Intelligent Automation & Soft Computing
DOI: 10.32604/iasc.2023.030091
Article

Tech Science Press

A Cross-Domain Trust Model of Smart City IoT Based on Self-Certification

Yao Wang[1], Yubo Wang[1], Zhenhu Ning[1,*], Sadaqat ur Rehman[2] and Muhammad Waqas[1,3]

[1]Faculty of Information Technology, Beijing University of Technology, Beijing, 100124, China
[2]Department of Natural and Computing Science, University of Aberdeen, UK
[3]School of Engineering, Edith Cowan University, Perth, 6027, Australia
*Corresponding Author: Zhenhu Ning. Email: nzh41034@163.com
Received: 18 March 2022; Accepted: 09 June 2022

Abstract: Smart city refers to the information system with Internet of things and cloud computing as the core technology and government management and industrial development as the core content, forming a large-scale, heterogeneous and dynamic distributed Internet of things environment between different Internet of things. There is a wide demand for cooperation between equipment and management institutions in the smart city. Therefore, it is necessary to establish a trust mechanism to promote cooperation, and based on this, prevent data disorder caused by the interaction between honest terminals and malicious terminals. However, most of the existing research on trust mechanism is divorced from the Internet of things environment, and does not consider the characteristics of limited computing and storage capacity and large differences of Internet of things devices, resulting in the fact that the research on abstract trust mechanism cannot be directly applied to the Internet of things; On the other hand, various threats to the Internet of things caused by security vulnerabilities such as collision attacks are not considered. Aiming at the security problems of cross domain trusted authentication of Intelligent City Internet of things terminals, a cross domain trust model (CDTM) based on self-authentication is proposed. Unlike most trust models, this model uses self-certified trust. The cross-domain process of internet of things (IoT) terminal can quickly establish a trust relationship with the current domain by providing its trust certificate stored in the previous domain interaction. At the same time, in order to alleviate the collision attack and improve the accuracy of trust evaluation, the overall trust value is calculated by comprehensively considering the quantity weight, time attenuation weight and similarity weight. Finally, the simulation results show that CDTM has good anti collusion attack ability. The success rate of malicious interaction will not increase significantly. Compared with other models, the resource consumption of our proposed model is significantly reduced.

Keywords: Smart city; cross-domain; trust model; self-certification; trust evaluation

1 Introduction

The smart cities refer to an information system with the IoT, cloud computing as its core technologies, government management, and industrial development are the essential contents. It features IoT of urban

resources, full integration of information resources, continuous innovation of application resources, and the gradual construction of system resources. It comprises multiple relatively independent information service subsystems, forming a large-scale integrated application and information service system based on massive information and intelligent analysis, which can be perceived, controlled, assembled and disassembled. The main functions involve digital urban management, digital law enforcement, smart grid, smart home, smart transportation, smart environmental protection, smart medical care and smart agriculture. The IoT technology drives the gradual formation of smart cities. The application of IoT technology enables data resources of smart cities to be fully integrated, tapped and utilized. Due to the multi-domain distribution and frequent movement of IoT terminals in smart cities, the smart city applications like smart buildings, smart pipe networks, smart street lights, smart communities, and smart transportation often use various IoT terminals with limited resources at a small cost for information collection and interaction. These IoT terminals may conduct frequent cross-domain information interaction. It is of great necessities to do trust management for preventing the terminal from interacting with malicious terminals during cross-domain information interaction.

Several trust models [1–3] have been proposed recently. The trust information in these trust models is based on collecting trust recommendations from neighbor nodes. Certain algorithms are used to calculate the trustee's trust value then decide whether to interact with it or not. However, it is challenging to collect trust recommendations across domains in the case of multi-domain distribution and frequent smart city terminals, which also consume a lot of time and bandwidth. In addition, some cross-domain trust models [4] have been proposed. Still, these models require the support of super nodes or trusted third parties, so they cannot meet the cross-domain trust evaluation of smart city IoT terminals under resource-constrained conditions. In this context, to ensure that the terminal can quickly and safely establish a trust relationship with the current domain in the cross-domain process and get rid of dependence on super nodes or trusted third parties, there is an urgent need to solve the cross-domain trust problem of IoT terminals when terminal resources are limited. To this end, we propose a cross-domain trust model of smart city IoT based on self-certification (CDTM). The CDTM model is based on the terminal self-certification mechanism [5]. The main idea is that after each information interaction of the terminal in the previous domain is completed, the two interacting parties will perform a trust evaluation based on the other's relevant behavior in the interaction process and generate a trust certification to the other party. After receiving the trust certification, the terminal will store it locally. Then, it will actively provide the trust certification to the other party to prove its credibility in the case of conduction of cross-domain information interaction. Compared with existing research, the advantages of the CDTM model proposed in this paper can be elaborated as follows.

(1) Based on the characteristics of self-certification, the terminal can actively collect and provide trust certification. Therefore, the trust information of the terminal in the previous domain can cross domains as the terminal moves, which solves the problem of difficulty in collecting trust recommendations during the cross-domain process. The CDTM model can quickly establish a trust relationship between a cross-domain terminal and the current domain.
(2) The trust certification is generated by fine-grained trust evaluation, which includes the remaining energy of the terminal in the interaction process, the number of data packets sent, the repetition rate of the data packets, and the transmission delay. Therefore, different weights can be assigned to various aspects according to different situations. In addition, the inclusion of a digital signature in the trust certification can prevent malicious terminals from tampering with the trust certification to improve its trust value.
(3) To alleviate collision attacks and improve the accuracy of trust evaluation, the comprehensive trust value is calculated by comprehensively considering the three factors of quantity weight, time decay weight and similarity weight. As a result, the model has a good anti-collision ability.

(4) Comprehensive simulations and performance analysis of the CDTM model is performed in a cross-domain situation. The results show that the CDTM model has strong resistance to collusion attacks. It can significantly improve trusted terminals' interaction success rate without increasing the interaction success rate of malicious terminals. At the same time, the resource consumption of CDTM is reduced compared with other models.

2 Related Work

Since 1996, H. Zheng [6] proposed trust management and proposed the trust management systems. The authors of the trust management model clearly defined trust based on the probability distribution method to predict the expected trust. Furthermore, the derivation of the credibility derived from the empirical recommendation were given. However, when calculating direct trust and third-party trust in this model, only a simple arithmetic average and a single credible value were used, which was too rough to effectively prevent various attacks by malicious entities in the network against the trust system itself. Saraereh et al. [7] combined Bayesian networks with entropy theory (ET) and offered a lightweight trust model for wireless sensor networks. The authors suggested that the weights of different trust aspects were derived from ET rather than setting manually. Yan et al. [8] designed a probability-based trust propagation model (HABIT), which optimizes the trust propagation path by considering the similarity between nodes. Based on the analysis of the attenuation coefficient in the trust model on the accuracy of trust calculation, combined with Bayesian theory, improved the accuracy of the trust calculation. Sathish et al. [9] proposed an intelligent beta reputation and dynamic trust evaluation model, which introduced proxy nodes to reduce the energy consumption of trust computing. A trust model based on friendship for the secure routing of mobile ad hoc networks was proposed by Shabut et al. [10]. The model combined direct trust and indirect trust to derive trust values and considered the decay of friendship over time. Benkerrou et al. [11] modelled node trust through credit value and proposed an IoT trust evaluation method based on trust and credit. Abderrahim et al. [12] clustered the nodes with community interest and proposed a trust evaluation model based on community interest. Zhu et al. [13] presented a trust evaluation model combining fuzzy sets and evidence theory. This model took the fuzzy membership function as the basic confidence function in evidence theory and fused the direct trust and indirect trust values according to the Dempster combination rule to optimize the adaptability and robustness of the model. Hazra et al. [14] proposed a trust model based on Bayesian networks for P2P file-sharing applications. The model calculated the trust value of nodes with direct trust and recommended trust. The direct trust was calculated by considering comprehensively multiple trust aspects. The trustors of the recommended trust were collected from the neighbor nodes of the trustee. Subsequently, Wu et al. [15] considered the time window and improved the scheme of [16] which could detect malicious entities.

Although the above models provided many trust evaluation methods, they did not consider the issue of cross-domain trust. Han et al. [17] introduced the gateway technology of the IoT trust model, designed a cross-domain node resource scheduling transaction table, and introduced evaluation indicators, such as trust factors. It also combined the context environment to calculate the trust degree of nodes in the domain. Su et al. [18] presented a community trust model based on topologically weighted (TPCT), in which trust information supports cross-domain. Huynh et al. [19] presented a trust model based on trust to aim at the dynamic changes of wireless sensor networks. The direct trust degree is settled through historical interaction records, and the indirect trust degree is calculated by assigning weight to the recommended trust between nodes. However, the cross-domain trust models [20–23] rely on super nodes or trusted third parties for trust evaluation, and nodes consume a lot of resources to obtain recommended trust degrees across domains. It cannot solve the cross-domain trust problem of IoT terminals in smart cities under resource constraints. Thus, the inherent characteristics of IoT terminals in smart cities are multi-domain distribution, diverse types, limited resources, and frequent movement. Therefore, the trust

model for cross-domain information interaction between IoT terminals needs to consider by establishing a trust relationship with the current domain when resources are limited while ensuring security.

3 Cross-Domain Trust Model Based on Self-Certifications

In a smart city IoT environment, terminals are distributed in multiple domains and move frequently. When a terminal accumulates a certain amount of trust in the previous domain, leaves the previous domain and joins the current domain, it is difficult for the terminal in the current domain to collect trust recommendations across different domains. It also takes a lot of time and bandwidth. In this case, the cross-domain terminal needs to re-establish the trust relationship with the current domain. It is unreasonable to ignore the trust information of the cross-domain terminal in the previous domain. Therefore, this paper proposes the CDTM model based on the self-certification mechanism. The cross-domain terminal actively provides the trust certifications stored in the previous domain interaction to prove its credibility. The model cross-domain trust evaluation diagram is shown in Fig. 1.

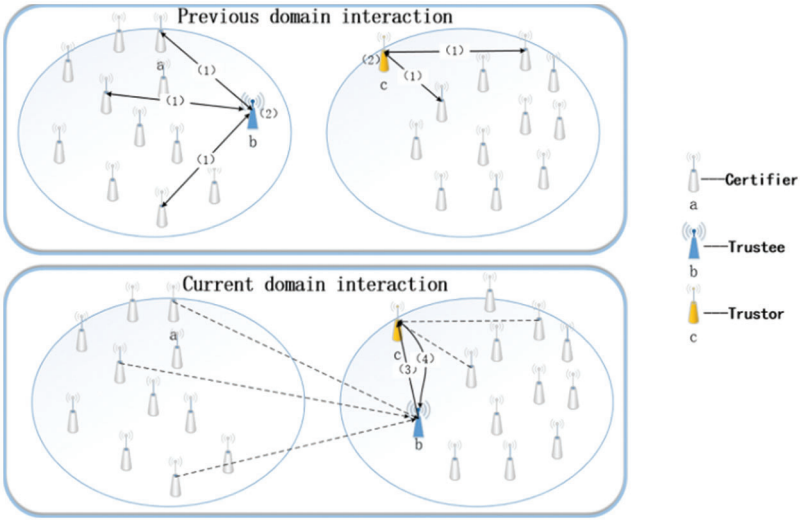

Figure 1: The CDTM model cross-domain trust evaluation diagram

In the CDTM model, the cross-domain terminal *b* is defined as the trustee, the terminal *c* in the current domain is defined as the trustor, and the terminal *a* interacting with the trustee in the previous domain is defined as the certifier. The overall process of cross-domain trust evaluation is as follows.

(1) The trustee *b* interacts with the certifier *a* in the previous domain. After the interaction is completed, the two parties conduct a trust evaluation on the other party's relevant behaviors in the information interaction process, and generate a trusted certificate and send it to the other party.

(2) After the trustee *b* receives the trust certification sent by the certifier *a*, it can use the relevant digital signature technology [22–24] to verify the trust certification's authenticity when necessary and choose the most beneficial *N* trust certification for storage and update.

(3) Trustee *b* moves across domains to the current domain and actively provides *N* trust certifications stored by itself to trustor *c*. Trustor *c* comprehensively considers the quantity weight, time decay weight and similarity weight to calculate the trustee's comprehensive trust value.

(4) If the trustee's total trust value reaches the trustor's trust threshold, the trustor *c* considers the trustee credible and agrees to interact with it. After the interaction is completed, the two parties will generate

a trust certification according to the other's appropriate behavior during the interaction and send it to each other. Otherwise, the trustor c thinks that the trustee b is not trustworthy. If the trustee b wants to interact with the trustor c, it must provide a more favorable trust certification.

The trustee actively stores and provides trust certifications. Hence, the previous domain's trust information can move along with the trustee across domains. Besides, the trust certification contains a digital signature, so the trustee cannot tamper with the trust certification to increase its trust value. Trustors only verify the digital signature or partial verification when needed. Compared with collecting trust recommendations across domains, the difficulty of verifying the digital signature is undoubtedly much more minor.

The CDTM model focuses on the situation where there are no public terminals between the two domains. If there are public terminals between the two domains, many classic trust models [25–27] can pass trust through these public terminals to solve the cross-domain problem. However, they cannot solve the cross-domain trust problem without public terminals.

3.1 Trust Certification Generation

In the CDTM model, after the trustee b interacts with the certifier a in the previous domain, the certifier a will send a trust certification to the trustee b after the successful interaction. The trust certification is expressed as a five-tuple as follows.

$$PoT(a,\ b) = (id(a),\ id(b),\ Tv(a,\ b),\ t_a,\ S_{ab}) \tag{1}$$

In (1), $id(a)$ and $id(b)$, respectively represent the unique identification (such as hardware address) of the certifier a and the trustee b; t_a represents the time stamp generated by the trust certification; S_{ab} represents the digital signature information; $Tv(a,\ b)$ is the total trust evaluation value of the certifier a to the trustee b during the interaction process, which can be expressed as

$$Tv(a,\ b) = (Tv(a,\ b,\ energy),\ Tv(a,\ b,\ send),\ Tv(a,\ b,\ repetition),\ Tv(a,\ b,\ delay)) \tag{2}$$

Among them, $Tv(a,\ b,\ energy)$, $Tv(a,\ b,\ send)$, $Tv(a,\ b,\ repetition)$ and $Tv(a,\ b,\ delay)$, respectively represent the remaining energy of the trustee b in the interaction process, the number of data packets sent, the data packets' repetition rate, and the transmission delay. The certifier a calculates the trustee's trust evaluation value through these four aspects of information. The specific calculation process is as follows.

(1) Definition E_init and E_remain respectively represent the initial energy value of the trustee b and the current remaining energy. The remaining energy of the terminal will affect the ability of data transmission. In the interaction process, the certifier a analyzes the trustee's data transmission activity b to obtain the percentage of the remaining energy E_init and the trustee's initial energy b. The more significant the proportion of remaining energy, the higher its credibility. The trust evaluation value of the certifier a to the trustee b in terms of the remaining energy is expressed as:

$$Tv(a,\ b,\ energy) = \frac{E\_remain}{E\_init} \tag{3}$$

(2) During the interaction, the certifier a requests n data packets from the trustee b, and b sends $m(m \leq n)$ data packets. Hence, the amount of data packets sent is one of the crucial indicators for calculating the trust value. When the value of m is closer to n, the trust evaluation value of the certifier a to the trustee b is higher; When m is much smaller than n, the trustee b may be a malicious terminal that deliberately intercepts information, and the certifier's trust evaluation of the trustee's data packet sending behavior drops sharply. Therefore, the trust evaluation value

$Tv(a,\ b,\ send)$ of the certifier a to the trustee b in terms of data packet sending behavior is proportional to m/n, which can be represented by a logarithmic function:

$$Tv(a,\ b,\ send) = \log_2(1 + \frac{m}{n}) \tag{4}$$

(3) During the interaction, the trustee's repetition rate of data packets b can effectively determine whether the trustee b intentionally sends duplicate data packets or insert wrong data packets. This is one of the essential indicators to measure data transmission service quality and determine trust value. The lower the data packet's repetition rate R sent by the trustee b, the higher the terminal's trust evaluation value in terms of the data packet's repetition rate behavior. $Tv(a,\ b,\ repetition)$ decreases as R increases. When the repetition rate R reaches or exceeds the critical value $\eta = 0.3$ [28], the possibility that the terminal is negative is very high. The change in the trust evaluation value of the certifier a to the trustee b in the repetition rate behavior of sending data packets conforms to the change of the exponential function, which can be expressed as:

$$Tv(a,\ b,\ repetition) = \begin{cases} 2 - \alpha^R, & if\ R < \eta \\ 0, & else \end{cases} \tag{5}$$

where $\alpha^\eta = 2$.

(4) During the interaction, due to factors such as network fluctuations or signal interference, data transmission will cause transmission delays. However, the transmission delay must fluctuate within a tolerable range, so it is necessary to set a critical value of the transmission delay to calculate the trust evaluation value based on the transmission delay. The critical value cannot be too large; otherwise, the terminal may have been maliciously attacked and controlled but has not been detected; it cannot be too small; otherwise, it may result in being judged as a malicious node before completing data transmission. When the trustee b transmits data packets to the certifier a, the certifier completely trusts the trustee if the transmission delay is less than the critical value b. On the other hand, as the transmission delay d exceeds the critical value ∂, the more likely the trustee b is a malicious terminal. The trust evaluation value of its transmission delay also drops rapidly. Therefore, the trust evaluation value of the certifier a to the trustee b in terms of transmission delay behavior can be expressed as:

$$Tv(a,\ b,\ delay) = \begin{cases} \chi^{\frac{d-\partial}{\partial}}, & if\ d \geq \partial \\ 1, & else \end{cases} \tag{6}$$

where $\chi = 0.01,\ \eta = 10$.

(5) The calculation method of the trust evaluation value $Tv(a,\ b)$ of the certifier a to the trustee b through the above four aspects is as follows.

$$Tv(a, b) = \lfloor \mu_e Tv(a, b, energy) + \mu_s Tv(a, b, send) + \mu_r Tv(a, b, repetition) + \mu_d Tv(a, b, delay) \rfloor \tag{7}$$

where $0 \leq \mu_e, \mu_s, \mu_r, \mu_d \leq 100$, and $\mu_e + \mu_s + \mu_r + \mu_d = 100$, which respectively represents the weighting coefficients of the four aspects of trust values. Each weight coefficient can be determined according to the requirements of specific application scenarios on terminal behavior. It can be seen that the trust evaluation values of the above four aspects are all between [0, 1], and the final trust evaluation value $Tv(a, b)$ is calculated to be in the range of [0, 100]. $Tv(a, b)$ is an integer instead of a floating-point number, effectively reducing the space for storing trust certificates and reducing the transmission energy consumption when providing trust certifications.

3.2 Store and Update N Trust Certifications

When the trustee b successfully interacts with the certifier a, the certifier a will calculate the trustee's trust evaluation value according to the four aspects described in (1) and generate a trust certification $PoT(a,\ b)$ to send to the trustee b. If the number of trust certifications stored locally by the trustee b is less than N, the trust certifications are stored locally. If N trust certifications have been stored locally, the trustee b will update the N trust certifications that are most beneficial to it. Due to the characteristics of trust, the more recent trust certification, the greater the influence. Therefore, when judging whether to update and store the trust certification, the trust evaluation value between different trust certifications is compared. Still, the influence of its trust evaluation value will decay over time. Here we introduce the time decay function:

$$ft(t_{now},\ t) = e^{-\lambda(t_{now}-t)} \tag{8}$$

The above formula t_{now} represents the current time, t represents the timestamp generated by the trust certification, and $\lambda \in (0,\ 1)$ is the weight coefficient, which can be dynamically adjusted. Therefore, it can be seen that $ft(t_{now},\ t)$ is a time function changing between (0, 1).

Therefore, the trustee b calculates the trust evaluation value of the trust certification based on the time factor:

$$Tv^{t}(a,\ b) = ft(t_{now},\ t) \cdot Tv(a,\ b) \tag{9}$$

According to each trust certification's trust evaluation value, the N trust certifications that are most beneficial to the storage and update are selected.

3.3 Three Weights of Trust Evaluation

When the trustee b wants to exchange information with the trustor c across domains, the trustee b actively sends the N trust certifications stored locally to the trustor c for trust evaluation. Since the trustee collects the trust certifications b, the trustee b may collude with other terminals to improve its trust value. Therefore, to mitigate this collusion attack and improve trust evaluation accuracy, this paper comprehensively considers the quantity weight, time decay weight, and similarity weight to calculate the total trust value.

3.3.1 Quantity Weight

To prevent two or more malicious terminals in the network from forming a collusive group to increase the trust value of the trustee deliberately b, trustor c requires the trustee b to provide enough trust certifications to improve the cost of the attack. However, the trustee's local storage resources b and the bandwidth consumption of multiple trust certifications transmissions must also be considered. According to the network state, the trustor c sets a system parameter to balance the terminal resource consumption and security issues, which indicates the number threshold. The number threshold δ is that the number of terminals participating in collusion within a time window Tw does not exceed $\lfloor(\delta - 1)/2\rfloor$. If it exceeds $\lfloor(\delta - 1)/2\rfloor$, the cost of collusion is too huge. At this time, the trustee b needs to provide $N = \delta$ trust certifications, and then the trustor c will consider the trust certifications provided by the trustee b to be reliable. Otherwise, the trust certifications provided by the trustee b are considered unreliable. So quantity weight function is defined as follows:

$$fn = \begin{cases} 0,\ if\ N < \delta \\ 1,\ else \end{cases} \tag{10}$$

3.3.2 Time Decay Weight

According to the trusting society's characteristics and the distributed IoT in smart cities, terminals are frequently cross-domain, and their computing environment may change rapidly. Therefore, in the comprehensive trust evaluation, the weight of the recently obtained trust certifications is greater, and some old ones may be completely unreliable. In a time window T_w, if the difference between the current time and the time generated by the trust certification exceeds T_w, the trust certification is considered entirely unreliable. Otherwise, the exponential time decay function weighs the trust certification. The time decay function is defined as follows:

$$ft(t_{now},\ t) = \begin{cases} 0 & t_{now} - t > T_w \\ e^{-\lambda(t_{now}-t)} & t_{now} - t \le T_w \end{cases} \tag{11}$$

Among them, t_{now} represents the current time, t represents the timestamp generated by the trust certification, and $\lambda \in (0,\ 1)$ is the weight coefficient, which can be dynamically adjusted. It can be seen that $ft(t_{now},\ t)$ is a time function changing between [0, 1). The weight of the trust certification generation time is different. The closer the time, the greater the weight; the farther, the smaller the weight.

3.3.3 Similarity Weight

When there are N trust certifications, it may bring opportunities for malicious terminals. A malicious terminal can maliciously increase the trust evaluation value of a specific terminal by sending a fake trust certification. Therefore, this paper takes the similarity between the trust evaluation value of the trust certification and the average evaluation trust value as the weight to reduce malicious terminals' influence. The closer to the average trust evaluation value, the greater the weight, and the farther the weight is, the greater the possibility of malicious defamation. First, calculate the average trust evaluation value of N trust certifications:

$$AVG(Tv) = \frac{1}{N}\sum_{i=1}^{N} Tv_i \tag{12}$$

The Euclidean space distance similarity discrimination method can calculate each trust certification's similarity weight. The specific calculation method is as follows:

$$fs_i = \frac{Tv_i}{Tv_i + \sqrt{\sum_{i=1}^{N}(AVG(Tv) - Tv_i)^2}} \tag{13}$$

3.4 Mutual Interaction

The trustor c calculates the trustee's total trust value according to the N trust certifications provided by the trustee b and the three weights described in 3.3. The calculation method is as follows:

$$T = \frac{fn \cdot \sum_{i=1}^{N} ft(t_{now},\ t_i) \cdot fs_i \cdot Tv_i}{N} \tag{14}$$

Each terminal will set a trust threshold Th according to its security level. If the trustee's total trust value reaches the trustor's trust threshold, the trustee c trusts the trustee b and agrees to interact with it. After the interaction is completed, both parties will generate a trust certification based on the other party's appropriate behavior during the interaction and send it to each other. Otherwise, the trustor c thinks that the trustee b is not trustworthy, and if the trustee b wants to interact with the trustor c, it needs to provide a more favorable

trust certification. From the subsequent interaction, trustor c and trustee b are also certifiers. Since the newly added terminal has not yet interacted with other terminals, the above method's total trust value is 0. Therefore, we set an initial default low trust value e for each terminal so that the newly added terminal can have the opportunity to interact with other terminals to obtain the trust certification and accumulate the trust value.

4 Simulation Experiment and Analysis

4.1 Experimental Environment and Settings

This paper uses the J2EE simulation environment to illustrate the performance of the CDTM model. Five disjoint trust domains 1 to 5 are deployed in the simulation. Each trust domain has 100 terminals, and their trust threshold Th is randomly generated between (0, 100). Set up three different terminals as trustees to move sequentially across domains, i.e., trusted, general, and malicious terminals. Trusted terminals provide high-quality services during the interaction process, general terminals provide high-quality and poor-quality services randomly, and malicious terminals provide poor-quality services. Simulation 1 verifies the correctness and effectiveness of the CDTM model by comparing the changes in the total trust value and the average successful interaction rate of the three kinds of terminals in an trusted environment. Simulation 2 verifies the CDTM model's security by comparing the changes in the comprehensive trust value of the malicious terminal in the case of collusion attacks between the CDTM model and the CR (Certified Reputation model) model [29]. Finally, simulation 3 compares and analyzes the performance and resource consumption of the CDTM model and other classic trust models in an trusted environment.

Since the trustee has no initial trust certifications in the simulation, domain 1 is the previous domain where the trustee accumulates the initial trust value. When the trustee is in a particular trust domain, the trustee conducts an interactive test with 100 terminals in the trust domain. If the number N of trust certifications stored by the trustee cannot reach the number threshold δ, the total trust value of the trustee is the default initial low trust value ε. If the number threshold δ is reached, the total trust value is calculated according to the CDTM model. The more considerable calculated comprehensive trust value and the initial low trust value ε is the trustee's total trust value. The interactive test is successful if the trustee's total trust value reaches the trustor's trust threshold Th, the interactive test is successful. The trustee and the trustor interact and provide each other with a trust certification after the interaction is completed. Regardless of whether the interaction is successful, the timestamp is increased by one after each test. The relevant parameter settings in the simulation are shown in Tab. 1.

Table 1: The relevant parameter settings in the simulation

Parameter	Value
Number threshold δ	25
Time window T_w	150
The time unit λ that controls the decay speed in the time decay function	1/50
Terminal default initial low trust value ε	15

4.2 Simulation 1: Correctness and Validity Verification of the Model

This simulation is deployed in a trusted environment, where the trustees are evaluated according to the trustee's appropriate behavior during the interaction. The model's correctness and effectiveness are verified by comparing the average comprehensive trust value changes of three terminals in a trusted environment and

the average interaction success rate in each domain. To improve the simulation experiment's accuracy, the number of experiments runs for each terminal is 100 times, and the simulation experiment results are shown in Figs. 2 and 3.

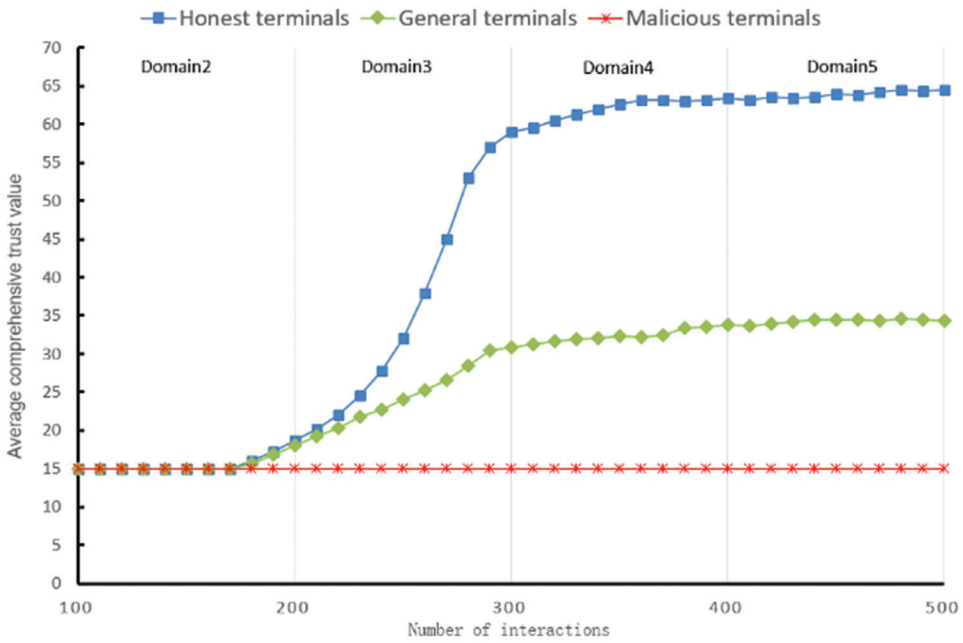

Figure 2: The average comprehensive trust value changes of three kinds of terminals in an trusted environment

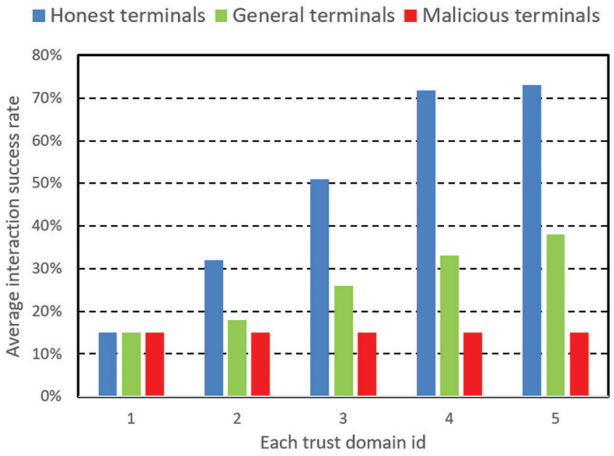

Figure 3: The average interaction success rate of the three kinds of terminals in each domain in an trusted environment

In Figs. 2 and 3, the trustee first accumulates and stores specific trust certificates in trust domain 1. Then, the trustee cross domains to trust domain 2 in order. At this time, the number of trust certifications of trustee has not reached the number threshold. As the number of successful interactions increases, the average comprehensive trust value of trusted terminals rapidly increases from the initial low trust value of 15 to a higher value of 63.2 and tends to stabilize; the average successful interaction rate also rapidly increased

to about 73% and tended to stabilize. The average comprehensive trust value of general terminals rises to a moderate value of 33.6 and tends to stabilize; the average interaction success rate increases to 38% and stabilizes. The malicious terminal's average comprehensive trust value is always the initial low trust value of 15, and the average interaction success rate remains unchanged at about 15%.

The simulation results show that the open terminal in the CDTM model can quickly increase its trust value to interact with other terminals in the cross-domain process in a trusted environment. Moreover, it will not improve malicious terminals' total trust value and interaction success rate. This shows that this model significantly enhances the interaction success rate of trusted terminals and does not increase the risk of interacting with malicious terminals, thus verifying this model's correctness and effectiveness.

4.3 Simulation 2: Security Verification of the Model

This simulation is deployed in a dishonest environment. In a dishonest environment, some malicious terminals in each domain provide favorable trust certifications for their malicious partners to improve their comprehensive trust value and provide unfavorable trust certifications for trusted trustees. The model's security is verified by comparing the CR model's anti-collision ability and the CDTM model in a dishonest environment. The CR model also uses self-certified trust, similar to the CDTM model. Therefore, this paper chooses the CR model to compare with the CDTM model. The CR model's trust value range is [−1, 1], which is different from the trust value range of [0, 100] in this article. To facilitate comparison, we increase the trust value range in the CR model to [0, 100]. Due to both the CR and CDTM models being based on a self-certification mechanism, the trustees will only store trust certifications that are beneficial to them. Therefore, it is only necessary to consider the situation in which malicious terminals increase their trust value through their colluding partners.

Since the CR model does not consider quantity weight, once there are colluding terminals, malicious terminals can provide trust certification that is beneficial to themselves through collusion so that their average comprehensive trust value will rise rapidly. On the other hand, in the CDTM model, the number of trust certifications is regarded as a substantial weight. Therefore, the trustor sets a system parameter δ according to the network status so that the number of terminals participating in the collusion within a time window T_w does not exceed $\lfloor(\delta - 1)/2\rfloor$. The simulation runs 100 times for each model, and the simulation results are shown in Fig. 4.

In Fig. 4, the solid line represents the change in the malicious terminal's average comprehensive trust value in the CR and CDTM models when the number of colluding terminals is 0–12 (that is, not more than $\lfloor(\delta - 1)/2\rfloor$). The dotted line represents the average comprehensive trust value of trusted terminals when stabilizing the two models. When there is a collusion terminal, the CR model's malicious terminal only needs to provide a false favorable trust certification provided by its collusion partner. In this case, the malicious terminal and the trusted terminal's average comprehensive trust value is very close, and other terminals cannot distinguish between them when performing trust evaluation. In the CDTM model, there is a large gap between malicious terminals and trusted terminals' average comprehensive trust value, and other terminals can effectively distinguish them when performing trust evaluation. The simulation results show that the CDTM model can resist collusion attacks more than the CR model.

4.4 Simulation 3: Comparative Analysis of Model Performance

This simulation compares the performance and resource consumption of the CDTM model, the BNTW (Bayesian Network with Time Window) model, and the CR model in a trusted environment. The BN (Bayesian Network) model is a classic trust model that uses direct trust and recommended trust to calculate a node's trust value. The direct trust is calculated by considering multiple trust aspects, and the trustor collects the recommended trust from the trustee's neighbor nodes. The BNTW model considers the

time window based on the BN model and has certain similarities with the CDTM model. Therefore, this paper chooses these two models to compare with the CDTM model. This simulation also needs to expand the trust range of the CR model and the BNTW model from [−1, 1], [0, 1] to [0, 100], respectively. This simulation compares the average trust value and average interaction success rate of the three terminals in each model. Each kind of terminal runs 100 times in each model. To save space, only the terminal cross-domain to domain 5 is given. The simulation results are shown in Figs. 5 and 6.

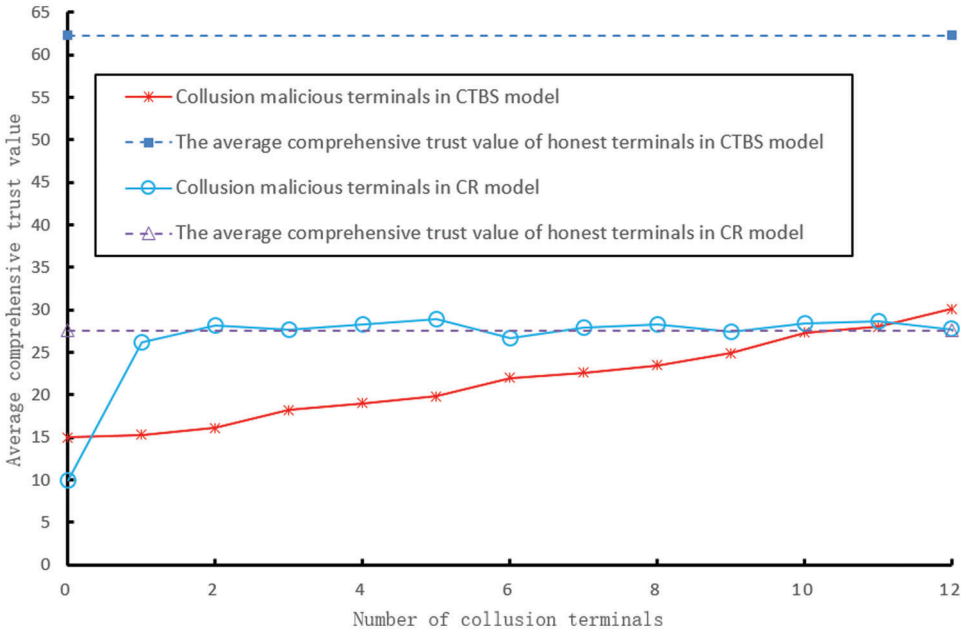

Figure 4: The changes in the average comprehensive trust value of malicious terminals in the two models with the number of colluding terminals

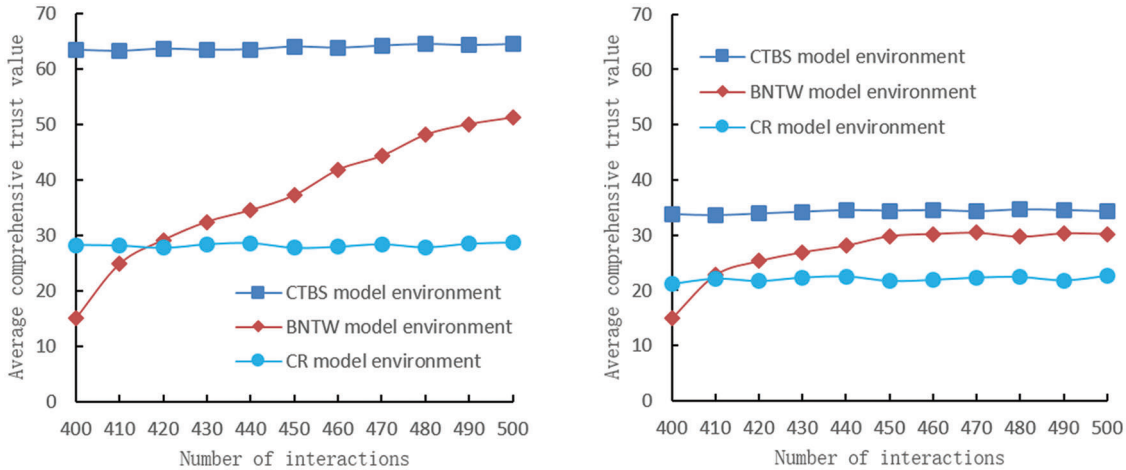

Figure 5: Comparison of changes in the average comprehensive trust value of trusted terminals (a) and general terminals (b) in domain 5 under different model environments

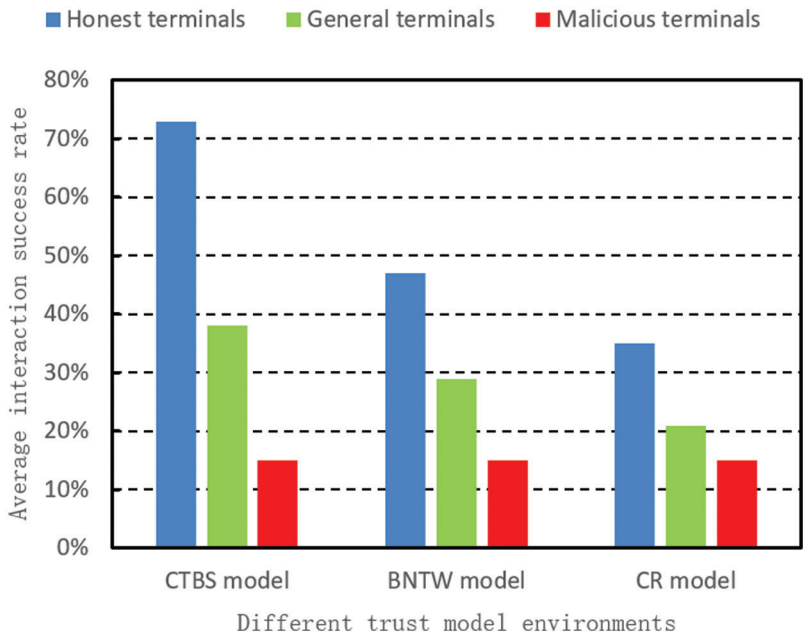

Figure 6: The average interaction success rate of the three kinds of terminals in each trust model environment

Fig. 5 compares the average comprehensive trust value of trusted and general terminals under different trust model environments. Because the malicious terminal's total trust value in each trust model is the default initial low trust value e, this part of the figure is omitted. Finally, Fig. 6 compares the average interaction success rate of the three kinds of terminals in different trust model environments.

Since both the CDTM and CR models adopt self-certified trust, they support the movement of trust information across domains. Therefore, the trusted terminal accumulated a certain amount of trust certifications information in the previous interaction between domains 1 and 4. Figs. 5a and 6 show that the trusted terminal maintains a relatively stable trust value in domain 5. However, there are some shortcomings in the CR model [30]. It calculates the similarity weight only by evaluating the distance of the value. When there is no previous interaction between the trustor and the trustee, the trustor has no trust evaluation, so the similarity weight is set to the low default value. As a result, the average comprehensive trust value of trusted terminals in the CR model is 28.4, far lower than 62.4 in the CDTM model. The CR model's average interaction success rate is about 35%, much lower than the 73% in the CDTM model. Because the BNTW model does not support cross-domain, the trusted terminal is equivalent to a newly added terminal when it moves to domain 5 in the BNTW model environment. As a result, the initial trust value is ε, and the trust value accumulation needs to be restarted. As the number of interactions increased, the average comprehensive trust value of trusted terminals rose rapidly from 15 to 51.2, and the average interaction success rate in domain 5 was about 47%. Figs. 5b and 6 show that general terminals' average comprehensive trust values in domain 5 in the CDTM model, BNTW model, and CR model environment are 34.2, 27.1, and 22.1, respectively. Moreover, the average interaction success rate is about 38%, 29% and 21%, respectively.

In the process of trust evaluation, both the collection and provision of trust information consume terminal resources. The number of interaction rounds between the trustor and the trustee is an essential indicator of terminal resource consumption. In the CDTM and CR models, the trustor receives the trust certifications provided by the trustee before the interaction and responds after the evaluation. If the trustor

agrees to exchange information after the interaction is completed, a trust certification is sent to the trustee, and a trust certification sent by the trustee is accepted. In the above process, the trustor needs at most 2 interaction rounds in a trust evaluation process, so the number of interaction rounds in each domain is $2 \times 100 = 200$. In the BNTW model, the trustor sends a recommendation trust request to 99 neighbors of the trustee and accepts the response. Therefore, the trustor needs 99 interaction rounds in a trust evaluation process, and the number of interaction rounds of the trustor in each domain is $99 \times 100 = 9900$.

The simulation results show that compared with the BNTW and CR models, the CDTM model significantly improves the average successful interaction rate of trusted terminals by approximately 55% and 109%, respectively. Simultaneously, the risk of interacting with malicious terminals is not increased, and the three models are about 15%. Furthermore, compared with the BNTW model, the CDTM model's consumption is significantly reduced in resource consumption to collect and provide trust information. However, the performance is comparable to the CR model. Therefore, in the environment of multi-domain distribution and frequent movement of IoT terminals in smart cities, when the terminal performs cross-domain information interaction, compared with the other two models, the CDTM model can quickly establish a trust relationship with the current domain with low resource consumption.

5 Conclusion

This paper proposes a cross-domain trust model based on self-certification for smart city IoT. Unlike most trust models, this model uses self-certified trust. In the previous interaction, the two parties who successfully interacted each time calculated the trust evaluation value based on the other party's remaining energy during the interaction, the sent number of data packets, the repetition rate of the data packets, and the transmission delay. Trustees actively provide their stored trust certifications to prove their credibility when performing cross-domain information interaction, enabling trust information to move across domains along with the terminal. After the trustor receives the trust certifications from the trustee, to alleviate collision attacks and improve the accuracy of trust evaluation, the total trust value is calculated by comprehensively considering the quantity weight, time decay weight and similarity weight. Finally, the simulation results show that the CDTM model can significantly increase the trusted terminals' success interaction rate without increasing the malicious terminals' success interaction rate. It has a solid ability to resist collusion attacks. At the same time, this model significantly reduces resource consumption compared with other models.

Acknowledgement: This paper was sponsored in part by Chongqing Industrial Control System Security Situational Awareness Platform, 2019 Industrial Internet Innovation and Development Project-Provincial Industrial Control System Security Situational Awareness Platform, Center for Research and Innovation in Software Engineering, School of Computer and Information Science (Southwest University, Chongqing 400175, China), and Chongqing Graduate Education Teaching Reform Research Project (yjg203032).

Funding Statement: This paper was sponsored in part by Beijing Postdoctoral Research Foundation (No. 2021-ZZ-077, No. 2020-YJ-006) and Chongqing Industrial Control System Security Situational Awareness Platform, 2019 Industrial Internet Innovation and Development Project-Provincial Industrial Control System Security Situational Awareness Platform, Center for Research and Innovation in Software Engineering, School of Computer and Information Science (Southwest University, Chongqing 400175, China), and Chongqing Graduate Education Teaching Reform Research Project (yjg203032).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] M. Adil, J. Ali, M. Khan, J. Kim, R. Alturki *et al.,* "An intelligent hybrid mutual authentication scheme for industrial internet of thing networks," *Computers, Materials & Continua*, vol. 68, no. 1, pp. 447–470, 2021.

[2] W. Sun, X. Chen, X. Zhang, G. Dai, P. Chang *et al.,* "A Multi-feature learning model with enhanced local attention for vehicle re-identification," *Computers, Materials & Continua*, vol. 69, no. 3, pp. 3549–3561, 2021.

[3] F. Zhu, Y. Ren, Q. Wang and J. Xia, "Preservation mechanism of network electronic records based on broadcast-storage network in urban construction," *Journal of New Media*, vol. 1, no. 1, pp. 27–34, 2019.

[4] T. Shanshan, M. Waqas, S. Rehman, T. Mir, Z. Halim *et al.,* "Social phenomena and fog computing networks: A novel perspective for future networks," *IEEE Transactions on Computational Social Systems*, vol. 9, no. 1, pp. 32–44, 2021.

[5] S. Tu, M. Waqas, S. Rehman, T. Mir, G. Abbas *et al.,* "Reinforcement learning assisted impersonation attack detection in device-to-device communications," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 2, pp. 1474–1479, 2021.

[6] H. Zheng and D. Shi, "A Multi-agent system for environmental monitoring using boolean networks and reinforcement learning," *Journal of Cyber Security*, vol. 2, no. 2, pp. 85–96, 2020.

[7] O. Saraereh and A. Ali, "Beamforming performance analysis of millimeter-wave 5G wireless networks," *Computers, Materials & Continua*, vol. 70, no. 3, pp. 5383–5397, 2022.

[8] Z. Yan, X. Li and M. Wang, "Flexible data access control based on trust and reputation in cloud computing," *IEEE Transactions on Cloud Computing*, vol. 5, no. 3, pp. 485–498, 2017.

[9] S. Sathish, A. Ayyasamy and M. Archana, "An intelligent beta reputation and dynamic trust model for secure communication in wireless network," *Industry Interactive Innovations in Science, (I3SET)*, pp. 395–402, 2017.

[10] A. Shabut, K. Dahal, I. Awan, "Friendship based trust model to secure routing protocols in mobile ad hoc networks," in *Int. Conf. on Future Internet of Things & Cloud*, IEEE, pp. 280–287, 2014.

[11] H. Benkerrou, S. Haddad and M. Omar, "Credit and honesty-based trust assessment for hierarchical collaborative IoT systems," in *Int. Conf. on Sciences of Electronics*, IEEE, 2017.

[12] O. Abderrahim, M. Elhdhili and L. Saidane, "TMCoI-SIOT: A trust management system based on communities of interest for the social internet of things," in *Wireless Communications & Mobile Computing Conf.*, IEEE, pp. 747–752, 2017.

[13] J. Zhu, "Wireless sensor network technology based on security trust evaluation model," *International Journal of Online Engineering*, vol. 4, no. 14, pp. 211, 2018.

[14] S. Hazra and S. Setua, "Probabilistic trust management in wireless communication system," in *Int. Conf. on Electrical & Computer Engineering*, IEEE, 2014.

[15] W. Wu, "Generating trusted graphs for trust evaluation in online social networks," *Future Generation Computer Systems*, vol. 31, pp. 48–58, 2014.

[16] F. Man, A. Ma and L. Zhang, "Inter-domain trust model based on gateway of IoT," *Computer Engineering & Design*, 2013.

[17] Q. Han, H. Wen and M. Ren, "A topological potential weighted community-based recommenlotion trust model for P2P networks," *Peer-to-Peer Networking and Applications*, vol. 8, no. 6, pp. 1048–1058, 2015.

[18] Y. Su, X. Gao and Y. Lu, "Credibility based WSN trust model," *Electronics Optics & Control*, 2018.

[19] T. Huynh, N. Jennings and N. Shadbolt, "Certified reputation: How an agent can trust a stranger," in *5th Int. Joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2006), DBLP*, pp. 1217–1224, 2006.

[20] M. Blaze, J. Feigenbaum and J. Lacy, "Decentralized trust management," in *IEEE Symp. on Security & Privacy. IEEE Computer Society*, pp. 164–173, 1996.

[21] R. Shankaran, V. Varadharajan and M. Orgun, "Context-aware trust management for peer-to-peer mobile ad-hoc networks," in *33rd Annual IEEE Int. Computer Software and Applications Conf.*, pp. 188–193, 2009.

[22] Z. Liu, J. Ma and Z. Jiang, "LCT: A lightweight cross-domain trust model for the mobile distributed environment," *Ksii Transactions on Internet & Information Systems*, vol. 10, no. 2, pp. 914–934, 2016.

[23] Y. Wang and J. Vassileva, "Bayesian network-based trust model," *Web Intelligence*, pp. 372–378, 2003.

[24] J. Dubey, "Bayesian network based trust model with time window for pure P2P computing system," in *IEEE Global Conf. on Wireless Computing and Networking (GCWCN), 2014*, IEEE, 2014.

[25] R. Gennaro, S. Goldfeder and A. Narayanan, "Threshold-optimal DSA/ECDSA signatures and an application to bitcoin wallet security," in *Int. Conf. on Applied Cryptography and Network Security*, vol. 9696, pp. 156–174, 2016.

[26] Y. Lindell, "Fast secure Two-party ECDSA signing," in *Annual Int. Cryptology Conf.*, vol. 10402, pp. 613–644, 2017.

[27] X. Yang, J. Wang and T. Ma, "A secure and efficient ID-based signature scheme with revocation for IOT deployment," in *2018 Sixth Int. Conf. on Advanced Cloud and Big Data (CBD)*, 2018.

[28] X. Li, J. Niu and S. Kumari, "A Three-factor anonymous authentication scheme for wireless sensor networks in internet of things environments," *Journal of Network and Computer Applications*, vol. 103, pp. 194–204, 2018.

[29] I. Kocher, "An experimental simulation of addressing auto-configuration issues for wireless sensor networks," *Computers, Materials & Continua*, vol. 71, no. 2, pp. 3821–3838, 2022.

[30] R. Dubey and J. Agrawal, "An improved genetic algorithm for automated convolutional neural network design," *Intelligent Automation & Soft Computing*, vol. 32, no. 2, pp. 747–763, 2022.