A Trust-Based Defensive System Model for Cloud Computing

Qian Zhou[1], Jiong Yu[1], and Feiran Yu[2]

[1] College of Information Science and Engineering, Xin jiang University, Urumqi, Xin jiang, China
{zhouqian,yujiong}@xju.edu.cn
[2] Control & Computer Technology School, North China Electric Power University, Beijing, China
yufeiran@yahoo.com.cn

Abstract. Because of the dynamic and open characteristics of the cloud computing, continuing malicious attacks happen frequently. Combining the idea of trusted cloud, a trust-based defensive system model for cloud computing has been constructed to guarantee the cloud security. Through real-timely monitoring, users' behavior evidences have been obtained and standardized; a new method for users' trust evaluation based on fuzzy AHP (Analytic Hierarchy Process, AHP) has been presented, it gradually determines the weights of behavior evidences, achieves quantitative assessment of behavioral trust; to provide great security defense for users, multiple detection engines have been used to conduct a comprehensive inspection of suspicious files and integrated decisions have been made. Experimental results show the system model can effectively eliminate the malicious behaviors from undesirable users, reduce users' damages caused by virus and achieve a two-way defense for both cloud and client.

Keywords: cloud security, trusted cloud, trust, behavioral evidence, fuzzy AHP, multiple detection engines.

1 Introduction

In the face of continuing malicious attacks, the simple methods of intrusion detection, virus detection and secure login protocol have been unable to cope with a variety of network attacks and damages, therefore cloud security and trusted cloud [1] came into being. Cloud security is intended to eliminate existing viruses, trojans and malicious files in the network. Trusted cloud is guaranteed safe from the user terminals; combining the idea of trusted network [2], it evaluates, forecasts, monitors and manages user behaviors to eliminate malicious attacks to data center from undesirable users and hackers in the cloud to enhance the security of the cloud environment.

In recent years, many scholars began the research of the trust mechanism to make up the defect of traditional security mechanisms that the trust of user behaviors was not considered. Song et al.[3] propose a dynamic trust model based on fuzzy logic under

E. Altman and W. Shi (Eds.): NPC 2011, LNCS 6985, pp. 146–159, 2011.

grid environment, this model has better capacities of detecting and defending malicious entities, while the downside is that the convergence of computing and system scalability are poor, it does not consider the calculation of the indirect trust and the trust can not reflect the overall credibility. Power-Trust[4] is a P2P reputation system based on the power law distribution, the system uses power law to collect feedback from local nodes, and gets global reputation through the super-nodes generated from queuing mechanism, significantly improves the accuracy of global reputation and accelerates the rate of polymerization. Jameel introduces vector operation mechanism to establish the trust model [5], its most notable feature is the introduction of the trust factor, historical factors and the time factor to reflect dynamic trust relationship, but it can not solve the cheating behaviors when recommending and has no risk analysis. A new trust quantitative model based on multi-dimensional decision-making properties is proposed in paper [6]; it introduces direct trust, risk function, feedback trust, activation function and physical activity level and other decision-making attributes to assess the complexity and uncertainty of trust relationship from various angles.

Through analyzing and comparing the existing trust models, studying the trusted cloud and reputation technology of the client cloud [7], in this paper we change traditional ideas of network defense which are for their own business and propose a trust-based defensive system model in the cloud environment. This model has integrated trust and risk evaluation mechanisms and provides maximum security defense to customers in the cloud and the network security and defense functions are provided as services to end customers.

2 Architecture of Defensive Model

Cooperation between network services, resources, applications provided by cloud computing and customers depends on the trust relationship between them. In this paper, the system model is built by borrowing the idea of trusted network, therefore the network entities' behavioral states can be monitored, behavioral outcomes can be assessed and abnormal behaviors can be controlled [8].

2.1 Physical Structure

The system consists of Cloud Client (CC), File Monitoring and Analyzing center (FMA), Behavior Monitoring center (BM) and Trust Management center (TM) in which CC is a lightweight terminal located in the client and the rest are high-performance large-scale servers in the cloud. Figure 1 shows model's physical structure.

CC is responsible for submitting samples of suspicious files and executes the final decision.

FMA is composed of file detecting engines and file behavior analyzing engine. Using virtualization technology and multi-engine detection mechanism, FMA takes a full inspection to the uploaded suspicious files and returns the test results to the customer.

BM is consisted of a behavioral evidence obtaining module and a standardizing module. It monitors customers' behaviors continuously, obtains and standardizes users' behavioral evidences.

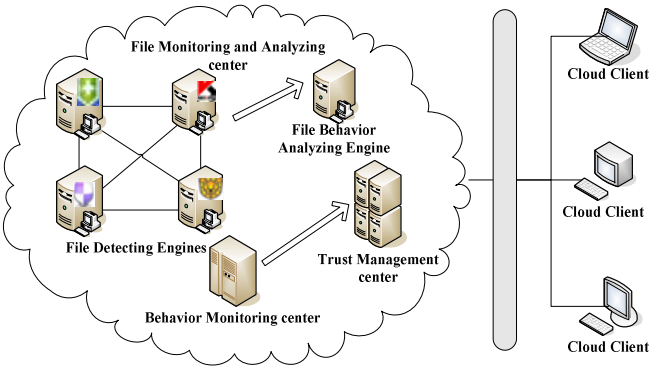

Fig. 1. Physical structure of the defensive system

TM is consisted of a behavior database, a behavioral trust evaluating module and a trust database. It stores users' behavioral evidences, evaluates users' trust degree and saves users' overall trust degree.

2.2 Logical Structure

To provide security and defensive services for the massive clients, this model is designed to take full advantages of the high computing power and storage power of the cloud computing. At the same time, the system monitors and assesses users' behaviors to eliminate malicious attacks from undesirable users which are in interaction in the cloud to achieve a two-way defensive purpose. Figure 2 shows the flow chart of the model.

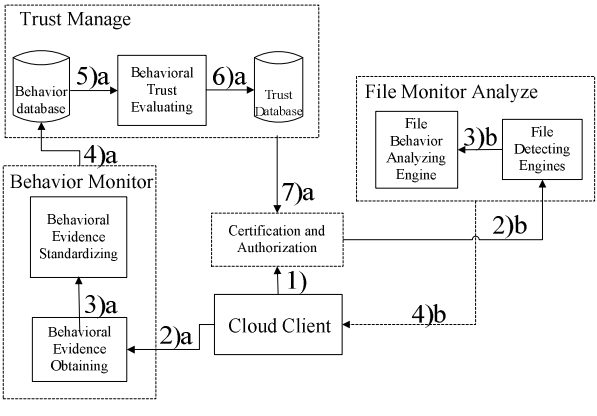

Fig. 2. Flowchart of the defensive system

The system flow can be divided into the following steps: 1) When a user login to the cloud, the cloud's certification and authorization module queries Trust Management

center to give the user a trust level and the corresponding operating authorities; 2)a The user is monitored by Behavior Monitoring center continuously and behavioral evidence obtaining module gets the user's behavioral evidences; 3)a Evidences obtained from the user will be standardized by Behavioral standardizing module; 4)a Behavior Monitoring center dumps standardized behavioral evidences to the behavior trust database in Trust Management center; 5)a Trust evaluation module uses the data in the behavior database to evaluate the user's trust value; 6)a Trust evaluation module sends the user's trust value to the trust database; 7)a In the basis of the user's trust value, the certification and authorization module authorizes real-time permission to the user; 2)b Suspicious file samples are submitted to File Monitoring and Analyzing center through Cloud Client by the client; 3)b Suspicious files that are unable to be determined by file detecting engines are submitted to file analyzing engine for real-timely dynamic behavioral analysis; 4)b The final decisive results of the suspicious files are returned to the client by Files Monitoring and Analysis center and the final decision is taken by the user.

The system's processing flow is composed by two extensions: 'a' process provides defense for the cloud through evaluating the user's behavior trust and giving the appropriate operating authorities to the user; 'b' process contains detecting and analyzing malicious files uploaded by the end user, establishing the latest and most complete trusted/malicious file database to provide maximum defense for users in the cloud.

2.3 Relevant Description

Definition 1. Suppose there are m measure items for measuring users trust degree, let E=$\{E_1, E_2, \ldots, E_m\}$ denote m kinds of behavioral evidence and the measured values are expressed as I=$\{I_1, I_2, \ldots, I_m\}$. Let w_i $(1 \le i \le m)$ represents the importance of the ith piece of evidence relative to the other evidences, and meet

$$0 \le w_i \le 1, \quad \sum_{i=1}^{m} w_i = 1$$

Definition 2. Let Tr denote the overall trust degree of the user evaluated by the cloud, denoted as Overall Trust Degree (OTD) and OTD is the basis for the user to get authorities during the interaction with the cloud.

Definition 3. Suppose set C is the behavior trust classification space, denoted as C=$\{C_1, C_2, \ldots, C_p\}$, respectively represents $p+1$ trust levels for Tr, where $0<C_p<1$ and $1 \le p \le m$. C has following properties: $C_i \cap C_j = \Phi (i \ne j)$, $C_1 < C_2 < \ldots < C$, and C_{k+1} is stronger than C_k, then we say C is an ordered partition class.

Definition 4. Suppose the cloud provides k levels of services denoted Se=$\{Se_1, Se_2, \ldots, Se_k\}$, Se is an ordered partition class, and the mapping function between Se and Tr is

$$fs(Tr)=\begin{cases} Se_k, & C_{k-1} < Tr \le 1 \\ \cdots & \\ Se_2, & C_1 < Tr \le C_2 \\ Se_1, & 0 < Tr \le C_1 \end{cases} \tag{1}$$

Where C_k etc. are defined by definition 3. When a user requests to the cloud for services, the user will be decided his service level based on his trust level to reduce existing potential risk of the user.

For example, some cloud operator provides 3 levels of services, then Se={Se_1, $Se2$, $Se3$}. Which, Se_1 denote services are denied, Se_2 denote files are read only, Se_3 denote files can be edited. Trust level space is set to C={C_1, $C2$}={0.3, 0.6}, the decision-making function for services is

$$fs(Tr)=\begin{cases} Se_3, & 0.6 < Tr \le 1 \\ Se_2, & 0.3 < Tr \le 0.6 \\ Se_1, & 0 < Tr \le 0.3 \end{cases}$$

If Tr = 0.7, then the decision-making process is $fs\ (Tr) = f\ (0.7) = Se_3$= file can be edited.

Definition 5. Let $Trp(0 \le Trp \le 1)$ denote the trust properties of the suspicious files after being detected by the cloud; suspicious files trust level space is denoted by V={V_1,V_2}, and $V_1 < V_2 \in (0, 1)$. So, the decision-making function for suspicious files trust levels is

$$fp(Trp)=\begin{cases} Trusted, & V_2 \le Trp < 1 \\ Unknown, & V_1 \le Trp < V_2 \\ Malicious, & 0 \le Trp < V_1 \end{cases} \tag{2}$$

3 Behavioral Evidences

3.1 Obtaining Evidences

User's behavioral evidences (evidences for short) can be obtained directly from the detection of system software and hardware, which are the base values for quantitatively assessing user's overall behaviors [9]. Current methods for obtaining evidences are: intrusion detection systems, such as Snort, which can detect worms, vulnerabilities, port scanning and a variety of suspicious behaviors; network traffic detection and analysis tools, such as Bandwidthd, you can view highly detailed IP traffic and network status; professional network data collection tools, such as Flunk's NetFlow Tracker, which can get real-time network bandwidth utilization and bandwidth usage of different users and so on.

3.2 Standardizing Evidences

Evidences have many forms: specific values, such as the number of scanning an important port; percentage, such as CPU utilization; binary, such as data integrity (1 if data is complete, 0 if data is incomplete). In order to facilitate numerical calculation and assessment to user behavior, evidences need to be standardized as positive increasing dimensionless values in the interval [0, 1].

Let $A= (a_{ij})_{m\times n}$ denote initial evidences matrix and we normalize A to evidences matrix $E=(e_{ij})_{m\times n}$, the rules are:

① Evidences of percentage and binary forms are already in [0,1] and they are translated into positive increasing values using equation (3)

$$e_{ij} = \begin{cases} a_{ij} & ,when\ a_{ij}\ is\ positive increasing \\ 1-a_{ij} & ,when\ a_{ij}\ is\ positive decreasing \end{cases} \tag{3}$$

② Evidences of specific values are normalized into positive increasing values in [0,1] using equation (4)

$$e_{ij} = \begin{cases} \dfrac{a_{ij}-(a_{ij})_{min}}{(a_{ij})_{max}-(a_{ij})_{min}} & ,when\ a_{ij}\ is\ positive increasing \\ \dfrac{(a_{ij})_{max}-a_{ij}}{(a_{ij})_{max}-(a_{ij})_{min}} & ,when\ a_{ij}\ is\ positive decreasing \end{cases} \tag{4}$$

Where, $(a_{ij})_{min}$ and $(a_{ij})_{max}$ respectively express the minimum and maximum evidence.

4 Evaluation Model Based on FAHP

AHP (Analytic Hierarchy Process) is a system analysis method combining qualitative and quantitative analysis [10]. The traditional AHP has following shortcomings [11]: it uses nine scales to construct judgment matrix and this is too complex in practice; it needs to judge and build matrix continuously until the matrix meets consistency verification with large calculation and low precision.

In order to overcome the problems in AHP, this paper uses fuzzy analytic hierarchy process (Fuzzy AHP). FAHP uses three scales to construct a judgment matrix to facilitate decision-makers easily decide in two factors which is relatively important; and the initial judgment matrix is transformed into the fuzzy consistent matrix that satisfies the consistency condition without consistency test.

4.1 A Hierarchical Structure of Evidences

Evidences are divided into several trust properties gradually by layer, and then broken down into specific evidences' types, which can effectively resolve the general and uncertain problems of user's behavioral trust under cloud computing. A hierarchical structure of evidences is shown in Figure 3.

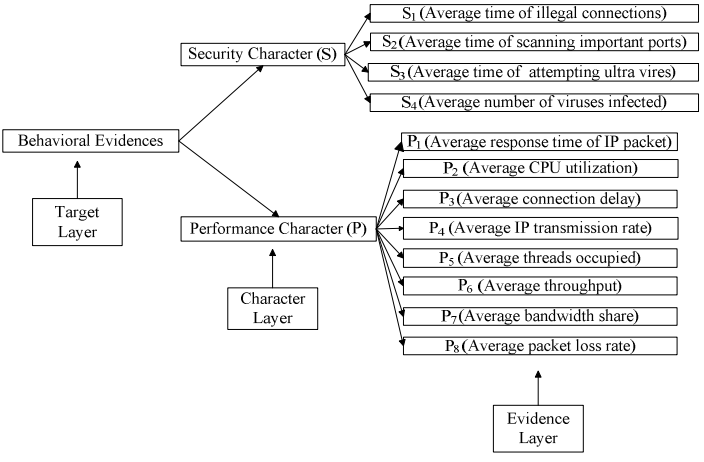

Fig. 3. Hierarchical structure of user's behavioral evidences

4.2 The Weight Determined by FAHP

(1) Establish an initial judgement matrix

In Figure 3, suppose there are m evidences related to the performance character, respectively denoted as $\{ep_1, ep_2, \ldots, ep_m\}$. For the m evidences, according to their importance relevant to the performance character, through pairwise comparison we get an m-order initial judgment matrix $EP = (ep_{ij})_{m\times m}$

$$ep_{ij} = \begin{cases} 0.5 & ,c(i) = c(j) \\ 1 & ,c(i) > c(j) \\ 0 & ,c(i) < c(j) \end{cases} \tag{5}$$

Where, $c(i)=c(j)$) express evidence ep_i is equally important as ep_j; $c(i)>c(j)$ express evidence ep_i is important than ep_j; $c(i)<c(j)$ express evidence ep_i is less important than ep_j.

(2) Steps of sum and transformation of line are used to convert the initial judgment matrix $EP=(ep_{ij})_{m\times m}$ to the fuzzy consistent matrix $Q=(q_{ij})_{m\times m}$ without consistency verification.

$$q_{ij} = \frac{q_i - q_j}{a} + 0.5 \tag{6}$$

$$\left(q_i = \sum_{k=1}^{m} ep_{ik}, i = 1,2,\ldots,m; a = 2m \right)$$

(3) When conducting sum of line for the fuzzy consistent matrix Q, self-comparison (i.e., diagonal element) is not contained. The weight vector $W= (w_1, w_2 \ldots w_m)^T$ is Calculated using equation (7)

$$w_i = {l_i}/{\sum_i l_i} \tag{7}$$

Where, l_i denote evidence p_i's importance relative to the upper performance character

$$l_i = \sum_{k=1}^{m} q_{ik} - 0.5, \quad i = 1,2,\ldots m \tag{8}$$

$$\sum_i l_i = {m(m-1)}/{2} \tag{9}$$

4.3 The Assessment of User Behavior Character

Suppose user behavior contains n items of features, and k denote the largest item value of the feature (if the item number does not reach k, the corresponding elements are 0); we use Section 3.2 to get standardized evidence matrix $E=(e_{ij})_{m\times n}$, $e_{ij}\in[0,1]$ represents the j th evidence of the i th character and its weight is $w_{ij}\in[0,1]$. Matrix diagonal values calculated by equation (10) are the assessment values of user character.

$$E\times W^T = \begin{pmatrix} e_{11} & e_{12} & \cdots & e_{1k} \\ e_{21} & e_{22} & \cdots & e_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ e_{n1} & e_{n2} & \cdots & e_{nk} \end{pmatrix} \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1k} \\ w_{21} & w_{22} & \cdots & w_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & \cdots & w_{nk} \end{pmatrix}^T \tag{10}$$

4.4 The Overall Trust Evaluation of User Behavior

Set the user feature vector is $F=(f_1,f_2,\ldots,f_n)^{\mathrm{T}}$, the weight vector of the features is $W_f^T=(w_1,w_2,\ldots,w_n)^{\mathrm{T}}$, user's overall trust degree Tr is calculated using equation (11)

$$Tr = 1 - F\times W_f^T = 1 - \sum_{i=1}^{n} f_i w_i \tag{11}$$

5 Evaluation Model for Suspicious Files

Suspicious files may be user's normal files, original virus files, files infected by virus, files corrupted by virus, malicious software and so on. In order to reduce the losses caused by viruses and malicious software, we use multi -engine detection mechanism and dynamic behavior analysis to test comprehensively the suspicious files uploaded by the user.

5.1 Multi-engine Detection Mechanism

As a single engine can not conduct a comprehensive inspection to suspicious files, file detection engine in File Monitoring and Analysis center (FMA) then uses multi-engine detection mechanism.

(1) Suppose we use n kinds of detection engines denoted as Eg=$\{Eg_1,Eg_2,\ldots,Eg_n\}$ and the accuracy of the engines is Per=$\{Per_1,Per_2,\ldots,Per_n\}$.

(2) Using n kinds engines to test a single suspicious file, we can get n kinds detection results R= $\{R_1, R_2, \ldots R_n\}$ ($R_i \in \{0,1\}$, 0 indicates that no virus is found, 1 indicates virus is found). Integrated trust property value Trp of the suspicious file is calculated using equation (12)

$$Trp = 1 - \sum_{i=1}^{n} Per_i \times R_i \Big/ n \tag{12}$$

(3) According to equation (2), the file's trust level can be determined. When the file is *Trusted* or *Malicious*, FMA will directly return the results to the user for the final decision.

5.2 Dynamic Behavior Analysis of Suspicious Files

When the file's trust level is *Unknown*, FMA will hand the file to the file behavior analyzing engine, which is consisted of feature database of malicious files and the virtual machine. The *Unknown* file will be run in the virtual machine for real-timely dynamic analysis of behaviors.

6 Simulation and Analysis

6.1 Experimental Simulation and Description

Experimental environment consists of 3 servers, 1 router and 6 clients. We use C#.Net to realize the certification and authorization module, the behavioral evidence standardizing module, the file detection engines and the behavioral trust evaluating module; suspicious files are run in VMware workstation and analyzed dynamically; kinds of services under the experimental environment are monitored continuously by NetFlow Tracker and the test scenario is a small cloud storage system.

(1) According to Section 2.3 Definition. 3, the user trust level is set C={0.3, 0.6}; user is authorized according to Definition. 4, this small cloud storage system provides three levels of service Se=$\{Se_1,Se_2,Se_3\}$ (Se_1 denote services are denied, Se_2 denote files are read only, Se_3 denote files can be edited and download); according to Definition. 5, suspicious files trust level is set V={0.4, 0.5}. Parameter settings are shown in Table 1.

Table 1. Parameter Settings

User	OTD: *Tr*	[0,0.3)	[0.3,0.6)	[0.6,1)
	Trust level	Low	Normal	High
	Service level	S_1	S_2	S_3
Malicious Files	*Trp*	[0,0.4)	[0.4,0.5)	[0.5,1)
	Trust level	*Malicious*	*Unknown*	*Trusted*

(2) When a user is in the process of interaction with the cloud, BM continuously monitors the user and puts his behavioral evidences into the behavior database, the trust evaluating module assesses the trust degree of the user to identify and predict the possible unforeseen circumstances and real-timely informs the certification and authorization module to revise to the service level of the user.

(3) This system uses multiple detection engines and their accuracy *Per* are shown in the following [12]: GData(*Per*=99.9%)、AntiVir(*Per*=99.8 %)、AVAST (*Per*=99.3%)、Norman(*Per*=96.6%)、Trend Micro(*Per*=90.3%), Kingsoft(*Per*=80.1%).

6.2 Example of Trust Evaluation

12 kinds of evidences (see evidence level in Figure 3) in a half-hour from a client are obtained by Behavior Monitoring center. Through Section 3.2, evidences are normalized and convert to average evidence values that are shown in Table 2.

Table 2. The average evidence values in a half-hour of a client

Performance Character (P)							
P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8
0	0	0	0	0	0	0	0
.62	.51	.88	.83	.74	.67	.73	.54

Security Character (S)			
S_1	S_2	S_3	S_4
0	0	0	0
.33	.28	.15	.08

We use Section 3.2 to determine the weight of performance and security characters respectively and take per- formance character for instance: Experience has shown that response time of IP packet, IP transmission rate, throughput and bandwidth share are the best reflection to user's performance character and they are of equal importance, therefore $[c(P_1)=c(P_4)=c(P_6)=c(P_7)]>[c(P_2)=c(P_8)]>[c(P_3)=c(P_5)]$. Construction of the Initial judgment matrix is constructed by equation (5)

$$EP=\begin{pmatrix} 0.5 & 1 & 1 & 0.5 & 1 & 0.5 & 0.5 & 1 \\ 0 & 0.5 & 1 & 0 & 1 & 0 & 0 & 0.5 \\ 0 & 0 & 0.5 & 0 & 0.5 & 0 & 0 & 0 \\ 0.5 & 1 & 1 & 0.5 & 1 & 0.5 & 0.5 & 1 \\ 0 & 0 & 0.5 & 0 & 0.5 & 0 & 0 & 0 \\ 0.5 & 1 & 1 & 0.5 & 1 & 0.5 & 0.5 & 1 \\ 0.5 & 1 & 1 & 0.5 & 1 & 0.5 & 0.5 & 1 \\ 0 & 0.5 & 1 & 0 & 1 & 0 & 0 & 0.5 \end{pmatrix}$$

The fuzzy consistent matrix is converted from *EP* by equation (6):

$$Q=\begin{pmatrix} 0.5 & 0.6875 & 0.8125 & 0.5 & 0.8125 & 0.5 & 0.5 & 0.6875 \\ 0.3125 & 0.5 & 0.625 & 0.3125 & 0.625 & 0.3125 & 0.3125 & 0.5 \\ 0.1875 & 0.375 & 0.5 & 0.1875 & 0.5 & 0.1875 & 0.1875 & 0.375 \\ 0.5 & 0.6875 & 0.8125 & 0.5 & 0.8125 & 0.5 & 0.5 & 0.6875 \\ 0.1875 & 0.375 & 0.5 & 0.1875 & 0.5 & 0.1875 & 0.1875 & 0.375 \\ 0.5 & 0.6875 & 0.8125 & 0.5 & 0.8125 & 0.5 & 0.5 & 0.6875 \\ 0.5 & 0.6875 & 0.8125 & 0.5 & 0.8125 & 0.5 & 0.5 & 0.6875 \\ 0.3125 & 0.5 & 0.625 & 0.3125 & 0.625 & 0.3125 & 0.3125 & 0.5 \end{pmatrix}$$

The weight vector of evidences relevant to the performance character can be calculated by equation (7)~(9): $W_P^T=\{0.161,0.107,0.071,0.161,0.071,0.161,0.161,0.107\}^T$. Through pairwise comparison, the importance of evidences relevant to security character is $[c(S_3)=c(S_2)]>c(S_1)>c(S_4)$, similarly, the results of their weight vector is: $W_S^T=\{0.208, 0.333, 0.333, 0.125\}^T$. The importance of user behavior character is $c(P)<c(S)$, then the weight vector of the characters is $W_f^T=\{0.25, 0.75\}^T$. According to equation (10), the assessed values of user behavioral characters are $F=E\times W^T=(0.656,0.222)^T$, then the evaluated value of user behaviors is calculated by equation (11): Tr=0.67. According to Section 6.1 we can see, this user has high credibility level that can edit and download files in the small cloud storage system.

6.3 Experimental Comparison

Experiment 1: The experimental comparison result of two different trust evaluation mechanisms respectively based on FAHP and AHP.

Figure 4 shows the following circumstance: In the interaction between the client and the cloud, when we gradually increase the proportion of malicious acts such as illegal connections and scanning important port, the changes of the overall trust degree. With increase in the proportion of malicious behaviors, trust value calculated by mechanism

based on FAHP has a large decline compared to that of AHP and is more compatible with human reasoning. Combined with the settings in Section 6.1, we find that user with low credibility can be monitored out by FAHP earlier and faster than AHP, which is conducive for certification and authorization module to update the client's service level and reduce the risk of the cloud.

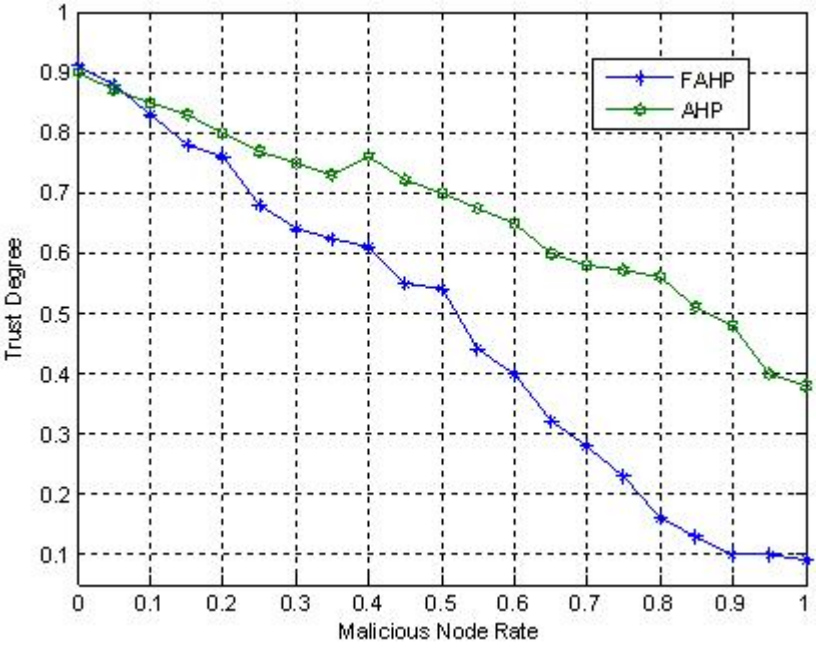

Fig. 4. Changes of a client's trust value

When using FAHP to construct the initial judgment matrix, because of the subjectivity of decision makers, determination of each element should be comprehensively evaluated by a number of experts. Also, the environment of the client should be considered when constructing a judgment matrix, if the client is in an unsafe environment (such as Internet cafes), the security character is important than performance character. Thus for the same client, even if evidences obtained are the same, the judgment matrix would be different under different environment, so do the assessed values of their behaviors.

Experiment 2: The comparison result of suspicious files detection based on multi-engines and single engine.

100 suspicious files are uploaded to the cloud and comprehensively detected by FMA. Experiments were carried out for the following three cases: 1) case is using multi-engines detection mechanism proposed in this paper to conduct a comprehensive inspection to suspicious files and determine their trust level; 2), 3) are using different single-engine to test, as a reference for the first case. Test results of suspicious files are shown in Table 3.

Table 3. Test results of suspicious files

	Trusted	*Unknown*	*Malicious*
Multi-engines	70.8%	4.2%	25%
Single-engine (AntiVir)	16.7%	0	83.3%
Single-engine (AVAST)	76%	0	24%

From Table 3, we can see that even for the same suspicious files, test results show great difference because of different detection engine (AntiVir and AVAST), so *Trusted* file could be mistaken for the *Malicious* one, thereby the client may be affected to make final decisions to suspicious files. Our system is based on multi-engines detection mechanism and conducts the integrated decision-making to test results, which can effectively avoid the above situations.

6.4 Performance Comparison

Using AHP to determine the weight of evidence, consistency test is needed. Because AHP uses nine scales to conduct the judgment matrix, excessive subjective judgments can easily bring deviation so that the matrix does not meet the consistency condition. At this point, the matrix need to be adjusted by number of experts' re-determine and a series of iterative computing, as the evidence layer is larger and the order of judgment matrix is higher, which means a large computing and communication overhead. In this paper, the experiment found that the average adjusted time is 3 when conducting the judgment matrix that meet the consistency condition using AHP method.

Without iterative calculation, FAHP method used in this paper can construct the fuzzy consistent matrix that meets the consistency condition just one-off. Therefore, under the same computation and communication, the computing time of our model can be shortened 1/3, so user's trust level and authorities can be judged faster. Therefore, the complex and time cost is greatly reduced.

7 Conclusion

Due to the severe security problems that cloud computing faces, combining the idea of trusted cloud, this paper presents a defensive system model based on trust. Taking the diversity of users' behavioral evidences into account, in order to conduct a more scientific and accurate trust evaluation, our system introduces FAHP to achieve the quantitative assessment of behavioral trust; considering the huge losses caused by viruses to the user, this system model uses multiple detection engines to conduct a comprehensive inspection of suspicious files. Experiments show that our model can not only effectively monitor and assess users' behaviors, but also provide users with security and protection services to achieve a two-way defense.

References

1. Li, H., Li, H.: The key technology and realization of trusted cloud security. Posts & Telecom press, Beijing (2010)
2. Lin, C., Peng, X.-h.: Research on trustworthy networks. Chinese Journal of Computer 28(5), 751–758 (2005)
3. Song, S.S., Hwang, K., Macwan, M.: Fuzzy trust integration for security enforcement in grid computing. In: Jin, H., Gao, G.R., Xu, Z., Chen, H. (eds.) NPC 2004. LNCS, vol. 3222, pp. 9–21. Springer, Heidelberg (2004)
4. Zhou, R.-F., Hwang, K.: Power-Trust: A robust and scalable reputation system for trusted Peer-to-Peer computing. IEEE Transactions on Parallel and Distributed Systems 18(4), 460–473 (2007)
5. Jameel, H.: A trust model for ubiquitous systems based on vectors of trust values. In: The 7th IEEE Int'1 Symp. on Multimedia, pp. 674–679. IEEE Computer Society Press, Washington (2005)
6. Li, X.-y., Gui, X.-l.: Trust quantitative model with multiple decision factors in trusted network. Chinese Journal of Computer 32(3), 405–415 (2009)
7. A white paper of the network defense solutions of Secure Cloud from Trend Micro, http://www.chinacloud.cn/show.aspx?id=339&cid=29
8. Lin, C., Tian, L.-q., Wang, Y.-z.: Research on user behavior trust in trustworthy network. Journal of Computer Research and Development 45(12), 2033–2043 (2008)
9. Ji, T.-g., Tian, L.-q., Hu, Z.-x.: AHP-based user behavior evaluation method in trustworthy network. Computer Engineering and Applications 43(19), 123–126 (2007)
10. Wang, L.-f., Xu, S.-b.: The introduction of AHP. China Renmin University press, Beijing (1990)
11. Liu, L., Wang, H.-q., Liang, Y.: Evaluation method of service-level network security situation based on fuzzy analytic hierarchy process. Journal of Computer Applications 29(9), 2327–2331 (2009)
12. A test report of anti-virus softwares, http://www.av-comparatives.org/images/stories/test/ondret/avc_od_aug2010cns.pdf