Research Article

Research on cloud computing service based on trust access control

Panjun Sun

Abstract

Recently, more and more network fraud incidents have damaged the interests of cloud service traders. To enhance mutual trust and win–win cooperation between the users and the cloud service provider, in the article, we construct a trust access control model for cloud services. First, we propose a trust evaluation method based on direct trust, trust risk, feedback trust, reward penalty, and obligation trust to express the complexity and uncertainty of trust relationship. Second, we propose trust evaluation and weight algorithm of trust factor by information entropy and maximum dispersion; therefore, our model has a better scientific and higher practical application value. Finally, we design related comparative experiments of three models to verify the efficiency, success rate, accuracy of trust evaluation, and privacy disclosure date, and these results show that our research performance is quite superior.

Keywords

Cloud service, access control, trust, information entropy, privacy

Date received: 21 August 2019; accepted: 12 November 2019

Introduction

With the rapid development of Internet and information technology, more and more users upload data in cloud computing. Cloud computing provides flexible and scalable services for individuals and organizations, which brings great convenience to users. However, complex architecture and data ownership bring great challenges and impacts on privacy and security in cloud computing.

Motivation

Traditional access control model relies on certificate authority and complex cryptography algorithm, but its application is severely limited in the distributed network environment of cloud computing. Trust evaluation has become an important method for malicious nodes detection, security assurance, and privacy protection in cloud computing.

The relation between privacy protection and trust is not a new issue, many solutions have been extensively used in data publishing, data searching, data mining, data aggregation, and other areas. In recent years, many relevant articles effectively promote the development of privacy trust research, but there are still some of the following problems.

- Many trust methods search feedback nodes through broadcast mode, which result in bandwidth overhead, further affect efficiency.
- Trust computing is a process of multiattribute decision-making, and many studies calculate the weight-based expert opinions, which is very difficult to dynamically adjudication for the system.

Corresponding author:

Panjun Sun, School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, 800 Dongchuan Road, Minhang District, Shanghai 200240, China.
Email: sunpanjun2008@163.com

Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (https://creativecommons.org/licenses/by/4.0/) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (https://us.sagepub.com/en-us/nam/open-access-at-sage).
Dynamics is a big challenge of trust evaluation, because the trust is a variable with time, many literature lack the description attenuation of trust, which affect the accuracy of prediction models.

Contributions
To deal with the above issues, the article proposes a model (Figure 1) for cloud computing service. The main contributions are as follows:

- We establish a trust evaluation model based on direct trust, trust risk, feedback trust, reward penalty, and obligation trust to describe the complexity and uncertainty of trust relationship more comprehensively.
- We propose weight method of trust decision factor by the information entropy and maximum dispersion, which surpasses the limitations of traditional weight methods for multiple attributes.
- We design several experiments to describe and analyze the performance evaluation for the trust model, which is examined from efficiency, success rate and satisfaction degree, accuracy and privacy disclosure rate; results certify that our research can effectively complete trust tasks and can protect privacy in the cloud environment.

The structure of this article is as follows. In the second section, we conclude some related work on trust, security, and privacy protection. In the third section, we present the related concept definitions of the model approach based on trust, obligation, privacy, permission, and other elements. In the fourth section, we establish a multiattribute trust weight model based on information entropy. In the fifth section, we design and discuss several experiments. In the sixth section, we summarize research and discuss future work.

Related work
Trust is a significant role in the cloud service that can verify the trust relationship between system entities and improve the security of the system. However, the collection and processing of trust evidence may lead to privacy disclosure, thus some related entities may be unwilling to provide related personal information for trust metrics.

Aluvalu and Muddana proposed an access control system based on obligation trust. Permission is determined by the role, and the trust degree of each user can be improved by the obligation. The popularity of composite services leads to important privacy and security issues. Rohit and Bharat designed an efficient solution to implement security policy in composite web services framework, which protected data privacy in the service life cycle. The solution enabled data owners to control data and reduced the risk of unauthorized access. Bhatia and Singh proposed a privacy-aware access control model, which included several privacy parameters, retention period, environmental conditions, and granularity level. Khaled and Zhu introduced a trust access control model (TB-AC) based on three factors (attribute, observation, and recommendation), the experimental results showed that TB-AC can evaluate access requests within acceptable processing time.

Both trust and risk become important research directions of privacy protection in cloud services. Junqi and Deyun constructed a fine-grained trust access control model, which combined risk and trust to allocate permission to...
reduce the privacy leakage. Further, Nogoorani and Jalili\(^8\) constructed a trust access control framework, the request of user can be permitted or denied by the access policies. In the framework, the site administrator can specify the user’s responsibility in the form of obligation. Matin and Nima\(^9\) proposed a method of opinion leader and control entity recognition based on reputation, input degree, and output degree, which eliminated the influence of troll entities in the cloud.

It is very difficult for service consumers to distinguish what is credible service or malicious service. Yan et al.\(^10\) proposed a formal policy specification language called P-SPEC, which can be used to describe the privacy policies of services and consumers’ privacy preferences. Furthermore, they proposed a privacy-aware service selection method, which consisted of a set of P-SPEC policies and introduced privacy metrics and privacy-sensitive specific policy matching algorithms, and implemented a prototype of concept to carry out relevant experimental research. Based on the customer feedback, global consultation feedback, and third-party feedback, Varalakshmi and Judgi\(^11\) proposed to evaluate trust service providers based on contextual feedback from different sources. In addition, unfair feedback is filtered to improve accuracy. Based on a set of evaluation indicators, Halabi and Bellaiche\(^12\) proposed a method to quantify and evaluate cloud security services, developed a paradigm using goal question measurement, and used a case to demonstrate the effectiveness and practicability.

Simeon and Iraklis\(^13\) proposed several context semantic representations of access control policies. More specifically, the approach accurately infused specific security and business requirements of these policies with respect to relevant knowledge and cultural relics and can make semantic inferences about policy compliance through prescribed structures. Service of encryption data becomes a challenging problem. Yan et al.\(^14\) proposed two schemes to protect the privacy of providers based on additive homomorphic encryption to support the traditional trust evaluation. The first scheme achieved better computational efficiency, and the second one provided greater security at the consumption of computational cost. Specifically, these algorithms can overcome the attacks of internal malicious evidence providers to a certain extent, even if trust evaluation is partially implemented in the form of encryption.

Both role-based access control (RBAC) and attribute-based access control are developed to protect privacy in the cloud. Lan and Vijay\(^15\) proposed a trust model to analyze and improve the security of data in the cloud storage systems by cryptographic RBAC schemes. The trust model provided a method to determine user’s credibility, which considered the role inheritance and hierarchy in the trust evaluation, and explained how to use trust assessment to reduce risks and improve the quality of decision-making. Xue and Xue\(^16\) explored a special attribute-based access control scenario in which the data owner allowed multiple users with different attribute sets to gain access rights. Further, by specifying translation nodes in access structure, a controlled cooperative access control scheme based on attributes is proposed. Smari and Clemente\(^17\) proposed an extended access control model based on topic-related attributes, which combined trust and privacy issues to make access control decisions sensitive to cross-organizational collaborative contexts.

### Access control based on multiple attributes

According to Figure 1, we express specific details and definitions in this section, and some important parameters are presented in Table 1.

#### Model architecture

In the section, we propose to construct an access control model, which includes trust, obligation, and risk mechanism. The following paragraphs will introduce some related concepts, and details are shown in Figure 2.

The model contains several basic elements: subject (user), object (resource), attribute, operation, obligation, trust, and policy. To effectively service, the access control system grants authorization to requester and then gives certain operation privileges.

- **Attribute**: it represents the characteristics of the node. Attr expresses an attribute, Attr = \{attr\(_1\), attr\(_2\), ..., attr\(_n\)\} denotes an attribute set.
- **Obligation**: it plays a constrained role in access control systems, which means that some operations need to

| Table 1. Meaning of some parameters. |
|--------------------------------------|
| **Parameter** | **Meanings** |
| D(S) = \{D\(_1\), D\(_2\), ..., D\(_n\)\} | \(Z\) entities of system |
| \(Y_1, Y_2, Y_3, Y_4\) | Trust function |
| \(\omega\) | Weight of trust function |
| TG(D\(_i\), D\(_j\), S, t) | Trust between D\(_i\) and D\(_j\) |
| RS = \{R\(_1\), R\(_2\), ..., R\(_n\)\} | N level trust degree |
| \(\psi(TG(D_i, D_j))\) | Trust decision |
| \(T_i\) | Decay time factor |
| R(D\(_i\), D\(_j\)) | Risk function |
| S = \{s\(_1\), s\(_2\), ..., s\(_n\)\} | Service level |
| Level | The level of a trust tree |
| \(\rho(F)\) | Feedback weight factor |
| F(D\(_i\)) = \{F\(_1\), F\(_2\), ..., F\(_n\)\} | Feedback entities of D\(_i\) |
| \(\eta \in [0, 1]\) | Quality factor |
| \(\chi \geq 1\) | Distance factor |
| \(c_i\) | Evaluation error at time \(t\) |
| \(E_p\) | Privacy disclosure of \(i\)th access |
| \(eq\) | Interaction access |
be performed after accessing objects. The obligation mechanism records the user operation process and allows the monitor to adjust the corresponding user rights in the policy, which can be expressed as
\[
\text{Obligation} = \{\text{subject, trigger, action}\}. 
\]
Under such an obligation mechanism, if a user finds a suspicious or malicious request in the record and notification table, the privacy settings can be updated in a timely manner to enhance privacy security.\(^{12}\)

Permission: it can operate on object, such as allowed, or rejected, \(\text{Perm} = \{\text{perm}_1, \text{perm}_2, \cdots, \text{perm}_n\}\).

Operation: it is the action that can be performed on the object, such as reading, writing, copying, editing, deleting, and so on, and the set of operation is described as follows: \(\text{OP} = \{\text{op}_1, \text{op}_2, \cdots, \text{op}_t\}\).

**Trust decision**

Let \(D_1, D_2, \cdots, D_Z \in D(S)\) express \(Z\) nodes in the system, according to the different roles, there are two categories: cloud service provider (CSP) and users. Suppose that the trust evaluation has many several factors \(Y_1(D_i, D_j), Y_2(D_i, D_j) \cdots Y_M(D_i, D_j)\) between the node \(D_i\) and \(D_j\) (\(D_j \in D(S)\)), the decision factors are \(Y = (Y_1, Y_2, \cdots Y_M)\), \(0 \leq Y_m(D_i, D_j) \leq 1\), \((m = 1, 2, \cdots, M)\). \(\omega_m\) is weight factor of \(Y_m(D_i, D_j)\), it satisfies the constraint condition:
\[
0 \leq \omega_m \leq 1, \quad \sum_{m=1}^{M} \omega_m = 1 \tag{1}
\]

\(TG(D_i, D_j, S, t)\) expresses the trust value between \(D_i\) and \(D_j\), as given in formula (2):
\[
TG(D_i, D_j, S, t) = \sum_{m=1}^{M} \omega_m Y_m(D_i, D_j) \tag{2}
\]
where \(S\) is provided by \(D_j\). \(TG(D_i, D_j, S, t)\) determines the quality of service, the higher the value of \(TG(D_i, D_j, S, t)\), the better the quality of service, and \(t\) is the time interval.

![Figure 2. The relationship between elements in access control.](image_url)
In a trust system, authorization is determined by the value of trust evaluation. Assume that $TG(D_i, D_j)$ has $K$ levels $R = \{R_1, R_2, \ldots, R_i, \ldots, R_K\}$, $0 \leq R_i \leq (i = 1, 2, \ldots, K)$. $R$ is an ordered space, assume that CSP can provide $S = \{s_1, s_2, \ldots, s_e\}$, the relation function $\psi(TG(D_i, D_j))$ between $S = \{s_1, s_2, \ldots, s_e\}$ and $TG(D_i, D_j)$ is defined as in formula (3):

$$
\psi(TG(D_i, D_j)) = \begin{cases} 
  s_p, & R_K < TG(D_i, D_j) \\ 
  s_{p-1}, & R_{K-1} \leq TG(D_i, D_j) < R_K \\
  \vdots & \vdots \\
  s_2, & R_1 \leq TG(D_i, D_j) < R_2 \\
  s_1, & 0 \leq TG(D_i, D_j) < R_1 
\end{cases}
$$

(3)

where $R = \{R_1, R_2, \ldots, R_i, \ldots, R_K\}$ is determined by the cloud service requirement, when $D_i$ requests service from $D_j$, authorization is dependent on the trust value. For example, a cloud service system provides three ranks of service, $S = \{s_1, s_2, s_3\}$, $s_1$ represents denial of service, $s_2$ represents the reading services, and $s_3$ represents both reading and writing services. The decision level space is $R = \{R_1, R_2, R_3\} = \{0, 0.3, 0.5\}$, the trust operation authorization is expressed as in formula (4):

$$
\psi(TG(D_i, D_j)) = \begin{cases} 
  s_3, & 0.5 < TG(D_i, D_j) \\ 
  s_2, & 0.3 \leq TG(D_i, D_j) \leq 0.5 \\
  s_1, & 0 \leq TG(D_i, D_j) < 0.3 
\end{cases}
$$

If trust degree of $D_j$ is $TG(D_i, D_j) = 0.2$, then the result is $\psi(TG(D_i, D_j)) = \psi(0.2) = s_1 = deny$.

Trust computing

To express the complexity and dynamics of trust, several factors, such as direct trust, trust risk, feedback trust, obligation trust, reward penalty, are introduced to describe the concept of trust.18

Direct trust

Direct trust usually consists of multiple factors, and related elements are chosen from the CSP record table based on interaction history.19

Weight calculation. To quantify the different roles of multiple indicators, these weights of multiple attributes are determined based on maximum entropy. There are $m$ users and $n$ attributes, evaluation score matrix $E(D)$ is expressed in formula (5), $e_{ij}$ is the evaluation value of $j$th attribute of the $i$th user:

$$
E(D) = \begin{bmatrix} 
  e_{11}, & e_{12}, & \ldots & e_{1n} \\
  e_{21}, & e_{22}, & \ldots & e_{2n} \\
  \vdots & \vdots & & \vdots \\
  e_{m1}, & e_{m2}, & \ldots & e_{mn} 
\end{bmatrix}
$$

(5)

Weight method of entropy: $E = (e_{ij})_{m \times n}$

$$
e_{ij} = -k \sum_{i=1}^{m} p_{ij} \ln p_{ij}, p_{ij} = e_{ij}/\sum_{i=1}^{m} e_{ij}, (i \leq j \leq n), k = 1/\ln m
$$

(6)

Weight of the $jth$ attribute:

$$
\hat{e}_j = (1 - e_{ij})/\sum_{j=1}^{n}(1 - e_{ij}), (1 \leq j \leq n), 0 \leq \hat{e}_j \leq 1, \sum_{j=1}^{n} \hat{e}_j = 1
$$

(7)

Decay time factor. Trust has timeliness, so the decay time factor is introduced to reflect trust more accurately. $t_i$ is the time slot of the $i$th successful transaction and the origin of the time, $t_0$ expresses the start time of the $i$th service, and $t_i^2$ represents the end time of the $i$th service. $t_0$ is the time slot of the origin time, $n$ is the amount of successful service, and the decay factor $T^{(i)}$ is given in formula (8):

$$
T^{(i)} = \frac{1}{2} \left[ \frac{t_i - t_0}{\sum_{j=1}^{n} (t_j - t_0)} + \frac{t_i^2 - t_1^2}{\sum_{j=1}^{n} (t_j^2 - t_1^2)} \right], \text{ and } \sum_{i=1}^{n} T^{(i)} = 1
$$

(8)

Compute direct trust. The comprehensive of formulas (6) to (8) can be used to evaluate the direct $T(D_i, D_j)$ from $D_i$ to $D_j$, $n$ is the amount of interaction times, as given in formula (9):

$$
Y_1(D_i, D_j) = \sum_{j=1}^{n} e_{ij}\hat{e}_j T^{(i)}
$$

(9)

Feedback trust

Feedback trust is the expected quantification of the transmission content of the node, such as $D_i$ trusts $D_j$, $D_j$ trusts $D_k$, so, $D_i$ trusts $D_k$ There are many possibilities in the trust network, how to select, aggregate, and calculate trust paths is a problem. Assume that $D_i$ is the parent node, all the nodes are children of $D_i$, neighbors also have neighbor nodes, so we construct a multilayer weighted digraph (MWD, a sample in Figure 3). $D(S)$ is a set of nodes, $C$ represents the trust value from grand node to the subnode, and $Y_i$ is the trust value. In the MWD, if the layer of root node is level 0, the layer of the direct neighbor is level 1, the level of neighbor’s neighbor is level 2, and so on

$$\{F_1, F_2, \cdots, F_l\} \text{ is set of feedback node, } F_k \text{ is a feedback node, and the feedback trust (such as in Figure 3) function}$$
multilayer weighted digraph. Figure 3. An MWD of computation feedback trust. MWD: multilayer weighted digraph.

is expressed as formula (10):

$$Y_2(D_i, D_j) = \left\{ \begin{array}{ll} \frac{\sum_{k=1}^{l}(\rho(F_k) \times Y_1(F_k, D_j))}{\sum_{k=1}^{l} \rho(F_k)} & I \neq 0 \\ 0 & I = 0 \end{array} \right. \tag{10}$$

where $l$ is the number of feedback nodes, $\rho(F_k)$ is a trust feedback factor, and according to the “six degrees of separation” and each level of feedback node, it is expressed as formula (11):

$$\rho(F_k) = \left\{ \begin{array}{ll} 1 & \text{level} = 0 \\ \prod_{m=0}^{k} Y_1(D_m, D_n) & \text{level} \geq 1 \end{array} \right. \tag{11}$$

where $Y_1(D_m, D_n)$ expresses the direct trust value between $D_m$ and $D_n$, and level represents the layer of the feedback trust node. In Figure 3, level $1 = 1$, $\rho(D_1) = 0.5$; level $2 = 2$, $\rho(D_2) = 0.5 \times 0.8 = 0.40$; and level $3 = 3$, $\rho(D_3) = 0.8 \times 0.7 \times 0.5 = 0.28$. Assume that the node $D_0$ needs the trust value of $D_{10}$, and there are two nodes $D_2$ and $D_6$ interacting with $D_{10}$, the value is $Y_1(D_2, D_{10}) = 0.4$, $Y_1(D_6, D_{10}) = 0.5$. According to formula (10), $\rho(D_2) = 0.4 \times 0.6 = 0.24$, $\rho(D_6) = 0.5 \times 0.6 = 0.30$, and $Y_2(D_2, D_{10}) = (0.24 \times 0.4 + 0.3 \times 0.5)/(0.24 + 0.3) \approx 0.455$.

Algorithm 1. Feedback node search.

```
Input $l$, level, $\eta, Z$.
Output $F(D_i)$.
1 $N(D_i)$ is neighbor node set of $D_i$;
2 $F(D_i) = \{F_1, F_2, \cdots, F_l\}$ is feedback node set of $D_i$;
3 if level($D_i, D_j$) > $\eta$ then end;
4 for (all $D_k \in N(D_i)$) do
5 if $Y_1(D_i, D_k) > \eta$ then
6 query $Y_1(D_i, D_k)$ in the database of $D_k$;
7 if $Y_1(D_i, D_k)$ exists then
8 $F(D_i) = F(D_i) + D_k$;
9 request feedback nodes $(D_k, D_i, \eta, \gamma)$;
10 return $F(D_i)$;
11 end.
```

section introduces distance factor and quality factor to adjust feedback trust.

The quality factor $\eta \in [0, 1]$ is a normal number, when the trust value of feedback node is $Y_1(D_i, D_j) \geq \eta$, the feedback information is credible; otherwise, $Y_1(D_i, D_j) < \eta$, the feedback node is incredible. The quality factor can enhance the convergence of the system by controlling the aggregation scale of feedback trust.

We define the distance factor $\gamma \geq 1$ as a normal number to limit the propagation range of feedback trust. When level($D_i, D_j$) $\leq \gamma$, the node will transfer the data to the adjacent node; otherwise, it will stop transmission. According to formulas (10) and (11), we can aggregate and select the trust of $k$ feedback entities, propose Algorithm 1 by combining feedback trust and distance factor, and level($D_i, D_j$) represents the distance between $D_i$ and $D_j$ in the MWD.

**Obligation trust**

According to obligation description, it is generally necessary to introduce the obligation concept in the access control system. Because a manager must perform obligation when a request can execute, the obligation trust function of node $D_i$ can be expressed as formula (12):

$$Y_3(D_i, D_j) = \sum_{b \in B} \gamma \times GB + \sum_{b \in B} \gamma \times OB \tag{12}$$

Obligation has an important impact on users’ trust and help to protect privacy in the cyberspace. In formula (12), $\gamma$ represents the obligation weight; $b$ represents the number of obligations that return the system; $OB$ represents the number of obligations that $D_i$ has not been completed during the certain time; $GB$ represents the number of obligations of $D_i$ in the certain time; and $B$ is the number of obligations of $D_i$ in the cloud system.
**Trust risk**

Although the literature also introduced the risk mechanism, the system did not consider the relationship between risk and quality of service. According to the principle of economics and perspective of service, risk function is expressed by formula (13):

$$R(D_i, D_j) = \psi(TG(D_i, D_j, S, t)) \times [1 - TG(D_i, D_j, S, t)]$$  \hspace{1cm} (13)

where $\psi$ represents the service quality of $D_i$, according to experience, the trust value is higher, the risk is smaller. $TG(D_i, D_j, S, t)$ shows the trust evaluation of the latest time, it is a positive proportional relationship between $R(D_i, D_j)$ and $1 - TG(D_i, D_j, S, t)$.

Trust risk function is mainly used to measure the potential unsafe between CSP and users, as expressed in formula (14):

$$Y_4(D_i, D_j) = 1 - R(D_i, D_j)$$  \hspace{1cm} (14)

According to formulas (13) and (14), the risk is related to the importance of service, the better service means greater risk, which is an inverse proportion between $Y_4(D_i, D_j)$ and $R(D_i, D_j)$.

**Reward penalty**

In the trust evaluation, we should reward honest behavior and penalize malicious behaviors. So, the reward penalty function can be expressed by formula (15):

$$Y_5(D_i, D_j) = 1 - \frac{\sum F(D_i, D_j)}{H}$$  \hspace{1cm} (15)

where $\sum F(D_i, D_j)$ is the number of failure times and $H$ is the number of service times. Because malicious nodes often intentionally destroy or provide false services, the service failure rate is very high, and the reward penalty function can penalize malicious behaviors.

**Weight of the trust attribute**

In the process of trust quantification, the effects of different attributes are different, so we propose a weight method. Let $W = (\omega_1, \omega_2, \ldots, \omega_m)$ expresses the different weight of the trust attribute, so, we get $Orness(W) = \sum_{i=1}^{m} \frac{(m-i)\omega_i}{m-1}$, where $\omega_i$ is the weight of the $i$-th trust attribute.

**Algorithm 2. Weight of the trust attribute.**

1. if $0 < m \leq 2$
2. then $\omega_1 = a$.
3. $\omega_2 = 1 - a$.
4. if $m > 2$
5. $\omega_1 \left[ \frac{m-1}{m} \alpha + 1 - m\omega_1 \right] = \left[ \frac{m-1}{m} \alpha - m \right] \omega_1 + 1$.
6. $\omega_m = \frac{m-1}{m} \alpha - m$.
7. for $i = 2$ to $m - 1$ do
8. $\omega_i = \alpha^{m-i} \omega_m$.
9. when $\omega_1 = \omega_2 = \cdots = \omega_m = \frac{1}{m}$
10. then $disp(W) = \ln m, a = 0.5$.
11. end.

and maximum dispersion degree $Disp(W) = - \sum \omega_i \ln \omega_i$.

Further deduction, $0 \leq Disp(W) \leq \ln m$, which satisfies three conditions:

$$\text{maximize} : - \sum_{i=1}^{m} \omega_i \ln \omega_i$$  \hspace{1cm} (16)

$$Orness(W) = \alpha, \alpha \in [0, 1]$$  \hspace{1cm} (17)

$$\sum_{i=1}^{m} \omega_i = 1, \omega_i \in [0, 1], i = 1, 2, \ldots, m$$  \hspace{1cm} (18)

From formulas (16) to (18) and the maximum dispersion degree, we get formulas (19) to (22):

$$\alpha = \text{Orness}(W) = \frac{1}{m} - \sum_{i=1}^{m} (m-i)\omega_i$$  \hspace{1cm} (19)

$$\ln \omega_i = \frac{i - 1}{m - 1} \ln \omega_m + \frac{m - i}{m - 1} \ln \omega_1 \Rightarrow \omega_i = \alpha^{i-1} \omega_m$$  \hspace{1cm} (20)

$$\omega_1 \left[ \frac{m-1}{m} \alpha + 1 - m\omega_1 \right] = \left[ \frac{m-1}{m} \alpha - m \right] \omega_1 + 1$$  \hspace{1cm} (21)

$$\omega_m = \frac{m-1}{m} \alpha - m$$  \hspace{1cm} (22)

Based on specific practical requirements, participant can compute $\omega_1, \omega_2, \omega_m$ by reasonable values of $\alpha$ and formulas (20) to (22). Further, an algorithm is proposed to compute the trust attribute weight.

In Algorithm 2, the weight of the trust attribute is determined by $m$ and $a$. In a specific application, $m$ is a definite value, the key is how to reasonably determine the value of $a$. In Table 2, when $a = 1$, then $\omega_1 = 1$ and $\omega_i = \omega_i \cdots = \omega_i = 0$; when $\alpha = 0$, then $\omega_m = 1$ and $\omega_1 = \omega_2 = \cdots = \omega_i = 0$; when $\omega_1 = 1$ and $\omega_2 = \omega_3 = \cdots = \omega_m = 0$.

**Table 2. The Orness of trust attribute**

| Weight | $a = 0$ | $a = 0.1$ | $a = 0.2$ | $a = 0.3$ | $a = 0.4$ | $a = 0.5$ | $a = 0.6$ | $a = 0.7$ | $a = 0.8$ | $a = 0.9$ | $a = 1.0$ |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| $w_1$  | 0.0    | 0.0050 | 0.0389 | 0.0715 | 0.1177 | 0.20   | 0.2483 | 0.3961 | 0.5116 | 0.7004 | 1.0    |
| $w_2$  | 0.0    | 0.7004 | 0.5116 | 0.3961 | 0.2483 | 0.20   | 0.1177 | 0.0715 | 0.0389 | 0.0050 | 0.0    |
| $w_3$  | 0.0    | 0.0144 | 0.0436 | 0.0683 | 0.0840 | 0.20   | 0.0840 | 0.0683 | 0.0436 | 0.0144 | 0.0    |
| $w_4$  | 0.0    | 0.0601 | 0.1039 | 0.1572 | 0.1419 | 0.20   | 0.1419 | 0.1572 | 0.1039 | 0.0601 | 0.0    |
| $w_5$  | 1.0    | 0.2151 | 0.3020 | 0.4089 | 0.4111 | 0.20   | 0.4111 | 0.4089 | 0.3020 | 0.2151 | 0.0    |
Algorithm 3. Feedback trust $Y_2(D_i, D_j)$.

Input: $N$, $n$, $H_f$, $\eta$, LEVEL;
Output: $Y_2(D_i, D_j)$;
1 construct MWD by formula (5) to (9);
2 compute $F(D_i)$ by Algorithm 1;
3 for (all $D_k \in F(D_i)$) do
4 compute $Y_1(D_i, D_k)$ by formulas (5) to (9);
5 compute $\mu_k$ by formula (12);
6 compute $Y_2(D_i, D_k)$ by formula (11);
7 end.

Algorithm 4. Total trust computing.

Input $H, P, M, F(D_i) = \{F_1, F_2, \ldots, F_l\}$;
Output $T(G(D_i, D_j, S, t))$;
1 construct feedback node net
2 calculate decision factors:
3 $Y_1(D_i, D_j)$ — direct trust function,
4 $Y_2(D_i, D_j)$ — feedback trust function,
5 $Y_3(D_i, D_j)$ — obligation trust function,
6 $Y_4(D_i, D_j)$ — trust risk function,
7 $Y_5(D_i, D_j)$ — reward penalty function;
8 compute the weight $\omega_m$ of trust function;
9 compute total trust $T(G(D_i, D_j, S, t))$;
10 end.

$\alpha = 0.5$, then $\omega_1 = \omega_2 = \cdots = \omega_j = \cdots = \omega_m = 1/m$; and when $0 < \alpha < 1, a \neq 0.5$, we can obtain several values of $\omega_i$.

**Total trust computing**

In cloud computing, malicious node can submit dishonest feedback to raise another malicious agents' reputation. It is an effective way to avoid malicious feedback by quality factor $\eta$. For example, $\eta = 0.60$ expresses that the system does not use nodes whose trust value is less than 0.60, Algorithm 3 is given below.

$D_i$ request service from the provider $D_j$, if the $D_i$ has no interaction record of $D_i$ in the database, according to Algorithms 1 to 3, so we can obtain Algorithm 4.

According to Algorithms 1 to 4, the system can decide to accept or reject request, whether $s_i$ is consistent with the trust function $T(G(D_i, D_j, S, t))$. If $\psi(T(G(D_i, D_j, S, t))) \geq s_i$, the system can provide $s_j$ to $D_i$, else refuse request.

**Experiment design**

In this article, we construct a trust model for cloud service (TACM), contain comparison and analysis of the two methods, multifaceted trust management framework (MTMF) based on a trust level agreement in a collaborative cloud\textsuperscript{[11]} and NMTR (a new method for trust and reputation evaluation in the cloud environments using the recommendations of opinion leaders' entities and removing the effect of troll entities\textsuperscript{[9]}). Experiment parameters are presented in Table 3.

![Convergence time t vs. simulation cloud service entity](image.png)

**Figure 4.** The aggregation time of three models.

Datasets are on the http://www.uoguelph.ca/qmahmoud/qws/, which contains 5000 services (Table 4).

**Efficiency evaluation**

In the section, we define $t_c$ as the time cost of trust aggregation under various network entities. $m_c$ represents the average storage space of all kinds of data structures, $m_{\text{total}}$ is the total storage space, $N$ is the number of nodes, and $m_c$ is defined as formula (23):

$$m_c = \frac{m_{\text{total}}}{N}$$  \hspace{1cm} (23)

**Time cost.** In Figure 4, with the enlargement of network node size, the aggregation time $t_c$ of three models increases rapidly, TACM needs less computation time than NMTR and MTMF. This shows that the TACM has better convergence performance, because feedback weight factor can
adjust the polymerization calculation scale, the trust computing does not take too much time; however, NMTR and MTMF do not have these functions.

Storage cost. Average storage cost is also an important index to measure cloud services.

As can be seen from Figure 5, with the increase of network size, the $m_c$ of the three models increases quickly, and the average storage of TACM is much less than NMTR and MTMF. In the TACM model, quality factor, trust risk, distance factor, and feedback weight factor can effectively reduce the search length of the trust chain, filter out many malicious nodes; but NMTR and MTMF do not have these factors, further analysis, although NMTR has many factors of trust evaluation, lacks relevant constraint conditions of trust process; therefore, the storage space is the most in the three models.

Success rate and satisfaction degree

In the section, we use two indicators: interaction success rate and satisfaction degree to compare the three models.

Success rate of normal service. As can be seen in Figure 6, the success rate of interaction is different in three methods. Because NMTR lacks risk defense mechanism and the reward penalty factor, when malicious nodes appear, which cannot guarantee the actual service capacity. MTMF introduces trust to meet personal requirement, but it lacks the time decay factor and reward penalty function. TACM does not have these shortcomings, multiple factors can guarantee a higher success rate, such as reward penalty can increase the probability of interaction and service willingness. Therefore, the rank of success rates is TACM > NMTR > MTMF.

Success rate of malicious services. In Figure 7, when the proportion of malicious services increases, the interaction success rate of NMTR and MTMF declines rapidly. However, in the TACM, the trust risk function can filter out some malicious nodes, and the quality factor can filter out some feedback nodes with lower trust value, so the success rate is higher than the other two models. The NMTR and MTMF lack reward penalty function and antirisk capability, so they perform poorly against malicious service.

Satisfaction with different service. As can be seen in Figure 8, NMTR has a lower satisfaction that does not guarantee the actual service in the cloud environment. In the TACM, time decay function solves the trust dynamic change of over time, reward penalty can increase the probability of successful interaction, quality factor enhances the quality of service, trust risk function can filter out the malicious nodes, so the user’s service satisfaction degree is high. The trust weight of MTMF model is subjective, the experiment result is not very good. In a word, the service satisfaction rate of TACM is best in the three models.

Accuracy evaluation

To further prove research on privacy protection, in the following experiments, we manually generated 50 K data, each data includes 1000 kinds of attributes, and the value of the attribute is in the $[0,1]$.

The accuracy is measured by the error to test whether the proposed algorithm can accurately provide confidence calculation. The error is smaller, the accuracy is higher. If $A_{t+1}$ is the true value, $TG_{t+1}$ is the evaluation value at $t+1$, these following three methods are introduced to measure the accuracy of trust evaluation.

Mean absolute deviation.

$$\text{MAD} = \frac{1}{n} \sum_{t=1}^{n} |TG_t - A_t|$$

(24)
Mean absolute deviation (MAD) can be used to metrics the deviation of evaluation results; thus, \( e_t \) is the error at time \( t \), \( e_t = TG_{t+1} - A_{t+1} \), and \( n \) is the amount of service transaction times.

According to Figure 9, the average MAD of TACM, MTMF, and NMTR is 0.0928, 0.1145, and 0.1009, respectively. When the number of transactions is more than 1000, the curve of TACM changes more smoothly.
than do those of NMTR and MTMF, this shows that fewer simulation interactions can also improve the accuracy of our model. TACM can integrate information entropy into trust evaluation algorithm, so the MAD of trust evaluation is lower than NMTR and MTMF.

**Root mean square error.** Root mean square error (RMSE) is the variance of the arithmetic square root, which is used to measure the deviation between the real value and evaluation value. The RMSE is shown in the following formula (25):

\[
\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{n} (TG_i - A_i)^2}
\]  

(25)

According to Figure 10, the RMSE of TACM, MTMF, and NMTR is 0.0958, 0.1145, and 0.1044, respectively. When the number of transactions is more than 1000, the curve of TACM changes more smoothly than those of NMTR and MTMF, this shows that fewer simulation interactions can also show the accuracy of our model. TACM adopts time decay, trust feedback, and obligation, the dynamic performance is relatively good, but MTMF and NMTR lack similar mechanisms, so the RMSE is the lower than NMTR and MTMF.

**Mean absolute percentage error.** Mean absolute percentage error (MAPE) is an error measurement way, which represents the accuracy of the evaluation. \(e_t\) is the error at time \(t\), \(e_t = TG_{t+1} - A_{t+1}, TG_{t+1}\) is the evaluation value, and \(n\) is the amount of experiment times:

\[
\text{MAPE} = \frac{1}{n} \sum_{t=1}^{n} \left| \frac{e_t}{A_t} \right| (\times 100\%)
\]

(26)

In Figure 11, the average MAPE of TACM, NMTR, and MTMF is 10.61%, 11.91%, and 12.25%, respectively. When the number of transactions exceeds 1000, MAPE of TACM is much lower than the other two models. Based on the comparison among Figures 9 to 11, TACM is better than NMTR and MTMF. Because the trust model of TACM has good dynamic adaptability and each weight factor is objective and accurate, the error rate is low, while NMTR and MTMF do not have the ability.
Based on the above sections, we further divide the 50 K data set into three categories of privacy sensitivity: high (H), medium (M), and low (L). Assuming that the trust level of consumer $T_i$ is below the threshold requirement for the $i$th interaction, this is considered as a privacy disclosure $E_{di}$. So, we define privacy disclosure rate as follows:

$$\text{Privacy disclosure rate} = \frac{\sum_{i=1}^{n} E_{di}}{rq}$$  \hspace{1cm} (27)$$

where $rq$ expresses all possible request, and $n$ is the amount of transactions.

Figures 12 to 14 show the privacy disclosure rates, respectively, when these categories of L, M, and H vary from 0 to 1. In Figure 12, the privacy disclosure rates of NMTR, TACM, and MTMF are from 0.289, 0.352, 0.361 to 0, respectively. In Figure 13, with the improvement of M, the privacy disclosure rates of TACM, NMTR, and MTMF are finally settled in 0.215, 0.272, and 0.285, respectively. In Figure 14, with the increment of H, the privacy disclosure rates of TACM, NMTR, and MTMF are settled in 0.281, 0.350, and 0.361, respectively. These experimental results show that TACM is better than MTMF and NMTR in terms of privacy protection.

In MTMF, trust relationship model is relatively simple, which cannot protect privacy well in cloud computing. In the NMTR, the lack of objective quantitative formula for the weight of trust attributes seriously impacts on privacy protection. TACM can not only use weight algorithm to adjust trust but also filter out the hidden dangers of insecurity through risk and penalty factors. Therefore, it is superior to MTMF and NMTR in privacy protection.

**Conclusion**

To mutual trust and win–win cooperation between the users and the CSP, in this article, we establish an access control model based on trust evaluation. First, we propose a multi-attribute trust model based on direct trust, trust risk, feedback trust, reward penalty, and obligation trust to describe the complexity of trust relationship; second, we propose a weight method of the trust attribute by the information entropy, which transcends the limitations of traditional multiple attribute weighting methods; third, we design several relevant experiments to evaluate adaptability, accuracy, and efficiency in the cloud environment.

Of course, there are still some shortcomings in this article, the dynamic cloud service selection model needed further improvement; the obligation can be optimized to improve the service accuracy. Further, we will study access control in the mobile networks; in addition, cloud service performance, reliability, and other aspects can be researched more depth in the future.

**Acknowledgment**

The author thanks the critics for their detailed comments and valuable suggestions, which greatly improve the quality of this article.

**Declaration of conflicting interests**

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

**Funding**

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work is financially supported by the National Development and Reform Commission, Information Security Special Project, Development and Reform Office, no. 1424 [2012].

**ORCID iD**

Panjun Sun https://orcid.org/0000-0002-4634-5886

**References**

1. Cloud Security Alliance. *Security Guidance for Critical Areas of Focus in Cloud Computing V4.0*. 2017. https://cloudsecurityalliance.org/artifacts/security-guidance-v4/

2. Zhifeng X. and Yang X. Security and privacy in cloud computing. *IEEE Commun Surv Tut* 2013; 15(2): 843–859

3. Aluvalu RK and Maddana L. A survey on access control models in cloud computing. India: Springer, 2015. Emerging ICT for Bridging the Future.
4. Rohit R and Bharat B. EPICS: a framework for enforcing security policies in composite web services. *IEEE Trans Serv Comput* 2019; 12(3): 415–428

5. Bhatia R and Singh M. A novel trust-based privacy preserving access control framework in web services paradigm. *Adv Intell Syst, Adv Intel Syst Comput* 2015; 308: 441–453.

6. Khaled R and Zhu Y. Multi-factor synthesis decision-making for trust-based access control on cloud. *Int J Coop Inf Syst* 2017; 26(4): 1750003.

7. Junqi D and Deyun G. TC-BAC: a trust and centrality degree-based access control. *Ad Hoc Networks* 2013; 11(8): 2675–2692.

8. Nogoorani SD and Jalili R. TIRIAC: a trust-driven risk-aware access control framework for Grid environments. *Future Gener Comp Sys* 2016; 55: 238–254.

9. Matin C and Nima JN. A new method for trust and reputation evaluation in the cloud environments using the recommendations of opinion leaders’ entities and removing the effect of troll entities. *Comput Hum Behav* 2016; 60: 280–292.

10. Yan Z, Ding W, Niemi V, et al. A privacy preserving authorisation system for the cloud. *Future Gener Comp Sys* 2016; 62 (C): 175–189.

11. Varalakshmi P and Judgi T. Multifaceted trust management framework based on a trust level agreement in a collaborative cloud. *Comput Electr Eng* 2017; 59: 110–125.

12. Halabi T and Bellaiche M. Towards quantification and evaluation of security of cloud service providers. *J Inf Sec Appl* 2017; 33: 55–66.

13. Simeon V and Iraklis P. Achieving security by design through ontology-driven attribute-based access control in cloud environments. *Future Gener Comp Sys* 2019; 93: 373–391.

14. Yan Z, Ding W, Niemi V, et al. Two schemes of privacy-preserving trust evaluation. *Future Gener Comp Sys* 2016; 62 (C): 175–189.

15. Lan Z and Vijay V. Trust enhanced cryptographic role-based access control for secure cloud data storage. *IEEE Trans Inf Forensics Secur* 2015; 10(11): 2381–2395.

16. Xue Y and Xue K. An attribute-based controlled collaborative access control scheme for public cloud storage. *IEEE Trans Inf Forensics Secur* 2019; 14(11): 2927–2942.

17. Smari WW and Clemente P. An extended attribute based access control model with trust and privacy: application to a collaborative crisis management system. *Future Gener Comp Sys* 2014; 31: 147–168.

18. Ray I and Ray I. Trust-based access control for secure cloud computing. In: Han KJ, Choi B-Y, and Song S (eds) *High performance cloud auditing and applications*. New York, NY: Springer, 2014, pp. 189–213.

19. Li X. A method for trust quantification in cloud computing environments. *Int J Distrib Sens N* 2016; 12: 1–7.

20. Rudolph-Lilith M and Muller LE. Algebraic approach to small-world network models. *Physical Review E* 2014; 89: 012812.

21. Fuller R and Majlender P. An analytic approach for obtaining maximal entropy OWA operator weights. *Fuzzy Set Syst* 2001; 124(1): 53–57.

22. Kim SH. An entropy-based analytic model for the privacy protection in open data. In: *IEEE International Conference on Big Data*, Washington, 5–8 December 2016. IEEE.

23. Tyagi AK and Niladhuri S. Never trust anyone: trust-privacy tradeoffs in vehicular Ad-Hoc Networks. *Brit J Math Comp Sci* 2016; 19(6): 1–23.

24. Shynu PG and Singh KJ. A comprehensive survey and analysis on access control schemes in cloud environment. *Cybern Inf Technol* 2016, 16 (1):19–38.