PDTM: Phase-based dynamic trust management for Internet of things

Runbo Su, Arbia Riahi Sfar, Enrico Natalizio, Pascal Moyal, Ye-Qiong Song

PDTM: Phase-based dynamic trust management for Internet of things

Runbo Su, Arbia Riahi Sfar, Enrico Natalizio, Pascal Moyal, Ye-Qiong Song

Abstract—Preventing the negative effects caused by misbehaving of nodes or malicious intrusions is an essential task in trust management (TM) for the Internet of Things (IoT). Although many TM models have been proposed and developed, the majority of these approaches assigns nodes trust scores by means of a single static evaluation mechanism without taking into consideration the heterogeneity of IoT nodes and network segments, and the challenges posed by context awareness and scalability. Thus, an applicable TM model is needed to overcome these limitations. In this paper, a phase-based dynamic TM (PDTM) model is proposed. This model enables nodes' trust scores to be calculated diversely and dynamically in terms of phases. Finally, numerical results show the effectiveness and the accuracy of the proposed model, and resilience against various types of trust-related attacks (TRA).

Keywords—Attacks, Trust management (TM), Internet of Things (IoT) security, Trust-related attack (TRA), Hybrid architecture.

I. Introduction

WITH an enormous amount of effort from both academia and industry, multifarious IoT applications have been developed: smart house, health care, traffic management, etc. A growing number of smart objects and devices (thereafter referred as "nodes") gained access to the Internet, and nowadays outnumbers the world population. Despite technological advancements in IoT, several security challenges are constraining IoT systems [1]: a huge number of connected nodes in a complex environment, over wired or wireless links augments not only the system's cost, but also the risk of considerable damages. In this regard, trust management (TM) plays a crucial role, as it enables the trustworthiness of nodes to be evaluated to maintain the reliability of an IoT system.

The concept of TM was originally introduced in 1996 [2]. Till now, TM has been witnessed as one of the significant solutions for IoT security. The usage of TM for some particular network environments, such as Peer-to-Peer (P2P) systems, wireless sensor networks (WSN), and mobile ad-hoc networks (MANET), has been extensively investigated [3].

However, there are still a number of unsolved issues [4]: First, **heterogeneity**, as a TM model should meet the requirements of IoT devices heterogeneity. Second, **context awareness**, as a TM model, effective for one application, may not be applied in multi-application/service cases. Third, **scalability**, as the increasing size of the IoT networks needs a feasible mechanism for processing a large amount of data. Fourth, **dynamicity**, which concerns the static assignment of trust scores and dynamic behavior of nodes over time discordance. Fifth, **resilience**, where the TM models should provide robustness, accuracy, effectiveness against TRA, but only few works consider the resilience of TM towards TRA.

In this context, we propose a phase-based dynamic TM (PDTM) model. Our contribution is fourfold. First, we introduce a clustering based architecture addressing the scalability issue. Second, we propose a trust computation scheme established on 4 phases model fitting various contexts. Third, for dynamic trust evaluation, we define the quality of service provider and rater, and their quantity factors depending on the number of services provided and rated. Finally, we design a node classification mechanism distinguishing effectively and accurately normal/bad nodes, weak/bad service provider/rater, and TRA attackers.

The rest of this paper is organized as follows. Section II reviews related contributions on TM models. Section III discusses the framework of the proposed model. Section IV explains the PDTM model in detail and demonstrates the usefulness of each phase. The simulation results and performance analysis are presented in Section V. Section VI draws the conclusion and an outline of the future work.

II. Related works

To solve the trust issues in an IoT context, diverse methods and technologies have been proposed and developed. A centralized context-aware TM model for a multi-service environment was proposed by Saied *et al.*, [5]. In this work, feedback is collected by a trusted manager, which updates the trust value by taking into account the quality of recommendation (QR). Chen *et al.* [6] designed a fuzzy TM model, named TRM-IoT, which enhances cooperation between sensor devices. The reputation of a node is computed by evaluating the end-to-end packet forwarding ratio, packet delivery ratio, and energy consumed. To conduct trustworthiness management in social IoT, Nitti *et al.* [7] proposed both subjective and objective methods counting social attributes, such as centrality, the community of interest , and friendship. Jayasinghe *et al.* [8] designed a data-centric framework to alleviate the effect of data inconsistency. Alshehri *et al.* [9] proposed a clustering-based model, called CITM-IoT, using Master Node (MN) to manage cluster nodes, and Super Node (SN) to address the cluster allocation of MN. Du *et al.* [10] presented a trust authorization monitoring model aiming at reducing the overall energy consumption and adjusting the malicious node detection thresholds dynamically, by employing the BP algorithm. Fang *et al.* [11] designed a fast and efficient TM scheme (FETMS) based on a distributed architecture to compute trust values using Bayes distribution. Although these works show significant results in Trust Management by applying diverse methods and technologies, they are still facing several limitations: only the authors in [7]

and [10] consider the heterogeneity of IoT system, and only the works of [5] and [8] address the context-aware issue. Furthermore, [8] lacks dynamicity and scalability, [5] and [6] also lack scalability. For the resilience, TM models of [5], [9], [11] treated the On-Off Attack (OOA), [5], [7] and [10] proposed solutions handling both Bad Mouthing Attack (BMA) and Ballot Stuffing Attack (BSA). The comparison between these TM models and our work is given in Table I.

TABLE I: Summary of related TM models

Ref.	H	CA	D	S	R				
					OOA	CBA	BMA	BSA	NCA
[5]	-	x	x	-	x	-	x	x	-
[6]	-	-	x	-	-	-	-	-	-
[7]	x	-	x	x	-	-	x	x	-
[8]	-	x	-	-	-	-	-	-	-
[9]	-	-	x	x	x	-	-	-	-
[10]	x	-	x	x	-	-	x	x	-
[11]	-	-	x	x	x	-	-	-	-
Our work	x	x	x	x	x	x	x	x	x

H = Heterogeneity, CA = Context Awareness, D = Dynamicity, S = Scalability, R = Resilience, OOA = On-off Attack, CBA = Conflicting Behavior Attack, BMA = Bad mouthing Attack, BSA = Ballot Stuffing Attack, NCA = Newcomer Attack, x = Addressed

III. System Model

In this section, we present the clustering architecture of the PDTM model, an overview of PDTM phases, and the attack model. All notations and acronyms used in this paper are gathered in Table II.

A. PDTM architecture

TM architectures can be mainly classified into two categories: centralized and distributed. Both have their own strengths and shortcomings; and some researches suggest combining both architectures [12], because neither full distributed nor full centralized models are fully optimal when facing the aforementioned issues. For this reason, clustering techniques in IoT have been introduced with the aim of balancing the resource loading, increasing the network scalability, and achieving efficient communications [13].

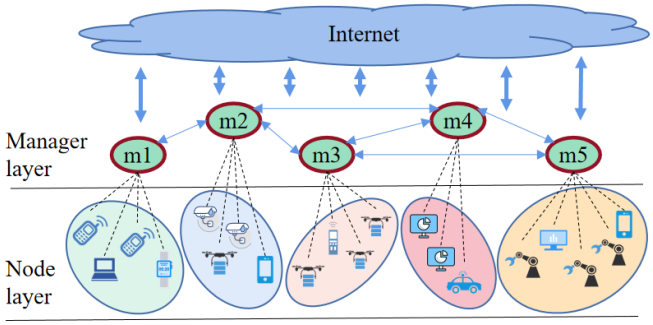


Fig. 1: Architecture of PDTM model

In the architecture of our proposed model, illustrated in Fig.1, IoT nodes are grouped into different clusters by community interests, and nodes controlled by the same manager can participate in cooperative services. Therefore, the PDTM architecture allows heterogeneous nodes with various properties to create links with each other in a multi-application environment. As a local central entity, managers are in charge of local TM and nodes access control. Moreover, inter-manager communication and trust evaluation are conducted in a distributed manner.

TABLE II: Symbol description

Symbol	Meaning
DS	Default score
PS	Pre-selection score
TS	Trust score
(a) QSP	Quality of service provider
(b) QSR	Quality of service rater
f_{ji}	Feedback from j to rate service of i
CN	Connected nodes in current cluster
R_{ji}	Nodes that rated i's service
R_{ij}	Nodes rated by i
A	Attributes
F	Factors of attributes
fct	Function
sco	Social
ctx	Context
CF	Conformity
T	Tag
S	Service type
C	Capability
ssp	Software specification
loc	Location
Des	Description of node
Pre	Prediction of node's return
msg	Message combining Pre and Des
PCF	Pre-selection conformity
MCF	Inter-manager conformity
G	Centrality
CO	Cooperativeness
N	Neighbors
MT	Inter-manager trust score
α, β	Quantity factors of (a) and (b)
SR_i	Set of services provided by i
SR_i	Set of services rated by i
ω, η, ε, φ	Weight factors
θ, λ	Behavior stability factors
p	Punishment factor

B. PDTM phases

Since nodes are able to perform dynamically over time, the trust scores should be determined distinctively depending on a concrete context. For instance, the prediction of newcomers' trustworthiness based on roughly assigning a fixed value will create unfairness. Similarly, estimating if nodes are qualified to assist in multi-service cases should be integrated into the TM model. To handle the cold-start and initialization issues, **access control** phase judges if the comer node respects the cluster community interest by computing the default score (DS) by means of gathered attributes. The **pre-selection** phase ranks adequate service providers by computing their pre-selection score (PS) to measure the relation between request mission

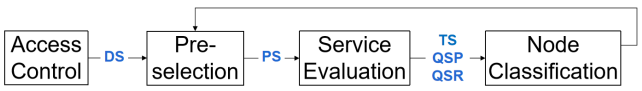


Fig. 2: Four phases of PDTM model

and node's function. In the **service evaluation** phase, nodes act as service providers and raters (QSP,QSR) and their overall trust score (TS) will be dynamically assessed by updating quantity and quality factors. Finally, the **node classification** phase categorizes the well- and badly-performing nodes, and malicious attackers through the use of a node classification scheme.

C. Attack model

In the simulation, we are concerned with the following types of TRA:

- Newcomer attack (NCA): attacker re-entries the system with a new identity to refresh its trust score.
- Inconsistent Behavior Attacks (IBA) can be of two distinct types:
  - On-off Attack (OOA): attacker switches its behavior between good and bad to maintain its trust score above a certain threshold.
  - Conflicting behavior Attack (CBA): attacker performs differently with different nodes.
- Unfair Rating Attacks (URA) also consist of two distinct kinds:
  - Bad mouthing attack (BMA): attacker sends false feedback to decrease trust score of good service provider.
  - Ballot stuffing attack (BSA): attacker gives high recommendation for malicious nodes to increase their reputation.

IV. Trust computation

In this section, we detail the trust computation procedure of the proposed TM model. We start by presenting local TM in each phase. We then explain the inter-manager TM.

A. Local TM

*1) **Access control**:* Recording the node by its asserted attributes is more advantageous in terms of security [14]. In our model, the attributes include 3 factors: the function factor (F_i^{fct}), the social factor (F_i^{sco}), and the context factor (F_i^{ctx}). The set of attributes of node i is denoted as:

$$A_i = \langle F_i^{\text{fct}} |\ F_i^{\text{sco}} |\ F_i^{\text{ctx}} \rangle. \tag{1}$$

There exists a large number of attributes that can be exploited in access control, such as nodes' role (sensor/actuator/hybrid), events' timestamp, user information and preference, durability, etc. In our model, we consider service types (S), capabilities (C) for function factor, software specification (ssp) for social factor, and node's location (loc) for context factor. For any node i, we can compare the attributes and cluster community interests by means of the conformity (CF) of 3 factors, given respectively by (2), (4) and (5).

- Conformity of function factor (CF_i^{fct})

$$CF_i^{\text{fct}} = \frac{1}{|CN|} \sum_{k \in CN} \frac{|T_i \cap T_k|}{|T_i \cup T_k|}, \tag{2}$$

where $|A|$ denotes the cardinality of the set A, CN is connected nodes in the current cluster and T_i is defined as a set of pairs of service types and capabilities of node i:

$$T_i = \{< s, c >: s \in S_i, c \in C_i\}. \tag{3}$$

- Conformity of social factor (CF_i^{sco})

$$CF_i^{\text{sco}} = \frac{1}{|CN|} \sum_{k \in CN} v_{ik}^{ssp}, \tag{4}$$

where v_{ik}^{ssp} is a binary value describing if the interaction between i and k is adoptable in terms of their software specification.

- Conformity of context factor (CF_i^{ctx})

$$CF_i^{\text{ctx}} = \frac{1}{2} \cdot (1 - \frac{loc_{im} \cdot loc_{ic}}{\|loc_{im}\| \times \|loc_{ic}\|}), \tag{5}$$

where loc_{im} and loc_{ic} correspond to the relative locations, i.e., $loc_{ab} = loc_b - loc_a$. In such manner, the ideal position is between the manager and the centroid, comer node situated too far from the manager or the centroid will not gain a great value. The centroid location loc_c is computed as follows:

$$loc_c = \frac{1}{|CN|} \sum_{k \in CN} loc_k. \tag{6}$$

With these conformity values, the default score of node i (DS_i) can be computed as follows:

$$DS_i = \omega^{\text{fct}} \cdot CF_i^{\text{fct}} + \omega^{\text{sco}} \cdot CF_i^{\text{sco}} + \omega^{\text{ctx}} \cdot CF_i^{\text{ctx}}, \tag{7}$$

where the ω variables are weights, such that $\omega^{\text{sco}} + \omega^{\text{fct}} + \omega^{\text{ctx}} = 1$. Fig. 3(a) demonstrates that the comer node is authorized to entry if $DS_i > 0.5$, the manager stores the description (Des_i) by adding trust fields after A_i:

$$Des_i = \langle A_i |\ TS_i |\ QSP_i |\ QSR_i \rangle. \tag{8}$$

Then, the node's description Des_i is included in the list of connected nodes (CN), the manager updates it according to the results issued during the service evaluation phase.

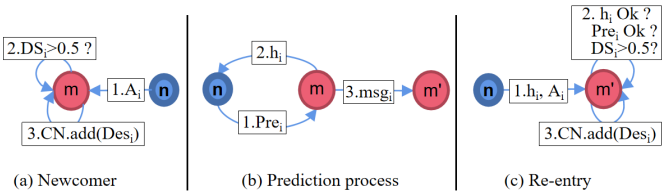


Fig. 3: Checking mechanism and prediction process in access control phase

Nodes can voluntarily leave or change cluster, e.g., an energy-constrained device, which left for being recharged and then reconnects to the network, should be treated as a returner rather than as a newcomer. For this purpose, the node must predict its return by informing the destination manager of the target location and time: Pre_i=$\langle \text{M}_\text{i}^{\text{re}} | \text{loc}_\text{i}^{\text{re}} | \text{t}_\text{i}^{\text{re}} \rangle$. The manager generates a message combining the prediction and the description: msg_i=$\langle \text{Pre}_\text{i} | \text{Des}_\text{i} \rangle$, then gives to the node a value, computed by one-way hash function h_i=$\text{Hash}(\text{msg}_\text{i})$ and transmits msg_i to the destination manager.

As shown in Fig. 3(c), the destination manager verifies the satisfaction of certain conditions: i) the received value==$\mathrm{Hash}(\mathrm{msg_i})$; ii) the information in prediction matches its entry; iii) the DS obtained by (7) $>$ 0.5. Nodes from the same cluster may continue to be assessed by the previous TS, QSP and QSR; and nodes from other clusters can only benefit previous QSR.

The access control phase in PDTM model enables to assign nodes a default score with conformity values by evaluating their attributes, and to distinguish the newcomer, returner of the same cluster, and cluster changer by usage of the triple checking rule illustrated in Fig. 3(c). Hence, newcomer attacker can not whitewash their reputation and nodes' attributes can be utilized in following trust evaluation as well.

*2) **Pre-selection**:* Upon receiving a request mission from nodes with QSR>0.5, the manager computes first the pre-selection conformity value by comparing the function factors of the candidate node and target mission:

$$PCF_i = \frac{|T_i \cap T_{tgt}|}{|T_i \cup T_{tgt}|}. \tag{9}$$

The manager can establish a list by ranking pre-selection scores (PS_i):

$$PS_i = PCF_i \cdot QSP_i. \tag{10}$$

The value of DS_i computed by (7) during the access control is used to replace QSP_i when a node is a newcomer. Finally, the manager selects the best-ranked nodes for service provision according to the requirement.

*3) **Service evaluation**:* Service raters j (including consumer nodes and assisting nodes in cooperative case) should send a feedback f_{ji} for service provider i to the manager with a note in the range of $[0, 1[$ (0 means no service conducted from service provider).

a) Trust score (TS): We set

$$TS_i = \alpha_i \cdot QSR_i + (1 - \alpha_i) \cdot QSP_i, \tag{11}$$

for

$$\alpha_i = \frac{|SR_i|}{|SR_i| + |SP_i|}, \tag{12}$$

where QSR_i and QSP_i are given by (13) and (15).

b) Quality of service rater (QSR):

$$QSR_i = \varphi \cdot CQSR_i + (1 - \varphi) \cdot LQSR_i, \tag{13}$$

$$CQSR_i = 1 - \frac{1}{|R_{ij}|} \sum_{j \in R_{ij}} |f_{ij} - \bar{f}_j|^{1/p}, \tag{14}$$

where φ belongs to $[0.5, 1[$, $\bar{f}_j$ is the average of j's notes, $CQSR$ and $LQSR$ are current and last values of QSR. $LQSR$=DS for newcomers.

The calculation of QSR is based on the comparison between the opinion of the rater node and the average value of other raters, which enables to distinguish the dishonest service raters. Therefore, an unfair rating from a dishonest rater either ruins a well-behaved node's reputation or boosts a misbehaved node's reputation, will be detected.

c) Quality of service provider (QSP):

$$QSP_i = \varepsilon \cdot CQSP_i + (1 - \varepsilon) \cdot LQSP_i, \tag{15}$$

where ε is set in the range $[0.5, 1[$, to weight the current value ($CQSP$) and the last value ($LQSP$). $LQSP$=DS for newcomers.

$$CQSP_i = \frac{1}{|R_{ji}|} \sum_{j \in R_{ji}} \theta_{ji} \cdot \lambda_{ji} \cdot QSR_j \cdot f_{ji}. \tag{16}$$

The unstable behaviors will be punished in terms of the time by θ_{ji} and service consumers by λ_{ji}. In other words, the unique opportunity that the node gains reputation is to keep providing satisfying services in a steady fashion.

$$\theta_{ji} = \mathrm{sinc}(1 - f_{ji}) \cdot \mathrm{sinc}(\Delta f_{ji})^{\Delta t} \tag{17}$$

$$\lambda_{ji} = 1 - |f_{ji} - \bar{f}_i|^{1/p}, \tag{18}$$

where Δt and Δf_{ji} are time gap and difference of last feedback (lf_{ji}) and present feedback (cf_{ji}), $\Delta t = t_{cf_{ji}} - t_{lf_{ji}}$ and $\Delta f_{ji} = |cf_{ji} - lf_{ji}|$, we set both 0 for the first time evaluation of newcomers. $\bar{f}_i$ is the average of i's notes rated by others and p is a punishment factor. The sinc function is defined as follows:

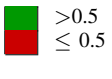
$$\mathrm{sinc}(x) = \begin{cases} 1, & \text{for } x = 0 \\ \frac{\sin(\pi x)}{\pi x}, & \text{for } x \neq 0 \end{cases} \tag{19}$$

There are two reasons why we choose the sinc function in (17). First, the range of the function is [0,1] for x in [0,1] because the function continues at 0. Second, the changes in the slope of the sinc function can be utilized to penalize the large Δf_{ji} and poor f_{ji}.

*4) **Node classification**:* By classifying the values of TS, QSP and QSR under good (>0.5) and bad (≤0.5), the node classification scheme illustrated in Table III enables the manager to categorize nodes into 6 groups: (a) Normal node (NN), (b) weak service rater (WSR), (c) weak service provider (WSP), (d) bad service rater (BSR)/ unfair rating attacker (URA), (e) bad service provider (BSP) / inconsistent behavior attacker (IBA), and (f) bad node (BN)/ mixed type attacker (MTA).

TABLE III: Node classification scheme

TS	QSP	QSR	Category
			(a) NN
			(b) WSR
			(c) WSP
			-
			-
			(d) BSR / URA
			(e) BSP / IBA
			(f) BN / MTA



From a global perspective, weak service provider or rater cases can be considered as intermediate inspection phases, because of $TS > 0.5$. Nodes belonging to the bad service provider or rater, or TRA attacker, namely group (d), (e), and (f), must be removed immediately in order to isolate the malicious nodes and prevent their negative effects.

B. Inter-manager TM

As heads of clusters, managers are somehow relatively small in quantity, and there is no service provision or rating unlike the local environment. Therefore, the inter-manager TM is aimed at reducing the cost of inter-manager communications and simplifying the cluster change for nodes, by identifying 'distant' clusters with the help of centrality (G_{mp}), cooperativeness (CO_{mp}), and inter-manager conformity (MCF_{mp}), given by (21), (22), and (23), respectively. The inter-manager trust (MT) of p evaluated by m can be computed as follows,

$$MT_{mp} = \eta^{\mathrm{G}} \cdot G_{mp} + \eta^{\mathrm{CO}} \cdot CO_{mp} + \eta^{\mathrm{fct}} \cdot MCF_{mp}, \quad (20)$$

where η variables are weights, such that $\eta^{\mathrm{G}} + \eta^{\mathrm{CO}} + \eta^{\mathrm{fct}} = 1$.

- Centrality

$$G_{mp} = \frac{|N_m \cap N_p|}{|N_m \cup N_p|}, \quad (21)$$

where for all i, N_i denotes the set of neighbors of i.

- Cooperativeness

$$CO_{mp} = \frac{AC_{pm} + 1}{AC_{pm} + DC_{pm} + 2}, \quad (22)$$

where AC_{pm} and DC_{pm} correspond to accepted and rejected nodes from p to m, respectively. The changing cluster scheme is discussed in Section IV-A1.

-Inter-manager conformity

$$MCF_{mp} = \frac{|TC_m \cap TC_p|}{|TC_m \cup TC_p|}, \quad (23)$$

where TC_i represents the union of all the tags of the connected nodes controlled by the manager i in the current cluster, i.e.

$$TC_i = \bigcup_{k \in CN_i} T_k \quad (24)$$

where the definition of the T_k's is unchanged as in Section IV-A1.

V. Simulation

This section analyses and verifies the performance of the PDTM model through simulations. First, we present the choice of simulation parameter values with their explanation. Then, we conduct the performance analysis of local TM per phase and of inter-manager TM.

A. Simulation Setup

As shown in Table IV, we set ω^{fct} 0.6 because the function factor is more relevant than other factors. For η^{CO}, we consider the cluster change, i.e., the cooperativeness value is more significant to demonstrate the inter-manager trustworthiness. Respecting the constraints of $\eta^{\mathrm{G}} + \eta^{\mathrm{CO}} + \eta^{\mathrm{fct}} = 1$ and $\omega^{\mathrm{sco}} + \omega^{\mathrm{fct}} + \omega^{\mathrm{ctx}} = 1$, we assign other parameters 0.2. ε and φ are given 0.5 for the reason that the last and the current evaluations are equally important. Finally, we set p 2.

TABLE IV: Simulation parameters

Parameter	Value	Parameter	Value
ω^{sco}, ω^{ctx} η^{G}, η^{fct}	0.2	ω^{fct} η^{CO}	0.6
ε, φ	0.5	p	2

B. Performance analysis per phase

1) Access control phase: Section IV-A1 presents the access control phase, which enables the default score assignment and triple checking mechanism for comer nodes. Fig.4 shows different scenarios for comers: (a) describes a node that exits and re-enters the system several times without checking mechanism, which means every entry must trigger the request of a new DS, to prevent it benefiting from previous trust values; (b) gives the example of detection of a returner from the same cluster, thus, this type of node can keep previous trust values. Different from (b), in the case (c), the node can only use its QSR in case of cluster changes. Finally, both (d) and (e) show denied entries, because of giving the false message and new DS <0.5, respectively.

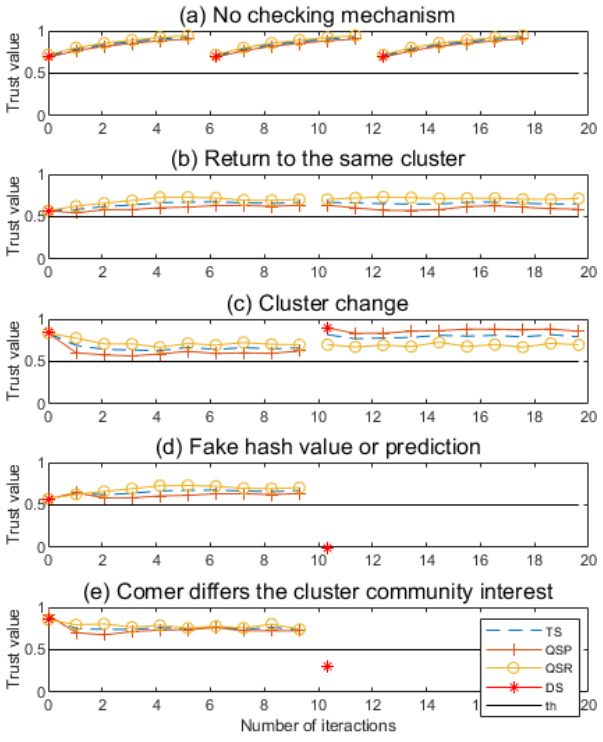


Fig. 4: Changes in trust values in five scenarios of access control.

2) Pre-selection phase: The pre-selection phase is aimed at accurately selecting the qualified service provider in order to avoid services from incapable or malicious nodes. Fig. 5 (a) demonstrates a capture of pre-selection conformity PCF and quality of service provider QSP values. The pre-selection scores PS illustrated in Fig. 5 (b) are used to rank the service provider to reject unqualified nodes with the help of two aforementioned values. The nodes with poor QSP or PCF will obtain low values for their PS, which is given by (10). Due to the fact that the manager selects the best-ranked candidates with high PS, nodes with low rank have little opportunity to provide service.

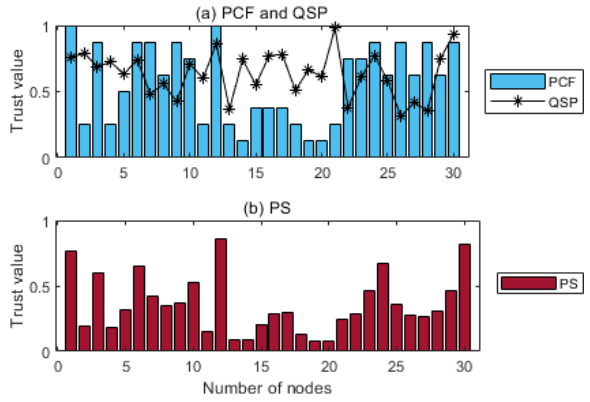


Fig. 5: An example of pre-selection score (PS) computation based on candidate nodes' pre-selection conformity (PCF) and quality of service provider (QSP)

3) Service provision & node classification phases: Following the node classification scheme presented in Section IV-A4, all nodes can be categorized into six categories, as illustrated in Fig. 6.

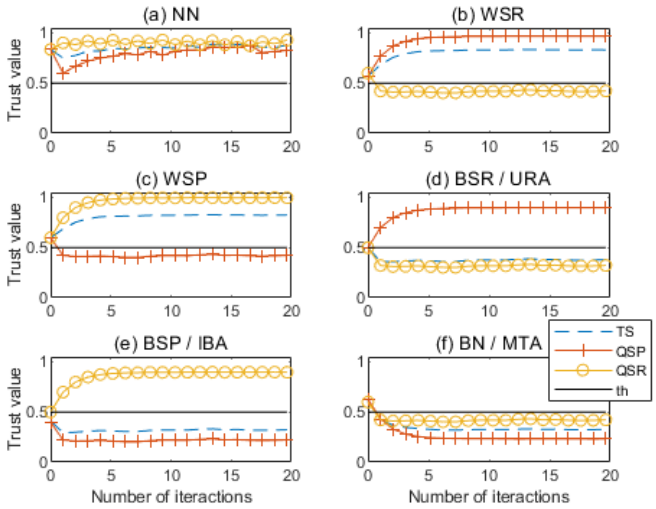


Fig. 6: Trust evaluations of six categories: (a) normal node (NN), (b) weak service rater (WSR), (c) weak service provider (WSP), (d) bad service rater (BSR)/ unfair rating attacker (URA), (e) bad service provider (BSP)/ inconsistent behavior attacker (IBA), and (f) bad node (BN)/ mixed type attacker (MTA). In order to visualize the changes in trust values in this figure even if some of the trust values are lower than 0.5, TM model continues their trust evaluation.

C. Resilience

1) NCA: The countermeasure for NCA is presented in Section IV-A1, the simulation results in Fig. 4 confirm its effectiveness.

2) OOA: Fig. 7 shows the OOA attacker behaves alternatively to keep its QSP above the threshold of 0.5. With the help of θ, the manager can detect the OOA attacker by measuring the stability of behaviors in terms of the time and punishing the services without good feedback. As discussed in Section IV-A3c, service provider nodes can only gain reputation by keeping stably providing good services.

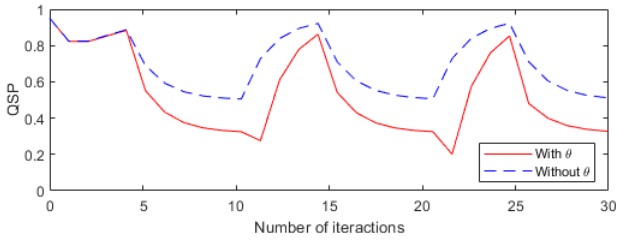


Fig. 7: Changes in quality of service provider (QSP) with θ and without θ in presence of on-off attack (OOA).

3) CBA: In Fig. 8, QSP of CBA attacker decreases progressively with the help of λ, which reduces the QSP value of nodes that behave differently with different nodes. In the simulation, we consider the CBA attacker behaves badly with 10% client nodes during its service provision. Table V illustrates various average feedback values of attacker node.

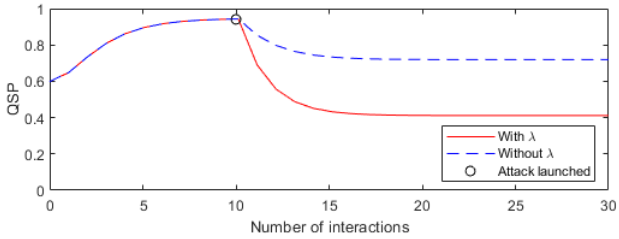


Fig. 8: Changes in quality of service provider (QSP) with λ and without λ in presence of conflicting behavior attack (CBA).

TABLE V: Feedback values for the attacker after the attack launched

Description	Value
Avg feedback from the attacked nodes	0.422
Avg feedback	0.767
Avg feedback without counting the attacked nodes	0.805

4) BMA/BSA: Both BMA and BSA belong to the URA, which leads a good service provider to be snubbed and a bad service provider to be promoted. To handle with them, comparing individual feedback with average level can determine the honesty of service raters. Figs. 9 and 10 demonstrate the changes in trust values in presence of BMA/BSA. The QSR evaluation enables to effectively detect unfair rating attacks (URA). As shown in Fig. 9, the attacked node's QSP recovers its trustworthiness since the attacker node has been isolated because of its trust score $TS < 0.5$. Analogously, malicious nodes' QSP drops after the isolation of the attacker node.

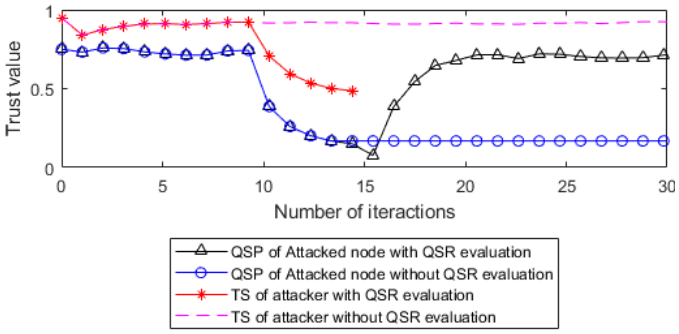


Fig. 9: Changes in trust values of both attacked and attacker nodes with quality of service rater (QSR) evaluation and without this evaluation in presence of bad mouthing attack (BMA).

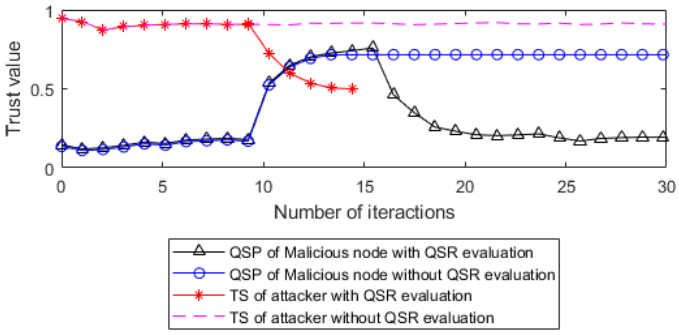


Fig. 10: Changes in trust values of both attacked and attacker nodes with quality of service rater (QSR) evaluation and without this evaluation in presence of ballot stuffing attack (BSA).

D. Inter-Manager TM

In Fig. 11 (a), the MCF shows managers (both evaluator and evaluated) have similar service types and capabilities. In the opposite way, managers in (b) are somehow not close in terms of conformity. This also causes cluster changes to be widely accepted in (a), but a few of them refused in (b). The MT value stays at a relatively high level above 0.5 in (a) and it is basically pasted into 0.5 in (b). Although managers do not provide service or rate service, they are capable of inclining other managers. For the evaluated managers with low MT, the evaluator manager can reduce the interaction frequency to economize on resources and energy.

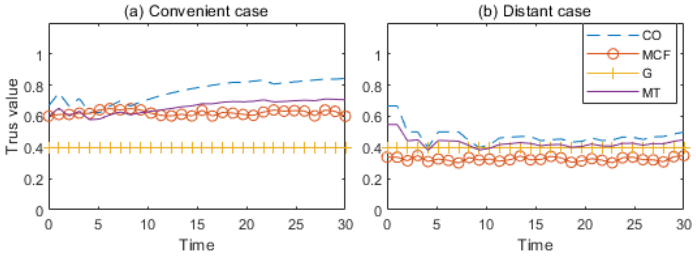


Fig. 11: Changes in values of cooperativeness (CO), centrality (G) , inter-manager conformity (MCF), and inter-manager trust score (MT) of convenient and distant cases.

VI. Conclusion

In this paper, we have presented a phase-based dynamic TM model named PDTM to address important issues in IoT systems. We have verified that the PDTM model is effective and accurate for dealing with initialization issue, service provider pre-selection, nodes classification, and importantly the countermeasures for several types of TRA, namely OOA, CBA, BMA, BSA, and NCA. As future works, we plan to move from simulation to implementation within real-world IoT devices that can act as both service provider and rater.

References

[1] A. R. Sfar, Y. Challal, P. Moyal, and E. Natalizio, "A game theoretic approach for privacy preserving model in iot-based transportation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 12, pp. 4405–4414, 2019.

[2] M. Blaze, J. Feigenbaum, and J. Lacy, "Decentralized trust management," in *Proceedings 1996 IEEE Symposium on Security and Privacy*, IEEE, 1996, pp. 164–173.

[3] R. K. Chahal, N. Kumar, and S. Batra, "Trust management in social internet of things: A taxonomy, open issues, and challenges," *Computer Communications*, vol. 150, pp. 13–46, 2020.

[4] A. Altaf, H. Abbas, F. Iqbal, and A. Derhab, "Trust models of internet of smart things: A survey, open issues, and future directions," *Journal of Network and Computer Applications*, vol. 137, pp. 93–111, 2019.

[5] Y. B. Saied, A. Olivereau, D. Zeghlache, and M. Laurent, "Trust management system design for the internet of things: A context-aware and multi-service approach," *Computers & Security*, vol. 39, pp. 351–365, 2013.

[6] D. Chen, G. Chang, D. Sun, J. Li, J. Jia, and X. Wang, "Trm-iot: A trust management model based on fuzzy reputation for internet of things," *Computer Science and Information Systems*, vol. 8, no. 4, pp. 1207–1228, 2011.

[7] M. Nitti, R. Girau, and L. Atzori, "Trustworthiness management in the social internet of things," *IEEE Transactions on knowledge and data engineering*, vol. 26, no. 5, pp. 1253–1266, 2013.

[8] U. Jayasinghe, A. Otebolaku, T.-W. Um, and G. M. Lee, "Data centric trust evaluation and prediction framework for iot," in *2017 ITU Kaleidoscope: Challenges for a Data-Driven Society (ITU K)*, IEEE, 2017, pp. 1–7.

[9] M. D. Alshehri, F. K. Hussain, and O. K. Hussain, "Clustering-driven intelligent trust management methodology for the internet of things (citm-iot)," *Mobile networks and applications*, vol. 23, no. 3, pp. 419–431, 2018.

[10] R. Du, C. Liu, and F. Liu, "Trust authorization monitoring model in iot," *International journal of performability engineering*, vol. 14, no. 3, p. 453, 2018.

[11] W. Fang, M. Xu, C. Zhu, W. Han, W. Zhang, and J. J. P. C. Rodrigues, "Fetms: Fast and efficient trust management scheme for information-centric networking in internet of things," *IEEE Access*, vol. 7, pp. 13 476–13 485, 2019.

[12] N. B. Truong, U. Jayasinghe, T.-W. Um, and G. M. Lee, "A survey on trust computation in the internet of things," *Information and Communications Magazine*, vol. 33, no. 2, pp. 10–27, 2016.

[13] L. Xu, R. Collier, and G. M. O'Hare, "A Survey of Clustering Techniques in WSNs and Consideration of the Challenges of Applying Such to 5G IoT Scenarios," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1229–1249, 2017.

[14] K.-Y. Lam and C.-H. Chi, "Identity in the internet-of-things (iot): New challenges and opportunities," in *International Conference on Information and Communications Security*, Springer, 2016, pp. 18–26.