-Tn =compute Trust(n);
4-if Tn<Trust_Thresh then
5-SendToServerAsMaliciousNode(n)
6-    end if
7-    end for
```

Algorithm 2. shows how the central server detects the malicious nodes. In this algorithm, all network nodes are first asked to calculate the trust of their neighbors and submit suspicious cases according to algorithm 1 for the server (lines 5 and 6). If the node identifies one or more of its neighbors as suspicious nodes and sends to the server, the server increases the amount of reports received for malicious activity of those nodes for one unit (lines 7 to 11). Then, if the number of received reports exceeds the threshold, the server will add this node to its blacklist (lines 13 to 17).

Algorithm 2- the algorithm for recognizing blacklist by server

```
1- Black List=∅;
2- for (Node n: Network Nodes)
```

```
3-  Num Of Reporti=0;
4-  end for
5- for (Node n: Network Nodes)
6-  Receive Reports(n);
7-  If it sends some Nodes as malicious Nodes then
8-  for(Node i: malicious Nodes)
9-  Num Of Reporti++;
10-end for
11-end if
12- end for
13- for (Node i: Network Nodes)
14-if num Of Reporti> Report_Thresh then
15-Add i to Black List;
16-end if

17- end for
```

Algorithm 3. shows the way node leaving the blacklist. In this algorithm, the server first sends a request for reassessment of trust to all neighbors of the nodes that are in blacklist, and if these nodes redirect the node to the server as a malicious node, the server again adds one unit to the number of incoming reports (lines 4 to 11). If the number of reports is less than the threshold of the sent reports, it means that the node is getting out of the malicious state and is a normal node (lines 12 to 16). Of course, this limit of threshold can be stricter than the threshold set in Algorithm 2, since in recalculating the nodes' trust; the node must secure the server's credibility. This algorithm should be repeated at specific intervals, which varies depending on the network application and the type of node.

Algorithm 3- Algorithm for leaving node from blacklist server

```
1- for (Node n: Black List)
2-     Num Of Reporti=0;
3- end for
4- for (Node n: Black List)
5-     for (Node i: n. Neighbours)
6-     Request Re calculating Trust(n);
7-     If it sends n as malicious Node then
8-     Num Of Reportn++;
9-     end if
10-    end for
11- end for
12- for (Node i: Black List)
13-     if num Of Reporti<Report_Thresh2 then
14-     remove i from Black List;
```

```
15-   end if
16- end for
```

5. Simulation and evaluation of results

In order to set up a trust management simulation environment based on the JADE library (Hanoosh, 2021), you must first download and run the Helios eclipse ide simulation environment. Then the proposed method is implemented with the basis paper algorithm (Mendoza & Kleinschmidt, 2018) for an IoT network using the supplied battery life model and then the obtained results are compared.

In the first experiment, depending on the number of malicious nodes existing in the network (the ratio of malicious nodes to the total number of nodes), the required time for finding the malicious nodes is obtained as shown in Figure 2. As you can see, by increasing the number of malicious nodes, this time also increases. The results of this comparison are shown in Table 1.

Table 1. Comparison results of the mean time for detecting malicious nodes in the proposed algorithm with article [1].

proposed algorithm	**basis article [1]**	**percentage**
27	40	10
30	48	20
32	52	30

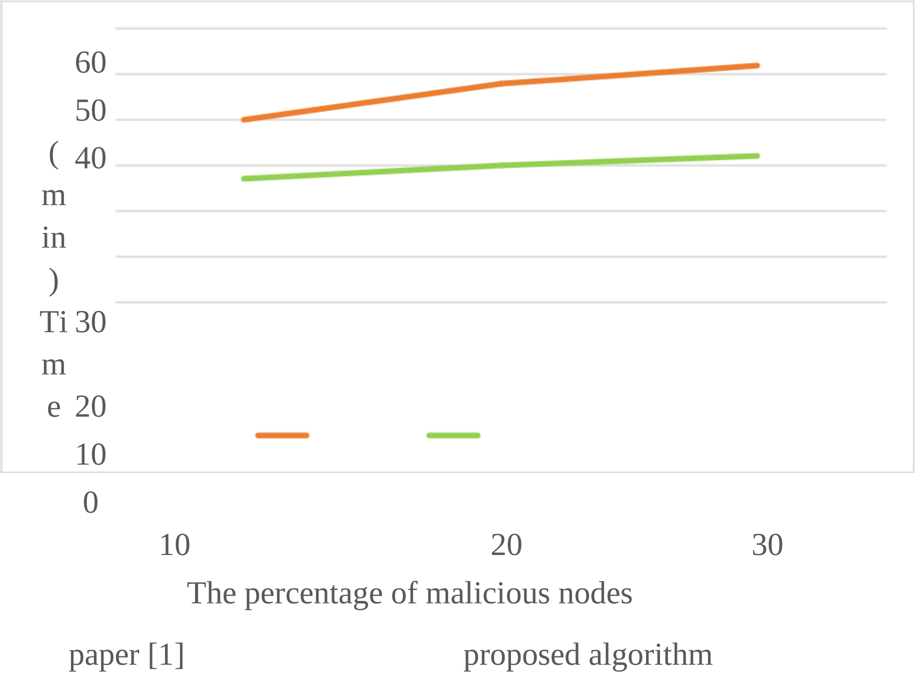

Figure 2. Comparison of the mean time for detecting malicious nodes in proposed algorithm and paper [1]

In the second experiment, the amount of energy consumption is calculated based on the number of packets sent in its frame size and the number of packets sent in its frame size. As you can see in Fig. 3, in our algorithm, given that trust is not calculated sequentially, it has a significant reduction in energy and because our algorithm creates less traffic and only by observing the suspicious behavior, it will calculate the trust. The time of detecting suspicious nodes in the network has significant reduction. Comparative energy consumption results of the proposed algorithm and paper (Mendoza & Kleinschmidt, 2018) are shown in Table 2.

Table 2. Energy consumption comparison results of the proposed algorithm with article (Mendoza & Kleinschmidt, 2018)

proposed algorithm	**basis article [1]**
3017	1534

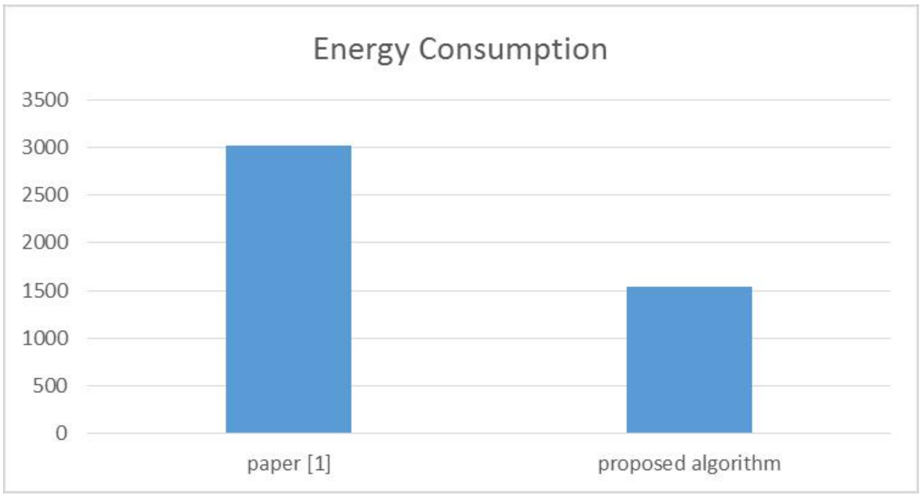

Figure 3. Energy Consumption Comparison with article [1]

6. Conclusion

In (Mendoza & Kleinschmidt, 2018), confidence is calculated in a manner that has a high energy consumption and, since in Networks of IoT, energy consumption objects of nodes has great importance; the proposed method should consider this point. One of the ways that can be used to reduce energy consumption and reduce network traffic caused by calculating trust is to only calculate the trust for nodes that are suspected. The suspicious node can also be identified in many ways, such as a node with unusual traffic. Therefore, the proposed method in this paper is to change the way of calculating trust so that energy consumption is minimized. We designed the proposed method in five different phases. In the proposed algorithm, instead of calculating trust for all neighbors, the only amount of trust for the suspicious neighbors will be computed, which is in the list of suspicious nodes. In this way, energy consumption will be reduced because calculating the trust is a process consuming energy. This process is only applicable in the nodes having an abnormal behavior. To identify these suspicious and abnormal nodes, an algorithm will be designed which is based on the amount of input and output traffic of each nodes. In this way, if the amount of traffic of an input to a node is greater than the output traffic, then that node can be considered as a suspicious node. As can be seen from the results, the proposed method has a lower energy consumption than the method presented in article (Mendoza & Kleinschmidt, 2018).

References

Altaf, A., Abbas, H., Iqbal, F., & Derhab, A. (2019). Trust models of internet of smart things: A survey, open issues, and future directions. Journal of Network and Computer Applications, 137, 93-111.

Awan, K. A., Ikram Ud Din, Ahmad Almogren, Mohsen Guizani, Ayman Altameem, and Sultan Ullah Jadoon. (2019). uted Trust Management Mechanism for Internet of Things.". RobustTrust–A Pro-Privacy Robust Distrib IEEE Access7.

Hanoosh, Z. (2021). A New Approach for Trust Calculation in Internet of Things. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(11), 6325-6332.

Kravari, K., & Bassiliades, N. (2017). Social principles in agent-based trust management for the Internet of Things. Paper presented at the 2017 19th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC).

Li, W., Song, H., & Zeng, F. (2017). Policy-based secure and trustworthy sensing for internet of things in smart cities. IEEE Internet of Things Journal, 5(2), 716-723.

Maddar, H., Kammoun, W., & Youssef, H. (2018). Effective distributed trust management model for Internet of Things. Procedia Computer Science, 126, 321-334.

Mendoza, C. V. L., & Kleinschmidt, J. H. (2018). A distributed trust management mechanism for the Internet of things using a multi-service approach. Wireless Personal Communications, 103(3), 2501-2513.

Otebolaku, A., & Lee, G. M. (2018). A framework for exploiting Internet of Things for context-aware trust-based personalized services. Mobile Information Systems, 2018.

Sharma, A., Pilli, E. S., Mazumdar, A. P., & Govil, M. (2016). A framework to manage trust in internet of things. Paper presented at the 2016 International Conference on Emerging Trends in Communication Technologies (ETCT).

Wang, J. P., Bin, S., Yu, Y., & Niu, X. X. (2013). Distributed trust management mechanism for the internet of things. Paper presented at the Applied Mechanics and Materials.