Sādhanā (2018) 43:136
https://doi.org/10.1007/s12046-018-0885-z

Dynamic trust management for secure communications in social internet of things (SIoT)

A MEENA KOWSHALYA[1,*] and M L VALARMATHI[2]

[1]Department of Computer Science and Engineering, Government College of Technology, Coimbatore 641013, India
[2]Department of Electrical and Electronics Engineering, Alagappa Chettiar Government College of Engineering and Technology, Karaikudi 630003, India
e-mail: meenakowshalya.gct@gmail.com; drmlv@gct.ac.in

MS received 22 September 2015; accepted 6 March 2018; published online 13 July 2018

Abstract. The world has faced three Information and Communication Technology (ICT) revolutions and the third ICT wave led to Internet of Things, the notion of anything, everything, anytime and everywhere. Out of the many visions of IoT, one revolutionary concept is to make IoT sociable i.e., incorporating social networking within Internet of Things. This revolution has led to the notion of Social Internet of Things (SIoT). Establishing a SIoT network or community is not so simple and requires integration of heterogeneous technology and communication solutions. This paper focuses on establishing a secure and reliable communication over nodes in SIoT by computing trust dynamically among neighboring nodes. Trust Management is an important area that has attracted numerous researchers over the past few years. The proposed DTrustInfer computes trust based on first hand observation, second hand observation, centrality and dependability factor of a node. Properties of trust such as honesty, cooperativeness, community interest and energy of a node are considered for computing trust. Also, this paper ensures secure communication among SIoT nodes through simple secret codes. Experimental results show that the proposed DTrustInfer outperforms the existing trust models significantly.

Keywords. Internet of Things (IoT); Social Networks (SN); Social Internet of Things (SIoT); trust; secret codes.

1. Introduction

Social Internet of Things (SIoT) is a new paradigm that integrates two technologies namely, Internet of Things (IoT) and Social Networks (SN). Internet of Things has enabled integration of various heterogeneous technologies and communications together; Social Networks are evolutions beyond Internet of Things. A different perspective and visualization of Internet of Things is to make it social by giving IoT a social structure and adding social responsibilities to the Things. This concept has led to what is called Social Internet of Things (SIoT). Social Internet of Things enable collaboration among objects via owners, the objects are not only smarter but also socially responsible. The authors in [1] defines Social Internet of Things as a social network of intelligent objects. The SIoT concept was introduced very recently and little research has been carried out in this field. Various frameworks and solutions exist for IoT, not all suits for SIoT. The authors in [2] have reported evolutions, applications, architecture, challenges and solutions for Internet of Things. Establishing a successful Social Internet of Things community is a complex task. Challenges like Data Management, Data discovery, Interoperability, Trust Management, Security, Privacy, Heterogeneity, and Fault Tolerance have to be handled. Social Internet of Things attracts enormous research work in these areas. The authors in paper [3] surveyed research challenges, architectures, design issues and platforms available for Social Internet of Things. Sherchan Wanita *et al* [4] has surveyed trust and its properties for Social Networks.

The proposed work focuses on establishing reliable communication with peer members of the SIoT community. Trust Management is considered crucial for SIoT. A node has to compute trust among its neighbors in order to enable trustworthy communications. A SIoT network is subject to Sybil attacks very commonly. Social networks grow enormously every year and thus malicious users also grow. A secure way of communication between nodes in a SIoT can be achieved by managing trust among nodes. If a node is trustworthy, it is assumed to be honest and give honest recommendations [5, 6]. The real challenge is to determine honest nodes and dishonest nodes. This paper exclusively presents DTrustInfer that computes trust of nodes dynamically and uses Secrete Codes to provide secure and reliable

*For correspondence

communication between nodes. The major contributions of the paper are as follows.

i) Establish trustworthy communications between nodes to ensure that the SIoT network comprises of majority of honest nodes. Few malicious nodes may be present in the network, since the SIoT network needs to be robust and a check for robustness should be made periodically.
ii) Establish secure and reliable conversation between trustworthy nodes by using simple secrete codes.

The rest of the paper is organized as follows: Section 2 presents the related work, section 3 presents about deriving trust and DTrustInfer algorithm was dealt with insection 4. Section 5 presents experimental results and in section 6, the conclusions are provided.

2. Related work

2.1 *Trust management in P2P networks*

The Trust Management solutions of P2P networks served as the foundations for Trust Management in SIoT. Numerous researches have been proposed for improving trust among peers. This paper reveals very recent and successful algorithms for trust management in P2P systems. A reputation based Trust for peer to peer communities was propsed [7]. It includes a coherent and adaptive trust model for comparing trust based on feedback. The algorithm uses feedback, total number of transaction, creditability of the feedback sources, transaction context factor and community context factor to compute trust. This algorithm is highly effective for a P2P environment and cannot be extended to IoT or SIoT where objects are dynamic. But these solutions can be modified to suit SIoT systems. Another interesting work proposed by [8] improves and encourages trust among P2P communities by managing peers reputations. It uses maximum likelihood probabilistic technique that reduces implementation overhead and ambiguous trust related semantics. Though simple and efficient, this technique is not pretty well scalable. And hence such a work cannot be extended for a SIoT system.

The authors in [9] proposed a highly scalable cluster based hierarchical trust management protocol for Wireless Sensor Networks (WSN) which detects malicious and selfish nodes. This protocol uses multi-dimensional trust attributes derived from social networks. This work served as a basic protocol for [10] classifying trust as objective and subjective trust. This work was the first technique to derive social trust from social networks in addition to Quality of Service (QoS) trust derived from communication networks. This paper considers [9] as the source in computing trustworthiness of nodes. The authors [11] present Gossip Trust which is a reputation aggregation scheme for unstructured P2P networks. By aggregating local trust this protocol computes global trust concurrently. With minor modification this protocol can be extended to structured P2P networks also. In a P2P system each peer should have the knowledge of other peers in order to decide whether or not to trust them. The authors [12] proposed a robust reputation mechanism for large scale P2P system where a peer combines several testimonials of other peers to determine trustworthiness. Without third party involvements this mechanism improves trust levels, identify malicious nodes and unreliable peers in a P2P system. For P2P networks a distributed scheme for inference of trust was proposed by [13]. The technique stores reputation information about users in a decentralized manner and uses this information to identify non cooperative users in a NICE system. Using this scheme, individual users can infer trust of other users thus leading to network reliability and trustworthiness. Trust and reputation may seem to be the same but the authors in [14] differentiate these two and propose a Bayesian network based trust model for P2P systems. Since the requirement of peers is different at various circumstances, this approach uses Bayesian networks to identify differentiated trust and combine different aspects of trust. The authors have tested the model for a file sharing scenario and it is the successful approach that used multiple facets of trust.

2.2 *Trust management in SIoT*

This section summarizes recent trust management solutions and strategies adopted for SIoT network. [10] presents an algorithmic approach of computing trust from behaviors of online social network. This paper also lists measurable trust metrics. The authors in [15] combine inferences to arrive at trust and distrust among nodes even if the nodes do not know each other. [16] presents a distributed trust management system for Internet of Things according to the three layering architecture method. The authors [17] proposed a dynamic Trust Management scheme for communication based SIoT environment. Multiple complex social relationships and basic properties of trust were used for dynamic trust management. As an extension of the previous paper, the authors in [18] consider two types of Community of Interest namely, Inter Community of Interest and Intra Community of Interest. Given these two as inputs the approach achieves best trust protocol settings. This protocol is scalable when compared to the author's previous work [18].

The authors in [19] proposed a trust and a reputation model that improves collaboration among nodes. Fuzzy sets were used to analyze the trust and reputation models. [20] proposed a fuzzy based approach to evaluate trust level across nodes in IoT. The same can be extended for SIoT. This fuzzy based approach is scalable and energy efficient. The authors in [21] have created a framework for inferring trust and distrust relationships in Online Social Networks. The network is decomposed into ego trust sub-network and mined for trust and distrust relationship. Graph data mining algorithms are employed for this purpose. Its possible to

derive various trust metrics from behavior of objects/nodes/things [22]. The proposed work takes into account honesty, cooperativeness, community interest and energy as primitive trust properties. According to these trust parameters, trust is calculated based on first hand information and second hand recommendation. Unlike [1], this paper does not derive trust from a subjective model and an objective model. A dependability factor is introduced along with direct trust, recommendation and centrality for trust calculation which helps in improving the application performance. The main difference between this paper and [1, 17] is that the use of the dependability factor which is the history of nodes behavior from other similar application environment.

3. Deriving trust

This paper proposes a novel secure framework for deriving trustworthiness among nodes in SIoT. The framework is shown in figure 1. From the experience of the user, we derive four properties of trust namely honesty, cooperativeness, community interest and energy. Using these properties direct trust and indirect trust are computed. The computed trust is analyzed based on varying weighing factors to maximize the application performance and establish a secure communication (figure 2).

3.1 *Deriving trust parameters*

The trust parameters are helpful to characterize a node according to its behavior, attitude and experience. Each trust property (honesty, cooperativeness, community interest and energy) is complimentary to each other and hence needs to be evaluated separately.

3.1a *Honesty*: Honesty of a node in the range (0,1) is considered as a prime factor since an honest node is assumed to always give proper and correct recommendation about its neighbors. This assures that the node is not malicious and helps improve the trustworthiness of the network. To evaluate honesty, a node relies on firsthand information i.e., direct trust. Direct trust is obtained by a nodes interacting with other node directly.

$D_{ij}^{honesty}(t)$ is the direct trust calculated between node i and j. Here i is the trustor and j is the trustee at time t. Node i computes $D_{ij}^{honesty}(t)$ between itself and node j.

3.1b *Cooperativeness*: Cooperative trust represents whether or not the trustee node is socially cooperative with the trustor. It is assumed that nodes with common friends are cooperative and behaves differently with others. In a SIoT environment, nodes cooperativeness can be predicted by its social ties. Socially cooperative nodes improve the application performance. Each device/object possesses a list of friends likely to be cooperative. This list will be updated by owners periodically. According to [1], the $D_{ij}^{cooperativeness}(t)$ is calculated as follows

$$\frac{friends(i) \cap friends(j)}{friends(i) \cup friends(j)}.$$

3.1c *Communite interest*: Community interest as proposed by [23] is another factor that enables communication between objects of communal interest. The objects are classified according to their parental relationships, co-work or co-location relationships. Objects with the same community interest are supposed to interact with each other very often leading to increased application performance. According to [1], the $D_{ij}^{cooperativeness}(t)$ is calculated as follows

$$\frac{community(i) \cap community(j)}{community(i) \cup community(j)}.$$

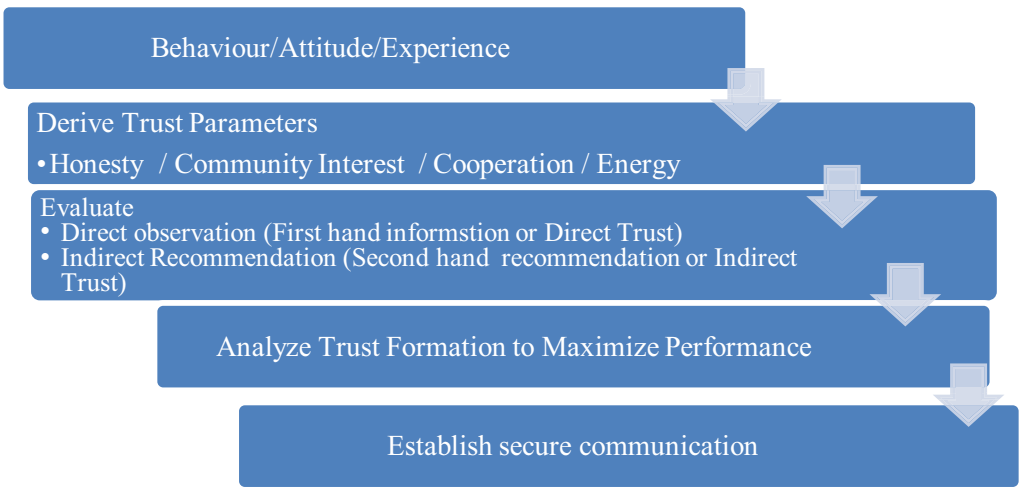

Figure 1. Process of deriving trust.

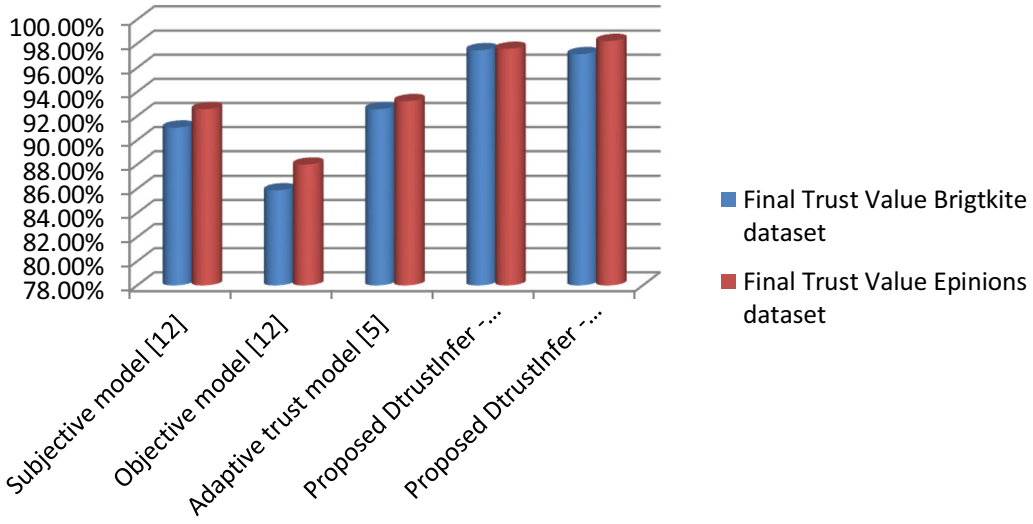

Figure 2. Comparison of trust values.

3.1d *Energy*: Energy of a node also plays an important role in communication and sharing of information. Almost all devices in SIoT are low power devices and less energy efficient devices. Thus energy of a node is to be given prime importance for collaboration purpose. Energy of a node is calculated as the product of power and time.

3.2 *Evaluate direct observations and indirect recommendations*

To improve trustworthiness among SIoT nodes, two types of trusts are evaluated namely First hand observation or Direct trust and Second hand recommendation or Indirect trust. Table 1 illustrates the parameters used for calculating trust. When nodes i and j interact directly with each other the trust is calculated as follows

$$T_{ij}^{X}(t) = \alpha T_{ij}^{X}(t)(t-\Delta t) + \alpha D_{ij}^{X}(\mathrm{t}) + \beta G_{ij}^{X} + \alpha\beta DP_{ij}^{X} \quad (1)$$

where X= honesty, cooperativeness, community interest and energy. $T_{ij}^{X}(t)$ is the past trust between i and j with respect to X. Δt is the time elapsed since the last trust update. Node i will use direct observation $D_{ij}^{X}(\mathrm{t})$ and its past trust $T_{ij}^{X}(t)(t-\Delta t)$ towards node j. Along with these two parameters node i also uses the centrality and dependability factor to compute trust. The centrality of a node is computed according to Eq. (2).

$$\text{Centrality } G_{ij}^{X} = \frac{\text{set of common friends between i and j}}{\text{Neighbors between i, j}} \quad (2)$$

Dependability factor is in the range (0, 1) which is obtained by the service provider of another similar SIoT environment. The behavior of the same node (past history) in a different environment is used to evaluate the trustworthiness. Nodes are identified by their semantics. The cost associated with retrieving the dependability factor is negligible and in few cases dependability factor will be 0 if the same objects do not participate anywhere in the outside world. When node i witnesses node k which has already experienced transaction with node j, then trust is computed according to Eq. (3). Node i uses k's recommendation to judge node j. Node i will not have any direct interaction with node j instead use recommendation of k towards j $R_{kj}^{X}(\mathrm{t})$ and the past trust value $T_{ij}^{X}(t)(t-\Delta t)$ to access j. Along with this, the centrality and dependability factor enables better computation of trust.

$$T_{ij}^{X}(t) = \gamma T_{ij}^{X}(t)(t-\Delta t) + \gamma D_{ik}^{X}(\mathrm{t}) + \gamma R_{kj}^{X}(\mathrm{t}) + \beta G_{ij}^{X} \quad (3)$$

$T_{ij}^{X}(t)\ (t-\Delta t)$ is the past trust value, $R_{kj}^{X}(\mathrm{t})$ is the recommendation that node k provides to node i about node j. There are possible chances of node k being malicious. If node k is not malicious $R_{kj}^{X}(\mathrm{t})$ equals D_{ik}^{X}. If node k is malicious it can perform bad mouthing attacks and propagate the same to node i. To prevent this happening, node i uses direct trust to access node k $D_{ik}^{X}(\mathrm{t})$ [1]. Together with these parameters, the trust is computed using centrality and dependability factor discussed in Eqs. (2) and (3).

Table 1. List of parameters used.

Parameter	Description
$D_{ij}^{X}(\mathrm{t})$	Direct trust of i towards j at time t in X
$R_{kj}^{X}(\mathrm{t})$	Recommendation of k from j at time t in X
$T_{ij}^{X}(t)$	Trust between i and j at time t in X
G_{ij}^{X}	Centrality of a node
$D_{ik}^{X}(\mathrm{t})$	Direct trust of i towards k at time t in X
DP_{ij}^{X}	Dependability factor
α, β, λ	Weighing factors

4. DTrustInfer algorithm

A SIoT network is a graph G(V,E) where V represents the number of vertices (objects/things/humans) and E represents the edges between them. The DTrustInfer algorithm takes input from a sub graph G(V,E). When a node wants to establish communication with another node, it computes and estimates the trustworthiness of the neighboring node. The node with the highest centrality is chosen as the authenticator A_i. The authenticator node A_i manages generation and distribution of Secret Codes that are to be padded with the messages. It also verifies user credentials during user churn. The algorithm begins by computing trust between nodes that need to communicate. When found to be trustworthy, the sender node pads the secret key along with the message. At the destination, the destined node separates message and secret key compares the secret key with the one that was distributed by the Authenticator A_i. Thus a check is made to ensure authentication of messages. Both trust and authentication make the SIoT network more robust. A SIoT network need not always possess honest nodes; malicious nodes may also be present within the network. Few false positives (labeling honest nodes as "Sybil's") and false negatives (labeling Sybil's as "honest") do exist in the network [24] but considered less harmful. This is needed for the system to tolerate some amount of malicious node behavior. It is proved that honest nodes are fast mixing [25] and Sybil's are not fast mixing enough like honest ones. Due to this nature, there exists a small cut in the graph between the honest region and Sybil region. This helps us easily identify the Sybil region.

Algorithm
Input: Subgraph G(V,E)
Output: Trust scores between nodes and Sybil Region
When node i wants to communicate with node j directly,
Compute trust using equation (1).
Else,
Compute trust using equation (3).
Choose the node with the highest centrality to be the Authenticator node A_i, use equation (2)
For all neighboring nodes of A_i,
    Allocate secret codes to each,
For i to establish communication between j,
    Check trustworthiness between i and j,
    Perform walks on the sub graph to identify small cuts
        if cut present
            The region is Sybil Region and contains attack edges
        Else
        Start communication by padding the secret code along the message
Repeat until done

5. Experimental results

To conduct our experiments large number of traces of objects were required. SWIM (Small World in Motion) simulator is used to generate traces of mobility of objects. The Brigtkite dataset and the Epinions dataset which are location based online social networks comprising of 1023 and 1088 nodes each were used. These traces where modified for the purpose of modeling objects that mimic human behavior. Table 1 shows trust values for the subjective model, objective model of [1], the adaptive trust model of [17] and the proposed DTrustInfer.

The configuration parameters used for the simulation environment is listed in table 2. A detailed comparative study between [10], [1] and the proposed work was done. The results are given in table 3. Experimental results show that the proposed method outperforms the other two methods and leads to increased application performance.

Table 2. Configuration parameters for Brightkite dataset.

Nodes	1023
Node radius	0.00948
Knowing time	1728000
Simulation seconds	950400
Cell distance weight	0.8
Node speed multiplier	1
Waiting time exponent	1.35
Waiting time upper bound	216000
Buckets per side	14

Table 3. Configuration parameters for Epinions dataset.

Nodes	1088
Node radius	0.009348
Knowing time	1327030
Simulation seconds	955460
Cell distance weight	0.8
Node speed multiplier	1
Waiting time exponent	1.35
Waiting time upper bound	216000
Buckets per side	14

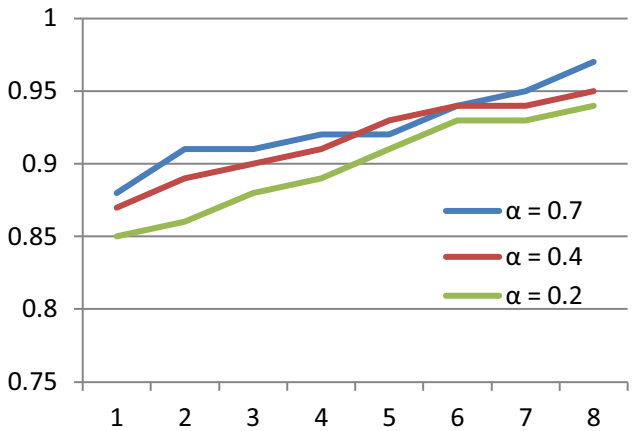

Figure 3. Non-malicious nodes with varying α values 0.2, 0.4, and 0.7. Larger the α, better the performance.

Table 4. Comparison of trust between subjective, objective, adaptive trust and DTrustInfer.

Model	Technique used	Final trust value	
		Brigtkite dataset (%)	Epinions dataset (%)
Subjective model [10]	Direct Trust	91.04	92.56
Objective model [10]	Direct trust and Recommendation	85.87	88
Adaptive trust model [1]	Direct trust and Recommendation	92.56	93.23
DTrustInfer	Direct trust, Recommendation and Dependability factor		
	a) Trust between i and j – Direct interaction	97.45	97.56
	b) Trust between i and j – indirectly using k	97.12	98.21

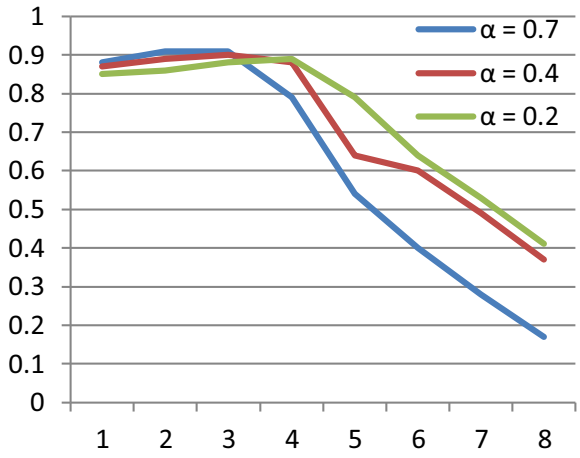

Figure 4. Malicious nodes with varying α values 0.2, 0.4, and 0.7.

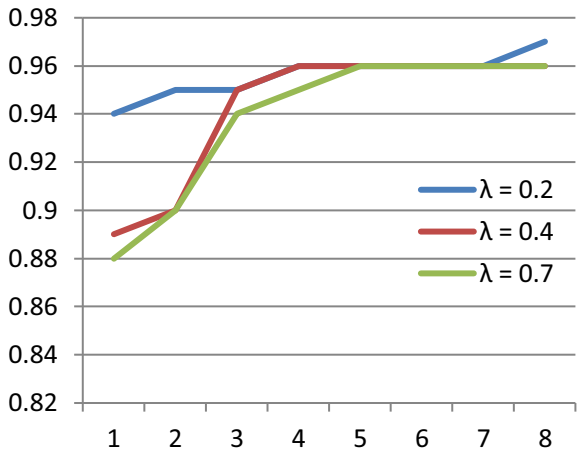

Figure 5. Non-malicious nodes with varying λ values 0.2, 0.4, and 0.7.

The Subjective model uses direct trust to compute trustworthiness of nodes. The trust of the nodes was found to be 91% since only direct interactions were involved to compute trust. This approach is the simplest and efficient method to derive trustworthiness between nodes. Since SIoT is composed of heterogeneous devices trust has to be derived even if nodes are far apart i.e., with no direct communication.

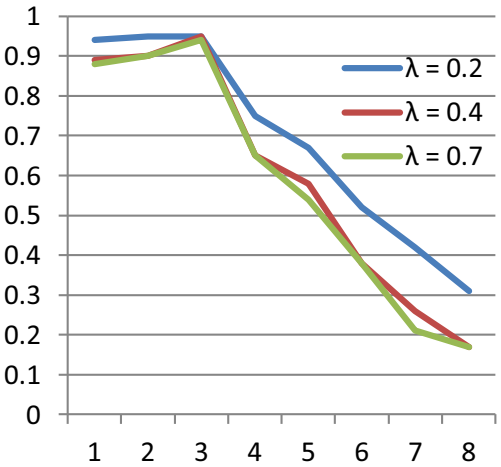

Figure 6. Malicious nodes with varying λ values 0.2, 0.4, and 0.7.

The objective model uses direct trust and indirect trust. The trustworthiness of nodes was found to be 85%. The recommendations of nodes may differ from one node to other i.e., opinions differ. This model requires trust to be globally stored by pre-trusted objects and fetched from a dynamic hash table that is practically not feasible in a SIoT environment. The adaptive trust model [1] achieves 92% trustworthiness of nodes by combining direct observation and indirect recommendations. The proposed model achieves 97% of trustworthiness between nodes. This outperforms the subjective, objective and the adaptive trust models. Figure 3 compares the subjective model, the objective model, the adaptive trust model and the proposed model. A subset of 100 nodes was chosen randomly from the generated traces of 1000 nodes and the weights were varied to analyze the performance (table 4).

For nodes i and j interacting with each other directly according to Eq. (1), α values were chosen to be 0.2, 0.4 and 0.7 and the β value was kept to be 0 to isolate its effect. The performance was compared between a non-malicious node and a malicious node. The larger the α value, application performance increases. The reason of larger α is that it has greater impact on the direct trust according to Eq. (1). Figures 4 and 5 show the behavior of

Table 5. Range of sybil and honest nodes.

	Brightkite	Epinions
Minimum degree	1	1
Maximum degree	289	267
Average value	16.32	16.23
Range of degree of Sybil nodes	1 to 5	1 to 4
Range of degree of Honest nodes	1 to 289	1 to 267
Node XL Version 1.0.1.229		

non-malicious and malicious node, respectively. It can be noted that the fluctuations are very high in figures 4 and 5. Similarly, when nodes i interact with k for evaluating node j as in figure 4, λ values chosen to be 0.2, 0.4 and 0.7 and β value was kept to be 0. λ, does not show any significant impact on the performance. This experiment was carried out only for the Brightkite dataset (figure 6).

Table 5 shows the minimum and maximum degree of nodes in the topology, the average value and the range of degree of honest nodes and Sybil nodes. NodeXL, an open source social network analysis tool was used to explore the network graph. The version of NodeXL usd was 10.0.1.229

6. Conclusion

This paper proposes a novel framework for improving trustworthiness among nodes in a SIoT environment. Since the SIoT systems are dynamically changing, computing trust is not a simple task. The paper has proposed a new model to compute trust among nodes in SIoT that uses firsthand observation (Direct Trust), Second hand recommendation (Indirect Trust), centrality and dependability of a node. The proposed model achieves 97% of trust when tested with the Brightkite and Epnions dataset. This is 6 times, 12 times and 5 times (for the Brightkite dataset) and 5 times, 9 times and 5 times better (for Epinions dataset) when compared to the subjective, objective and the adaptive trust model which were the very recent trust models for SIoT. As an extension to this, the weighing factors were adjusted to analyze the best application performance. The weighing factors α directly affects the first hand observations and hence was chosen to analyze application performance. Also, messages between nodes were strictly authenticated by secret codes. The proposed method could also efficiently find Sybil regions in the SIoT environment. As a future work, the same can be extended in a real time application.

References

[1] Bao F and Chen I-R 2012 Dynamic trust management for the internet of things applications. In: *International Workshop on Self Aware IoT*, pp. 1–6

[2] Chen D, Chang G, Sun D, Li J, Jia J and Wang X 2011 TRM-IoT: a trust management model based on fuzzy reputation for internet of things. *Comput. Sci. Inf. Syst.* 8(4): 1207–1228

[3] Valarmathi M L, Kowshalya M and Aarthi M 2015 Research challenges in the social internet of things (SIoT) – a survey. In: *Proceedings of National Conference on Science Research and Information Technology*, pp. 128–133

[4] Sherchan W, Nepal S and Paris C 2013 A survey of trust in social networks. *ACM Comput. Surv.* 45(4): 1–33

[5] Meena Kowshalya A and Valarmathi M L 2017 Trust management in the social internet of things. *Wirel. Pers. Commun.* 96(2): 2681–2691

[6] Meena Kowshalya A and Valarmathi M L 2017 Trust management for reliable decision making among social objects in the social internet of things. *IET Netw.* 69(4):75–80

[7] Xong L and Liu L 2004 PeerTrust: supporting reputation based trust for peer to peer electronic communities. *IEEE Trans. Knowl. Data Eng.* 16(7): 843–857

[8] Despotovic Z and Aberer K 2004 Maximum likelihood estimation of peers performance in P2P Networks. In: *Second Workshop on the Economics of Peer-to Peer Systems*

[9] Bao F, Chen R, Chang M J and Cho J-H 2012 Hierarchical trust management for wireless sensor networks and its applications to trust-based routing and intrusion detection. *IEEE Trans. Netw. Serv. Manag.* 9(2): 169–183

[10] Nitti M, Girau R and Atzori L 2014 Trustworthiness management in the social internet of things. *IEEE Trans. Knowl. Data Manag.* 26(5): 1–11

[11] Zhou R and Hwang K 2008 Gossip-based reputation aggregation for unstructured peer to peer networks. *IEEE Trans. Knowl. Data Manag.* 20(9): 1282–1295

[12] Yu B, Singh M and Sycara K 2004 Developing trust in large scale peer to peer systems. In: *Proceedings of First IEEE Symposium on Multi-Agent Security and Survivability (MASS)*, pp. 1–10

[13] Sherwood R, Lee S and Bhattacharjee B 2006 Cooperative peer groups in NICE. *Comput. Netw.* 50(4): 523–544

[14] Wang Y and Vassileva J 2005 Bayesian network-based trust model in peer to peer networks. In: *Agents and Peer-to-Peer Computing Lecture Notes in Computer Science*, pp. 23–34

[15] Saied Y B, Olivereau A, Zeghlache D and Laurent M 2013 Trust management system design for the internet of things: a context-aware and multi-service approach. *Comput. Secur.* 39(B): 351–365

[16] Mahalle P N, Thakre P A, Prasad N R and Prasad R 2013 A fuzzy approach to trust based access control in internet of things. In: *3rd International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace and Electronic Systems*

[17] DuBois T, Golbeck J and Srinivasan A 2011 Predicting trust and distrust in social networks, privacy, security, risk and trust (PASSAT). In: *Third International Conference on Social Computing (SocialCom)*, pp. 418–424

[18] Wang J P, Bin S, Yu Y and Niu X 2013 Distributed trust management mechanism for the internet of things. *Appl. Mech. Mater.* 2463–2467

[19] Bao F, Chen R and Guo J 2013 Scalable, adaptive and survivable trust management for community of interest based internet of things systems. In: *Eleventh International Symposium on Autonomous Decentralized Systems (ISADS)*, pp. 1–7

[20] Bachi G, Coscia M, Monreale A and Giannotti F 2012 Classifying trust/distrust relationships in online social networks, privacy, security, risk and trust (PASSAT). In: *International Conference on Social Computing (SocialCom)*, pp. 552–557

[21] Atzori L, Iera A and Morabito G 2010 The internet of things: a survey. *Comput. Netw.* 54(15): 2787–2805

[22] Atzori L, Iera A and Morabito G 2011 SIoT: giving a social structure to the internet of things. *IEEE Commun. Lett.* 15(11): 1193–1195

[23] Atzori L, Iera A, Morabito G and Nitti M 2012 The social internet of things (SIoT): when social networks meet the internet of things: concepts, architecture and network characterization. *Comput. Netw.* 56(14): 3594–3608

[24] Yu H 2011 Sybil defenses via social networks: a tutorial and survey. *Proc. ACM SIGACT* 42(3): 80–101

[25] Yu H, Kaminsky M, Gibbons P and Flaxman A 2008 SybilGuard: defending against sybil attack via social networks. *IEEE Trans. Netw.* 16(3): 576–589