A Subjective Model for Trustworthiness Evaluation in the Social Internet of Things

Michele Nitti*, Roberto Girau*, Luigi Atzori*, Antonio Iera**, and Giacomo Morabito***
*University of Cagliari, Italy, michele.nitti, l.atzori@diee.unica.it, roberto.girau@yahoo.it
**University of Reggio Calabria, Italy, antonio.iera@unirc.it
***University of Catania, Italy, giacomo.morabito@dieei.unict.it

***Abstract*— The integration of social networking concepts into the Internet of Things (IoT) has led to the so called Social Internet of Things (SIoT) paradigm, according to which the objects are capable of establishing social relationships in an autonomous way with respect to their owners. The benefits are those of improving scalability in information/service discovery when the SIoT is made of huge numbers of heterogeneous nodes, similarly to what happens with social networks among humans. In this paper we focus on the problem of understanding how the information provided by the other members of the SIoT has to be processed so as to build a reliable system on the basis of the behavior of the objects. We define a subjective model for the management of trustworthiness which builds upon the solutions proposed for P2P networks. Each node computes the trustworthiness of its friends on the basis of its own experience and on the opinion of the common friends with the potential service providers. We employ a feedback system and we combine the credibility and centrality of the nodes to evaluate the trust level. Preliminary simulations show the benefits of the proposed model towards the isolation of almost any malicious node in the network.**

I. Introduction

In the Internet of Things (IoT) [1], everything real becomes virtual. This means that each person and thing has a locatable, addressable, and readable counterpart on the Internet. These virtual entities can produce and consume services and collaborate toward a common goal. The car driver knows about the status of her car and of the roads towards her destination thanks to the autonomous communications among the sensors and actuators installed in her car, in other vehicles encountered along the path, and along the road.

These scenarios are possible with an intense interaction between objects and related services. Indeed the most fascinating applications are those where the things collaborate to realize a complex service to improve the quality of life of people. For instance, in [2] the authors introduce the idea of objects able to participate in conversations that were previously available to humans only. Analogously, the research activities reported in [3] consider that, being things involved into the network together with people, social networks can be built based on the Internet of Things and are meaningful to investigate the relations and evolution of objects in loT. In [4] and [5], explicitly, the Social IoT (SIoT) concept is formalized, which is intended as a social network where every node is an object capable of establishing social relationships with other things in an autonomous way with respect to its owner, with the potentials to solve problems of network navigability and information/service discovery when the IoT is made of huge numbers of heterogeneous nodes.

Until now, all proposals focused on the definition of the relationships and interactions among objects and on the design of reference architectures and protocols. Still paradigm lacks in some basic aspects such as understanding how the information provided by the other members has to be processed so as to build a reliable system on the basis of the behavior of the objects. In this work we address this uncertainty and analyze strategies to establish trustworthiness among nodes (i.e. things) in the SIoT. The challenge is to build a reputation-based trust mechanism for the SIoT that can deal effectively with certain types of malicious behavior aimed at misleading other nodes.

With these problems in mind, we propose a subjective trust model to construct a management system for the objects' trustworthiness, which should drive the consumption of the services and the information delivery towards trusted nodes. The major contributions of the paper are: (i) definition of the problem of trustworthiness management in the SIoT; (ii) definition of a trust model where each node computes the trustworthiness of its friends on the basis of its own experience and on the opinion of the friends in common with the potential service provider; (iii) evaluation of the benefits of the trustworthiness management in the IoT.

II. Background

A. The Social Internet of Things

The idea of using social networking elements in the Internet of Things to allow objects to autonomously establish social relationships is gaining popularity in the last years. The driving motivation is that a social-oriented approach is expected to put forward the discovery, selection and composition of services and information provided by distributed objects and networks that access the physical world. Within the resulting object social network, a key purpose is to publish information and services, find them, and discover novel resources to support the implementation of complex services and applications. This can be achieved in a trusty and efficient way by navigating a social networks of "friend" objects, instead of relying on typical Internet discovery tools that cannot scale to billions of future devices.

In this paper, without losing of generality, we refer to the social IoT model proposed in [5] (we use the acronym SIoT to refer to it). According this model, a set of forms of

978-1-4673-2569-1/12/$31.00 ©2012 IEEE

socialization among objects are foreseen. The *parental object relationship* is defined among similar objects, built in the same period by the same manufacturer (the role of family is played by the production batch). Moreover, objects can establish *co-location object relationship* and *co-work object relationship*, like humans do when they share personal (e.g., cohabitation) or public (e.g., work) experiences. A further type of relationship is defined for objects owned by the same user (mobile phones, game consoles, etc.) that is named *ownership object relationship*. The last type fo relationship is established when objects come into contact, sporadically or continuously, for reasons purely related to relations among their owners (e.g., devices/sensors belonging to friends); it is named *social object relationship*.

B. State of the Art in P2P Networks

The closest works to the topic addressed in this paper deals with the trustworthiness management in P2P networks. To calculate a peer trustworthiness, a system has to store the reputation information, encourage and decide how to share these information and utilize them to efficiently calculate a trustworthiness value.

There are different approaches that can be used to store trustworthiness information. As described in [6], all information can be stored in a centralized storage to foster sharing and managing information; however, it easily leads to a single point of failure. In [7] information is distributed in storage peers; this approach reduces the network overhead but is not able to deal with the case of a malicious node or a node with a low trust value being a storage peer. In the rater-based storage approach [8], each peer stores trustworthiness information about the peers it has observed and can then decrease the possibility of tampering with the reputation information.

For a reputation system it is important to incentivize the peers to cooperate and solve some well-known problems. A solution is proposed in [9], where a peer can buy and sell reputation information to other peers and loses credit if it behaves maliciously. When a peer decides to share its information, the system has to cope with how to share it efficiently. This problem can be differently handled: local share, part share, and global share.

Once the information is collected, it is important to use a computation system that is able to extract a reliable value of the trustworthiness. A simple mechanism consists in using the arithmetic average [10] of all the reputation values a node has received. Other models consider to weight the reputation value in different ways: in [11], the authors use different weights for acquaintance and stranger peers, while in [12] the weight is chosen based on the last reputation value a node has received; the algorithm in [13] considers the similarity between two peers in terms of released feedback to weight the reputation value. In [7], the authors assume the existence of a digraph of social links between peers, where reputation values are assigned to the link based on the transaction between the peers at the end of the link.

III. Notation and Problem Definition

In our model, the set of nodes in the SIoT is $\mathcal{P} = \{p_1, ..., p_i, ...p_M\}$ with cardinality M, where p_i represents the identity of a generic node. In our problem setting, let the network be described by an undirected graph $\mathcal{G} = \{\mathcal{P}, \mathcal{E}\}$, where $\mathcal{E} \subseteq \{\mathcal{P} \times \mathcal{P}\}$ is the set of edges, each representing a relation between a couple of nodes. Let $\mathcal{N}_i = \{p_j \in \mathcal{P} : p_i, p_j \in \mathcal{E}\}$ be the friends of node p_i, namely the nodes that share a relation with node p_i, and $\mathcal{K}_{ij} = \{p_k \in \mathcal{P} : p_k \in \mathcal{N}_i \cap \mathcal{N}_j\}$ be the set of common friends between p_i and p_j.

In the following we interchangeably refer to friend nodes and adjacent nodes to indicate two nodes that share a relation.

Let $\mathcal{S}_j$ be the set of services that can be provided by p_j. The reference scenario is represented by p_i requesting a particular service S_h. We assume that the Service discovery component in the SIoT receives the request of this service from p_i and returns a set of nodes $\mathcal{Z}_h = \{p_j \in \mathcal{P} : S_h \in \mathcal{S}_j\}$ to it that are able to provide the service S_h. For each of these potential service providers, the Service discovery component returns a set of edges $\mathcal{R}_{ij} = \left\{p_{ij}^a p_{ij}^b\right\}$, which represents the sequence of social links that constitute the selected path from p_i to p_j in the SIoT. At this point, the Trustworthiness Management component is expected to provide the key function of listing the trust level of any node in $\mathcal{Z}_h$. This is the objective of our work.

IV. Subjective trust management model

A. Basic elements

In the above scenario, we envision a subjective trustworthiness model, where each node p_i computes the trustworthiness of its $\mathcal{N}_i$ friends on the basis of its own experience and on the opinion of the $\mathcal{K}_{ij}$ friends in common. We refer to this trustworthiness with T_{ij}, i.e., the trustworthiness of node p_j seen by node p_i. If p_i and p_j are not friends then the trustworthiness is calculated by word of mouth through a chain of friendships. A node trustworthiness is determined through evaluation of its behaviour performed by the nodes in the network that interacted with it. Such reputation reflects the degree of trust that other nodes in the social network have on the given node on the basis of their past *direct* (direct interactions) or *indirect* (through intermediate nodes) experiences. To this we identify major important factors that have been derived by similar ones used in P2P networks trustworthiness algorithms:

- **A feedback system** allows a node p_i to provide an evaluation of the service it has received by the provider p_j. Feedback is represented by f_{ij}^l, which refers to each transaction l and can be expressed either in a binary way ($f_{ij}^l \in \{0, 1\}$, i.e., p_i rates 1 if it is satisfied by the service and 0 otherwise), or using values in a continuous range ($f_{ij}^l \in [0, 1]$) to evaluate different levels of satisfaction.
- **The total number of transactions** between two nodes, indicated by N_{ij}, that enables the model to detect if two nodes p_i and p_j have an abnormally high number of transactions.

- **The credibility** of node p_i, indicated as C_{ij}, represents a key factor in evaluating the information (feedback and trust level) provided by the nodes. It can assume the values in the range $[0, 1]$ where 1 represents full credibility for the node.
- **The transaction factor** ω_{ij}^{l} indicates the relevance of transaction l between node p_i and node p_j. It is used to discriminate important transactions, $\omega_{ij}^{l} = 1$, from irrelevant ones , $\omega_{ij}^{l} = 0$, and can be used as a weight for the feedback. This parameter avoids nodes to build up their trustworthiness on small transactions and then maliciously behave for an important one. In addition it can be used to discriminate the transactions and consider trusted a node only for a certain type of service.

To the above, we add other two key factors that exploit the main features of the social network among objects:

- **The relationship factor** F_{ij} indicates the type of relation that connects p_i with p_j and represents a unique characteristic of the SIoT. It is useful to either mitigate or enhance the information provided by a single friend. Table I shows the values of the relationship factor for every relation type, where higher values indicate higher trustworthiness. This is a possible setting that we use in this paper on the basis of the following reasoning. However, alternative values can be used if justified by different principles. Between two objects that belong to the same owner and then are linked by a OOR, it is very unlikely to find a malicious node and for this reason the highest factor value is assigned to this kind of relationship. Similar reasoning has been followed for the CLOR and the CWOR cases, since they are established between domestic objects or objects of the same workplace, respectively. SORs are relationships established between objects that are encountered occasionally and for this reason are associated to a smaller factor. Finally, the POR are the most risky, since they are created between objects of the same brand but that never met and depend only on the model object. If two nodes are tied by two or more relationships, the strongest relation with the highest factor is considered.
- **The centrality** of node p_i, indicated as R_{ij} (with respect to p_j). It provides a peculiar information of the social network since if a node has many relationships or is involved in many transactions, it is expected to assume a central role in the network. As described in [14], centrality is "related to group efficiency in problem-solving, perception of leadership and the personal satisfaction of participants".

A further important characteristic of the IoT members is also considered:

- **The computation capabilities** of an object, namely its intelligence I_j. It is a static characteristic of the object since it does not vary over the time but depends on the type of the object considered only. Indeed, we expect that a smart object has more capabilities to cheat with respect to a "dummy" object, and this leads to riskier transactions. Accordingly, we identify four different class of objects, where each class is defined on the basis of the computation capabilities, and assign to each class a different value, as shown in Table II: Class1 is assigned to mobile objects with great computational and communication capabilities, such as smartphones, tablets, and vehicle control units; Class2 is assigned to static objects with significant computing capabilities; objects such as displays, set top boxes, smart video cameras belong to this class; Class3 is assigned to objects with only sensing capabilities, that is, any object capable of providing a measure of the environment status. Finally, Class4 is assigned to the RFID-tagged objects.

TABLE I
TRANSACTION FACTOR

Ownership Object Relationship	OOR	0.9
Co-Location Object Relationship	CLOR	0.8
Co-Work Object Relationship	CWOR	0.8
Social Object Relationship	SOR	0.6
Parental Object Relationship	POR	0.5

TABLE II
COMPUTATION CAPABILITIES

Class 1	Smartphone, tablet	0.8
Class 2	Set top box, smart video camera	0.6
Class 3	Sensor	0.4
Class 4	RFID	0.2

B. Subjective Trustworthiness

In this approach, each node stores and manages the feedback needed to locally calculate the level of trustworthiness. This is intended to avoid single points of failures and infringement of the values of trustworthiness. We first describe the scenario of node p_i and p_j adjacent, i.e. when they share a social relationship, and we define T_{ij}, namely the trustworthiness of node p_j seen by p_i, as follows

$$T_{ij} = \alpha R_{ij} + \beta I_j + \gamma O_{ij}^{dir} + \delta O_{ij}^{ind} \tag{1}$$

where α, β, γ, and δ are used to give different weight to the different terms in the above sum, and they are such that $\alpha + \beta + \gamma + \delta = 1$ in order to keep the trustworthiness value between 0 and 1. In eq. (1) it is clear that node p_i computes the trustworthiness of its friends on the basis of their centrality R_{ij}, of their intelligence I_j, of its own direct experience, O_{ij}^{dir}, and on the opinion of the $\mathcal{K}_{ij}$ common friends with node p_j, O_{ij}^{ind}.

In this context, the centrality of p_j is defined as follows

$$R_{ij} = |\mathcal{K}_{ij}| \,/\, |\mathcal{N}_i| \tag{2}$$

and represents how much the node p_j is central in the "life" of p_i. This aspect helps to prevent malicious nodes that build up a lot of relationships to have high values of centrality. If

two nodes have a lot of friends in common, this means they have similar evaluation parameters about building relationships, even more if we consider the possibility to terminate a relationship with a very low value of trustworthiness.

When a node p_i needs information about the trustworthiness of a node p_j, it checks the last direct transactions and determines its own opinion as described in the following

$$O_{ij}^{dir} = \begin{cases} F_{ij} & \text{if } N_{ij} = 0 \\ \left(\dfrac{\log(N_{ij}+1)}{1+\log(N_{ij}+1)}\right)(\epsilon O_{ij}^{lon} + \chi O_{ij}^{rec}) + \\ + \left(\dfrac{1}{1+\log(N_{ij}+1)}\right) F_{ij} \\ \text{if } N_{ij} > 0 \end{cases} \quad (3)$$

In eq. (3) two opinions are calculated, using different sizes for the temporal windows of observation: O^{lon} for the long-term opinion and O^{rec} for the short-term opinion. Also in this case two different weights are defined for the long and short term opinion, that is ϵ and χ such that $\epsilon + \chi = 1$.

It is important to note how, even if no transactions are available for node p_i to judge the node p_j ($N_{ij} = 0$), a first evaluation has been obtained considering the type of relation that links the two nodes. When other information becomes available from the transactions between p_i and p_j ($N_{ij} > 0$), the relationship factor starts to lose its importance and eventually only the opinion built up with past transactions is considered.

The long and short-term opinions needed in eq. (3) are defined as

$$O_{ij}^{lon} = \sum_{l=1}^{L^{lon}} f_{ij}^{l} \omega_{ij}^{l} \Big/ \sum_{l=1}^{L^{lon}} \omega_{ij}^{l} \quad (4)$$

$$O_{ij}^{rec} = \sum_{l=1}^{L^{rec}} f_{ij}^{l} \omega_{ij}^{l} \Big/ \sum_{l=1}^{L^{rec}} \omega_{ij}^{l} \quad (5)$$

where L^{lon} represents the temporal window for the long-term opinion and L^{rec} the is the analogous for the short-term opinion, with $L^{lon} > L^{rec}$ and l indexes from the latest transaction to the oldest ones. Moreover, the transaction factor ω_{ij} is used to weight the feedback so to distinguish important transactions from unimportant ones. Indeed, the short-term opinion is useful when evaluating the risk associated with a node, i.e., the possibility for a node to start acting in a malicious way or oscillating around a regime value after building up its reputation. It makes possible to suddenly spoil the service requesting nodes. In fact, the long-term opinion is not sensitive enough to detect this scenario since it needs a long time to change the accumulative score.

The indirect opinion can be expressed as

$$O_{ij}^{ind} = \sum_{k=1}^{|\mathcal{K}_{ij}|} \left(C_{ik} O_{kj}^{dir}\right) \Big/ \sum_{k=1}^{|\mathcal{K}_{ij}|} C_{ik} \quad (6)$$

where each of the $\mathcal{K}_{ij}$ friends in common gives its own direct opinion of the node p_j. All these opinions are then weighted by p_i, based on the credibility C_{ik} of the node that provides it. The credibility is calculated as

$$C_{ik} = \eta O_{ik}^{dir} + \mu R_{ik} + \rho(1 - I_k) \quad (7)$$

where $\eta + \mu + \rho = 1$. From (7) we see that C_{ik} depends on the direct experience between the two nodes, on their centrality and on their intelligence. Its computation requires adjacent nodes to exchange information on their direct opinions and list of friends, which may be an issue. To reduce the traffic load, it is possible for node p_i to request the indirect opinion only to those nodes with a high credibility value.

Eqs. (2) - (7) allow us to finally compute the subjective trustworthiness in (1). Indeed, for the idea itself of subjective trustworthiness, all the formula we have shown in this section are not symmetric and in general $T_{ij} \neq T_{ji}$.

If the node that requests the service p_i and the node that provides it p_j are not close, i.e. are not in a direct relationship, then the computation of all the trustworthiness values can be done by multiplying all the trustworthiness values between adjacent nodes in the considered route from the requester to the provider, that is

$$T'_{ij} = \prod_{d=i}^{j-1} T_{d,d+1} \quad (8)$$

At the end of each transaction, p_i assigns a feedback f_{ij}^{l} to the service received. In the case of the adjacent nodes p_i and p_j, p_i directly assigns a feedback f_{ij}^{l} to p_j and also to the friends in $\mathcal{K}_{ij}$ that have contributed to the calculation of the trustworthiness by providing O_{ik}^{dir} according to the following

$$f_{ik}^{l} = \begin{cases} f_{ij}^{l} & \text{if } O_{kj}^{dir} \geq 0.5 \\ 1 - f_{ij}^{l} & \text{if } O_{kj}^{dir} < 0.5 \end{cases} \quad (9)$$

The reference node p_k receives a feedback according to the opinion value it suggested to p_i, to reward/penalize it for its advice. In case of more than one degree of separation, the intermediary nodes can propagate the feedback up to the provider, only if the previous node, i.e. the node that propagates the feedback, has a credibility greater than a threshold.

V. Experimental Evaluation

A. Simulation Setup

To conduct our performance analysis, we would need mobility traces of a large number of objects. Since this data is not available to date, we resorted on the mobility model called *Small World In Motion* (SWIM) [15] to generate the needed traces. The idea behind the use of SWIM lies in its ability to accurately match the social behavior of humans beings like it has been proven to happen when using the most popular mobility traces available in CRAWDAD [16]. However, the output of the SWIM model is a trace of the position of humans. We then assume that each user owns a set

TABLE III
SETTING OF WEIGHTS DURING SIMULATIONS

Parameter	Description	Value
α	weight of the centrality	0.15
β	weight of the object characteristic	0.15
γ	weight of the direct opinion	0.4
δ	weight of the indirect opinion	0.3
ϵ	weight of the long-term opinion	0.5
χ	weight of the short-term opinion	0.5
η	weight of the direct opinion in the credibility	0.7
μ	weight of the centrality in the credibility	0.15
ρ	weight of the intelligence in the credibility	0.15

of things that are connected to the SIoT and that during any movement the user carries half of these objects and leaves the others at home. Objects that stay at home create co-location relationships. Every node belongs to a specific model, so that objects of the same model share a parental object relationship. The other relationships are created on the basis of the owners movements.

We run the experiment with 800 nodes (by default), considering that each person possesses an average of 7 objects. Two different behaviors can be considered in a social network: one is always benevolent and cooperative so that we call the relevant node social nodes. The other one is a strategic behavior corresponding to an opportunistic participant who cheats whenever it is advantageous for it to do so; we call the relevant node malicious nodes. The percentage of malicious nodes is denoted by mp and it is set by default to $mp = 25\%$. Malicious node behaviors can be *collusive* or *non-collusive*. A node with a non-collusive behavior provides bad services and false feedback; it can occasionally choose to cooperate in order to confuse the network. We denote with mr the percentage of time in which these nodes behave maliciously (by default $mr = 100\%$). In a collusive environment, malicious nodes create groups that cooperate to grow each other trustworthiness; we suppose, for simplicity, that a group of malicious nodes is identified by nodes tied with a OOR, so that for $mp = 25\%$ the number of collusive groups is set to 32 groups. At the start of each transaction, the simulator chooses randomly the node requesting the service, and a certain percentage of nodes that can provide the service. The response percentage is denoted by res and is set to 5%. The malicious node can then be the node requesting the service, the node providing the service or the node providing its opinion about another node. Table III shows the values for the weights that have been used during simulations. Additionally, the number of transaction in the long-term (L^{lon}) and short-term opinions (L^{rec}) have been set to 50 and 5, respectively. Finally, each object randomly belongs to one of the computation capabilities classes.

After a node chooses the provider of the service on the basis of the highest computed trustworthiness level, it sends the service request to it. Depending on how the SIoT model is implemented, the service can be delivered either through the nodes that discover the service, i.e., the social network is also used to transmit the service requests and responses on top of the existing transport network (overlay structure) or hop-by-hop trough the existing communication network, i.e., the requester and the provider directly communicate (non-overlay structure). In the latter case, a malicious node can alter the service only if it is the provider. In the first case, a malicious node can interfere with the deliver of the service even if it is in the route from p_i to p_j.

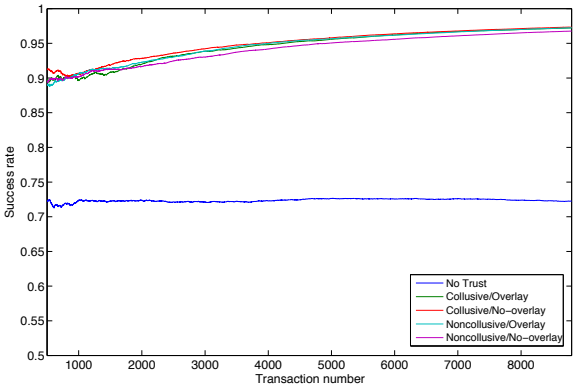

Fig. 1. Transaction success rate

B. Simulation Results

We define the transaction success rate as the ratio between the number of successful transactions and the total number of transactions. Fig. (1) shows the success rate in non-collusive and collusive scenarios while using and not using an overlay network. The case in which a trust model is not used is also presented. We can observe how in the collusive scenario, the behavior of the subjective approach is almost equal to the non-collusive case, i.e., this approach is immune to this kind of attacks. This arises from the idea itself of a subjective approach. Indeed, when a node requires the trustworthiness values of a member inside a collusive group, the only information it needs to know from other nodes, and that can then be malicious, are those related to the indirect opinion (eq. (6)), since all other information is stored locally in the node itself. This information is weighted with the credibility of the node that provides it (eq. (7)), which depends on the node own experience only.

We want now to show the robustness of our approach according to the malicious nodes concentration. In all the experiments we collect the output after 11000 casual transactions have been completed in the network so that the system is in a steady state. Then we perform 100 additional transactions to study the system behavior in response to different values of the concentration of malicious nodes mp in both non-collusive and collusive case (fig. 2) and for overlay and non-overlay structures. We can observe that there are only slight differences between the different configurations. Thus, our approach is robust to collusive behavior and it is able to isolate

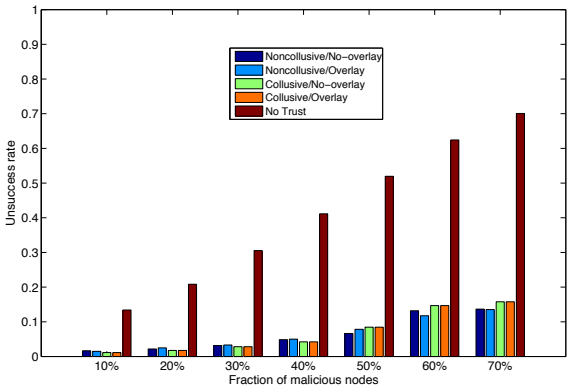

Fig. 2. Transaction percentage errore with variable mp

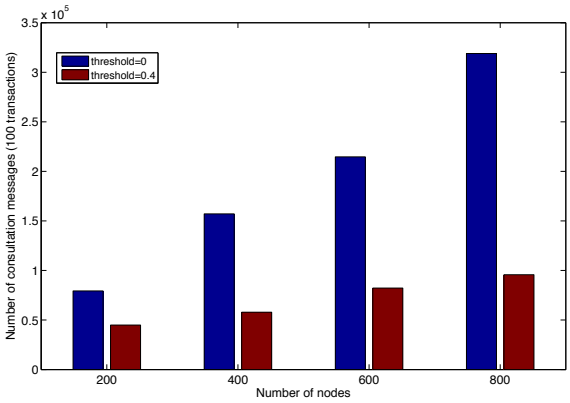

Fig. 3. Trust computation overhead

malicious nodes in the route. However, in our approach the error percentage never exceed the 15%.

So far, we demonstrated how the proposed approach deals against malicious behavior. We are now interested in evaluating the runtime overhead and how it scales with respect to the number of nodes. In our model, every node stores the information about the trust value locally. When a node needs to know the trustworthiness of another node, it uses the information about its own experience and asks to its friends for their opinion. These operations are replicated at each hop during the discovery of the nodes that can provide the service. The request for friends' opinion can be accomplished by asking to all of them (flooding) or only to that friends that have a trustworthiness above a certain threshold. The runtime overhead is then strictly correlated to the number of hops between requester and provider. The results about runtime overhead for different number of nodes and 100 transactions in this case are shown in fig. 3.

If we analyze this behavior, we have to consider that service discovery and trustworthiness computation can be carried out at the same time. Moreover, we have considered the service providers are uniformly distributed over the network, while it has been proved that friends share similar interests, the so-called homophily [17], so that it is highly probable to find a service in the friends list. These observations can reduce the runtime overhead in our approach, but, at this time we do not have enough information to take them into account.

VI. Conclusions

In this paper we have focused on the management and evaluation of trustworthiness in the SIoT context to allow objects to interact in a safe and resistant way to malicious attacks. To this end we have first analyzed the factors that influence the evaluation of trustworthiness and then we have proposed a subjective approach, where each node has its own view of the network. To demonstrate the effectiveness of our algorithm against malicious behaviors we have run a large simulation campaign and have shown strong and weak aspects under several point of views.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787 – 2805, 2010.

[2] P. Mendes, "Social-driven internet of connected objects," in *Proc. of the Interc. Smart Objects with the Internet Workshop*, 25th March 2011.

[3] L. Ding, P. Shi, and B. Liu, "The clustering of internet, internet of things and social network," in *Proc. of the 3rd Inter. Symp. on Knowl. Acquis. and Modeling*, 2010.

[4] E. Kosmatos, N. D. Tselikas, and A. C. Boucouvalas, "Integrating rfids and smart objects into a unified internet of things architecture," *Advances in Internet of Things*, vol. 1, no. 1, pp. 5–12, 2011.

[5] L. Atzori, A. Iera, and G. Morabito, "Siot: Giving a social structure to the internet of things," *Communications Letters, IEEE*, vol. 15, no. 11, pp. 1193 –1195, november 2011.

[6] P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman, "Reputation systems," *Commun. ACM*, vol. 43, pp. 45–48, December 2000.

[7] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The eigentrust algorithm for reputation management in p2p networks," in *Proc. WWW'03*. New York, NY, USA: ACM, 2003, pp. 640–651.

[8] A. A. Selcuk, E. Uzun, and M. R. Pariente, "A reputation-based trust management system for p2p networks," in *Proc. of CCGRID 2004*. Washington, DC, USA: IEEE Computer Society, 2004, pp. 251–258.

[9] R. Jurca and B. Faltings, "An incentive compatible reputation mechanism," in *Proc. AAMAS'03*. New York, NY, USA: ACM, 2003, pp. 1026–1027.

[10] Z. Liang and W. Shi, "Enforcing cooperative resource sharing in untrusted p2p computing environments," *Mob. Netw. Appl.*, vol. 10, December 2005.

[11] Y. Wang and J. Vassileva, "Bayesian network-based trust model," in *Proceedings of the 2003 IEEE/WIC International Conference on Web Intelligence*, ser. WI '03. Washington, DC, USA: IEEE Computer Society, 2003.

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

[13] L. Xiong and L. Liu, "Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities," *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, vol. 16, pp. 843–857, 2004.

[14] L. Freeman, "Centrality in social networks conceptual clarification," *Social networks*, vol. 1, no. 3, pp. 215–239, 1979.

[15] S. Kosta, A. Mei, and J. Stefa, "Small world in motion (swim): Modeling communities in ad-hoc mobile networking," in *SECON 2010*, june 2010, pp. 1 –9.

[16] J. Leguay, A. Lindgren, J. Scott, T. Friedman, J. Crowcroft, and P. Hui, "CRAWDAD data set upmc/content (v. 2006-11-17)," Nov. 2006.

[17] H. Bisgin, N. Agarwal, and X. Xu, "Investigating homophily in online social networks," in *Proc. WI-IAT 2010*, vol. 1, 31 2010-sept. 3 2010, pp. 533 –536.