VIBRANT TRUST MODEL FOR USER CONSENT

P.Monalisa,
Research Scholar,
Research Department of Computer Science,
Indo-American College,
Cheyyar– 604 407, India.

M.Sasikumar,
Assistant Professor,
PG and Research Department of Computer Science,
Indo-American College,
Cheyyar– 604 407, India

Abstract: Development of authorization mechanisms for secure information access by a large community of users in an open environment is an important problem in the ever-growing Internet world. In this paper we propose a vibrant dynamic trust model for user authorization, rooted in findings from social science. Unlike most existing computational trust models, this model distinguishes trusting belief in integrity from that in competence in different contexts and accounts for subjectivity in the evaluation of a particular trustee by different trusters. Simulation studies were conducted to compare the performance of the proposed integrity belief model with other trust models from the literature for different user behavior patterns. Experiments show that the proposed model achieves higher performance than other models especially in predicting the behavior of unstable users.

Keywords : Trust model , Integrity trust ,Competence trust, network users

I.INTRODUCTION

The everyday increasing wealth of information available online has made secure information access mechanisms an indispensable part of information systems today. The mainstream research efforts for user authorization mechanisms in environments where a potential user's permission set is not predefined mostly focus on role-based access control (RBAC), which divides the authorization process into the role-permission and user-role assignment. RBAC in modern systems uses digital identity as evidence about a user to grant access to resources the user is entitled to.

However, holding evidence does not necessarily certify a user's good behavior. For example, when a credit card company is deciding whether to issue a credit card to an individual, it does not only require evidence such as social security number and home address, but also checks the credit score, representing the belief about the applicant, formed based on previous behavior. Such belief, which we call dynamic trusting belief, can be used to measure the possibility that a user will not conduct harmful actions. In this work, we propose a vibrant trust model for user authorization. Mechanisms for building trusting belief using the first-hand (direct experience) as well as second-hand information (recommendation and reputation) are integrated into the model.

II. LITERATURE REVIEW

The problem of establishing and maintaining dynamic trust has attracted many research efforts. One of the first attempts trying to formalize trust in computer science was made by Marsh. The model introduced the concepts widely used by other researchers such as context and situational trust. Many existing reputation models and security mechanisms rely on a social network structure.

Pujol et al. propose an approach to extract reputation from the social network topology that encodes reputation information.

Walter et al. propose a dynamic trust model for social networks, based on the concept of feedback centrality. The model, which enables computing trust between two disconnected nodes in the network through their neighbor nodes, is suitable for application to recommender systems.

Lang proposes a trust model for access control in P2P networks, based on the assumption of transitivity of trust in social networks, where a simple mathematical model based on fuzzy set membership is used to calculate the trustworthiness of each node in a trust graph symbolizing integrations between network nodes.

Similarly, Long et al. propose a Bayesain reputation calculation model for nodes in a P2P network, based on the history of interactions between nodes. Wang et. Al. propose a

simple trust model for P2P networks, which combines the local trust from a node's experience with the recommendation of other nodes to calculate global trust. The model does not take the time of feedback into consideration, which causes the model to fail in the case of nodes with changing behavior. Reliance on a social network structure limits wide applicability of the mentioned approaches, especially for user authorization. FC Trust uses transaction density and similarity to calculate a measure of credibility of each recommender in a P2P network. Its main disadvantages are that it has to retrieve all transactions within a certain time period to calculate trust, which imposes a big performance penalty, and that it does not distinguish between recent and old transactions. SF Trust is a double trust metric model for unstructured P2P networks, separating service trust from feedback trust. Its use of a static weight for combining local and recommendation trust fails to capture node specific behavior.

Das et al. propose a dynamic trust computation model for secure communication in multi-agent systems, integrating parameters like feedback credibility, agent similarity, and direct/indirect trust/recent/historical trust into trust computation. Matt et al. introduce a method for modeling the trust of a given agent in a multi-agent system by combining statistical information regarding the past behavior of the agent with the agent's expected future behavior.

A distributed personalized reputation management approach for e-commerce is proposed by Yu et al. The authors adopt ideas from Dempster-Shafer theory of evidence to represent and evaluate reputation. If two principals "a" and "b" have direct interactions, b evaluates as reputation based on the ratings of these interactions. Otherwise, b queries a Trust Net for other principals' local beliefs about a. The reputation of "a" is com-puted based on the gathered local beliefs using Dempster-Shafer theory.

Sabater and Sierra propose a reputation model called the Regret system for gregarious societies. The authors assume that a principal owns a set of socio grams describing the social relations in the environment along individual, social and ontological dimensions. The performance highly depends on the underlying socio grams, although how to build socio grams is not discussed.

Skopik et al. propose a dynamic trust model for complex service-oriented architectures based on fuzzy logic. Zhu et al. introduce a dynamic role based access control model for grid computing. The mod-el determines authorization for a specific user based on its role, task and the context, where the authorization decision is updated dynamically by a monitoring module keeping track of user attributes, service attributes and the environment.

Fan et al. propose a similar trust model for grid computing, which focuses on the dynamic change of roles of services. Liu et al. propose a Bayesian trust evaluation model for dynamic authorization in a federation environment, where the only context information is the domain from which authorization is requested.

Ma et al. propose a genetic algorithm for evaluating trust in distributed applications. Nagarajan et al. propose a security model for trusted platform based services based on evaluation of past evidence with an exponential time decay function. The model evaluates trust separately for each property of each component of a platform, similar to the consideration of competence trust in our proposed model Although these approaches integrate context into trust computation, their application is limited to specific domains different from the one considered in our work.

III.OVERVIEW OF THE TRUST MODEL

The trust model we propose in this paper distinguishes integrity trust from competence trust. Competence trust is the trusting belief in a trustee's ability or expertise to per-form certain tasks in a specific situation. Integrity trust is the belief that a trustee is honest and acts in favor of the truster. Integrity and benevolence in social trust models are combined together. Predictability is attached to a competence or integrity belief as a secondary measure. The elements of the model environment, as seen in Figure 1, include two main types of actors, namely trusters and trustees, a database of trust information, and different con-texts, which depend on the concerns of a truster and the competence of a trustee. let us assume that buyer B needs to decide whether to authorize seller S to charge his credit card for an item I (authorize access to his credit card / contact information).

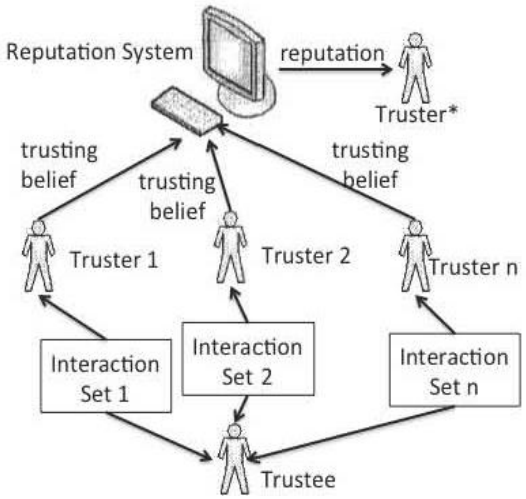


Figure 1: Elements of Model Environment

The elements of the model in this case are:

- Trusters are the buyers registered to the auction site.
- Trustees are the sellers registered to the auction site.
- The context states how important for B the shipping, packaging and item quality competences of S for item I are. It also states how important for B the integrity of S is for this transaction.

B can gather trust information about S from a database maintained by the site or a trusted third party. This in-formation includes the ratings that S received from buy-ers (including B's previous ratings, if any) for competence in shipping, packaging and quality of I as well as S's integrity. It also includes the ratings of buyers (including B) for sellers other than S in different contexts and ratings of S for different items. Trust evaluation is recorded in the database when a buyer rates a transaction with a seller on the site.

Operations Defined on Trust Model

This section presents the operations defined on the trust model. The notations in Table 1 are used for presentation. The notation with superscript v is the value of a belief. The one with superscript p is the associated predictability.

Direct trust for competence denoted by $DTC^{v}_{t1}\rightarrow(c)$ is null, if t, has not interacted with u_1 in context c. Direct trust for integrity denoted by $DTI^{v}_{t1}\rightarrow$ (c) is null if t_1 had no direct experience with u_1 before. Otherwise, it is a real number in the range of [0, 1]. Competence reputation denoted by $RC^{v}_{u1}\rightarrow$,i(c) is null, if no truster knows about u_1 in context c. Integrity reputation denoted by Rl^{v}_{u1} is null, if no trusters interacted with u_1 before. Otherwise, they are real numbers in the range of [0,1]. Reputation is an aggregation of trust beliefs from different trusters. Details of competence and integrity reputation are presented below.

TRUST MODEL NOTATION

$TC^{v}_{t_1\rightarrow u_1}(c)$, $TC^{p}_{t_1\rightarrow u_1}(c)$:	t_1's initial or continuous trusting belief in u_1's competence in context c.
$DTC^{v}_{t_1\rightarrow u_1}(c)$, $DTC^{p}_{t_1\rightarrow u_1}(c)$:	t_1's competence belief about u_1 in c based on direct experience (called direct com-petence trust).
$RC^{v}_{u_1}(c)$, $RC^{p}_{u_1}(c)$:	u_1's competence reputation in context c.
$TI^{v}_{t_1\rightarrow u_1}$, $TI^{p}_{t_1\rightarrow u_1}$:	t_1's initial or continuous trusting belief in u_1's integrity.
$DTI^{v}_{t_1\rightarrow u_1}$, $DTI^{p}_{t_1\rightarrow u_1}$:	t_1's integrity belief about u_1 based on direct experience (direct integrity trust).
$RI^{v}_{u_1}$, $RI^{p}_{u_1}$:	u_1's integrity reputation.

Table 1: Trust Model Notation

IV.EXPERIMENTAL EVALUATIONS

Secured Trust's performance and shows its effectiveness under different adversarial strategy. We have carried out our experiment to achieve four main objectives. Firstly, we evaluate its accuracy in terms of trust computation in the presence of malicious agents under two settings. The second experiment shows how quickly it adapts to strategically oscillating behavior. In the third set of experiments we demonstrate the robustness of Secured Trust compared to other existing trust models under different scenarios. Lastly, we show its effectiveness under the load balancing scheme.

Comparison with Other Trust Model

In this set of experiments we will demonstrate the efficiency of Secured Trust against other existing trust models. In these experiments an agent first computes and compares the trust values of the responding agents (i.e., agents who respond to a transaction request) and chooses the agent with the highest trust value for interaction. A transaction is successful if the participating agent is cooperative i.e., if it is a good agent. In all the experiments, we compute STR as the evaluation criterion under different scenarios. The experiment proceeds in iterations where in each iteration each agent in the system initiates one transaction.

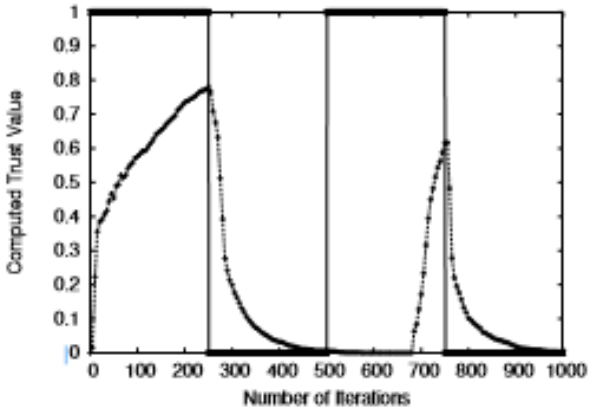


Figure 2.Effectiveness against dynamic personality

We have discarded the transactions initiated by malicious agents from the calculation of STR. We execute a total of 100 iterations in one experiment and compute the average STR. Since the responders to a transaction request are generated at random, we take the mean (along with the 95% confidence interval) of 30 experiments for each scenario. We compare our model with SFTrus , FCTrust, P2P recommendation trust model (for short we will use Reco Trust), trust model of users' behavior (for short we will use User-Trust), dynamic trust model for multi agent systems (for short we will use MAS-Trust) and Peer Trust.

First, we calculate STR against the variation of percentage of malicious agents, malicious per while keeping malicious res to 100% and collusion to 0%. As from Figure 2. We see that both Secured Trust and Peer Trust show superiority over the remaining trust models as the amount of malicious agents in the network increase beyond 40%. Due to the ease of accessibility, networks today are home to a significantly large number of malicious agents, especially the internet holds great threats as it teems with malicious agents (in the form of botnets). In other words, threats and risks are implicitly increasing as network applications are widening. So, in such networks Secured Trust would be the best option. In the next experiment we want to observe the impact of collusion on STR.

So, for this experiment we set malicious per to 60% because as the number of malicious agents increase their collusive impact becomes greater. We also set malicious res to 100%. Figure 3 represents the computed STR against collusion. Due to the experimental randomness, the gradient of the curves may vary from experiment to experiment. In Figure 3, we see that SFTrust, MAS-Trust and User-Trust have negative gradient so in their case STR is actually decreasing as collusive group size is increasing. The remaining four trust models remain unaffected by collusion but we see that again, Secured Trust and Peer Trust show superiority over others.

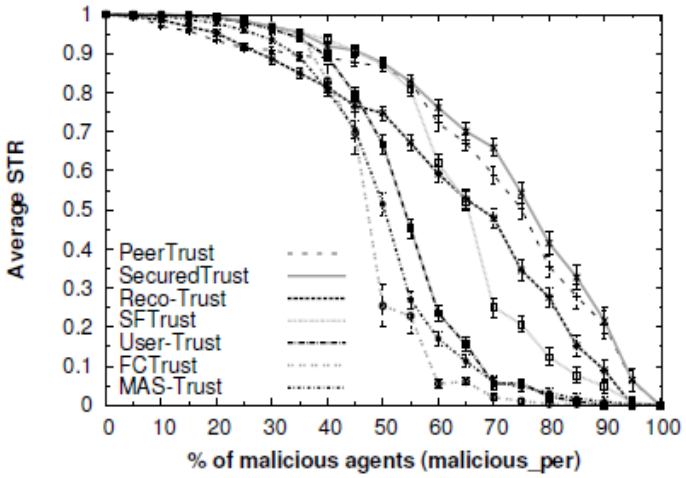


Figure 3: Comparing Secured Trust with other models in terms of average STR with 95% confidence interval

from agents with low feedback credibility as a result they have no impact on STR. The low credibility itself results from the personalized similarity measure. In order to attain high credibility malicious agents would have to provide honest feedback which goes against their true nature.

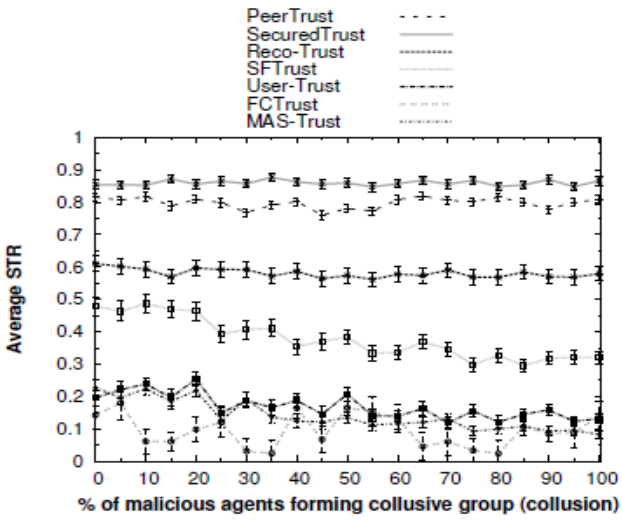


Figure 4: Comparing Secured Trust with other existing trust models in terms of average STR with 95% confidence interval

In the third experiment we analyze the impact of *malicious res* on STR. As we saw in Figure 4 that the malicious agents tend to fool other agents by oscillating between good and malicious nature. In this experiment we test two scenarios with

malicious per set to 40% and 60% respectively while *collusion* is set to 0% in both the cases. Figure 4 represents the computed STR against *malicious res*.

From the figures we see that Secured Trust out performs all other trust models significantly and in these cases Peer Trust suffers the most. This is because our model keeps track of sudden rise and fall of trust by agents and penalizes any agent showing frequent trust fluctuations. While other models fail to identify the strategic alternations made by malicious agents, our model quickly distinguishes such alternations through our deviation reliability metric. Thus, Secured Trust can successfully restrain strategically altering behavior of malicious agents.

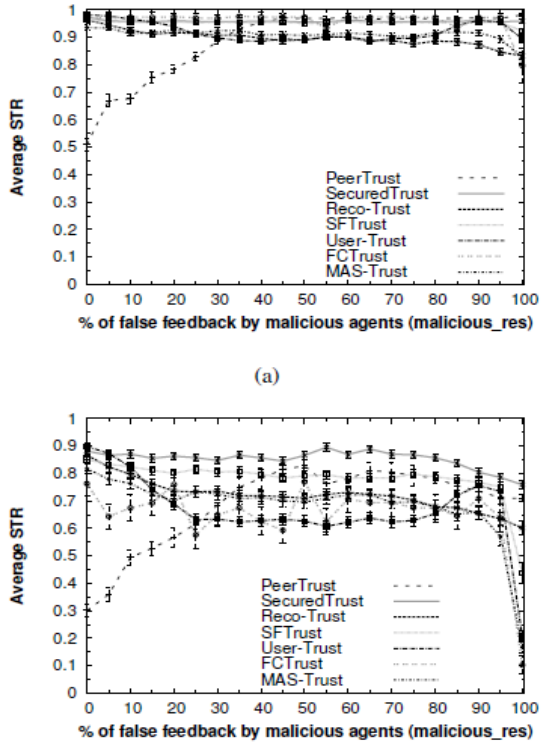


Figure 5: Comparing Secured Trust with other existing trust models in terms of average STR with 95% confidence interval against malicious res (a) 40% malicious agents (b) 60% malicious agents

In the next experiment we determine the number of times malicious agents are selected as service providers in the presence of oscillating malicious behavior. Here we run the experiment for a total of 500 iterations with *malicious res*set to 50%, *malicious per* set to 40% and *collusion* set to 0%. However, we divide the 500 iterations into four equal slots, so each slot contains 125 iterations. Malicious agents oscillate between good and malicious nature from one slot to the next starting with good nature. Then we compute the number of times malicious agents are selected as service providers to transactions initiated by only good agents. From Figure 5 we see that in the initial slot malicious agents are selected numerous times. This is understandable because in the first slot they start off by behaving good so there is no reason to reject them, but in the following slots this number should decline as we now know their true nature. We see that our trust model performs best in isolating the malicious agents and thus reducing unauthentic transactions compared to other models. The reason behind our model's superiority is that we keep track of sudden rise and fall of trust with the intent to heavily punish any agents showing such trust fluctuations. Finally, we compare the computation time required by the different trust models.

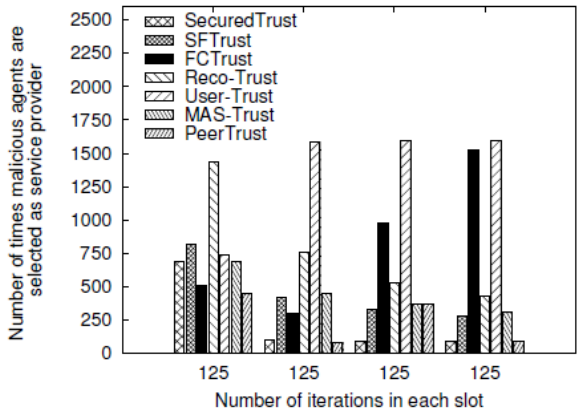


Figure 6: Comparing Secured Trust with other existing trust models in terms of the number of times malicious agents are selected as service providers.

For this purpose we compute the amount of times it takes for the trust models to execute 200 iterations with *malicious per* set to 50%, *collusion* set to 0% and *malicious res* set to 100%. We take the average of 30 runs. From Figure 6 we see that PeerTrust requires the largest amount of time while FCTrust requires the lowest. Our trust model requires on average 1.2 seconds to execute 200 iterations which is slightly higher than some of the remaining trust models. This is understandable as we have considered more components compared to the other trust models. For example we have considered sudden rise and fall of trust as well as historical trend of agent behavior all of which are not considered by other models. As a result these trust models fail to effectively filter out malicious agents when they start to show oscillating behaviors. So, we are sacrificing a very small amount of computational overhead for the sack of better resilience.

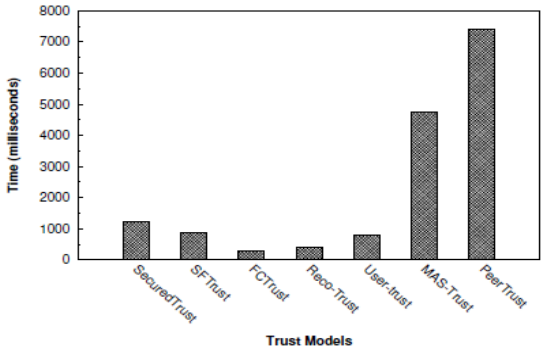
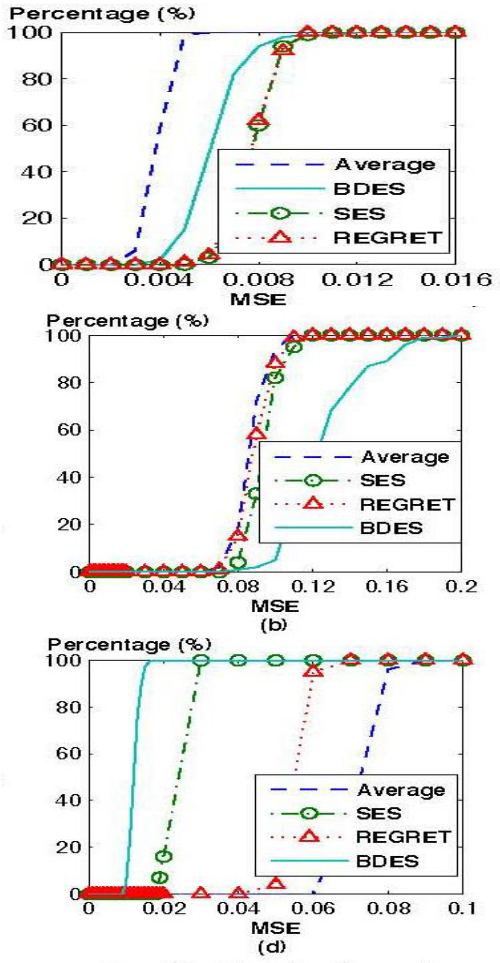


Figure 7: Comparing Secured Trust with other existing trust models in terms of the computational time required to execute 200 iterations

V.RESULTS AND OBSERVATIONS

When the trustee has the stable behavior pattern, the Average algorithm outperforms the other algorithms in terms of MSE. Its MSEs range from 0.002 to 0.01. Around 99% of them are less than 0.005.



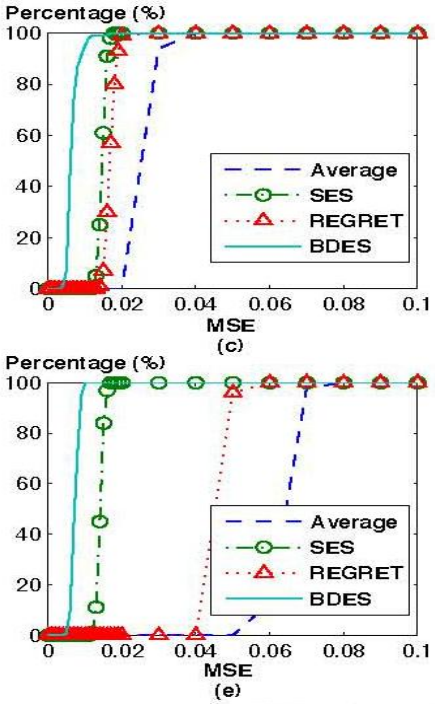


Figure 8: Distribution of mean squared error (MSE) for (a) Stable (b) Random (C) Trend (d) Jumping and (e) Two phase behavior patterns

The REGRET and SES algorithms have almost the same performance, which is worse than that of the BDES algorithm. Ninety percent of the MSEs of the REGRET and SES algorithms are less than 0.009, while the same percentage of MSEs of the BDES algorithm are less than 0.0078. The BDES algorithm introduces larger MSE than the other three algorithms when the trustee has the random behavior pattern. The MSEs range from 0.07 to 0.20. Ninety percent of them are less than 0.16.

The MSEs of the other three algorithms are very close. All of them are in the range of 0.06 to 0.12. BDES performs better than the other algorithms in terms of introducing less MSE when the trustee has the trend behavior pattern. Its smallest MSE is about 0.005. Ninety nine percent of its MSEs are less than 0.012, which is the smallest one among all the MSEs introduced by the other algorithms. The Average algorithm has the worst performance. Its MSEs are in the range of 0.02 to 0.04, 94% of them are less than 0.03. The SES algorithm performs slightly better than the REGRET algorithm. Its MSEs range from 0.012 to 0.018, while 99% of the MSEs of REGRET are in the range of 0.014 to 0.02.

When the trustee has the jumping or two-phase behavior pattern, the BDES algorithm has much better performance than the other algorithms. Even its largest MSE is smaller than the smallest one introduced by the other algorithms. For a trustee with the jumping behavior pattern, the ranges of the MSEs are 0.009 to 0.017 for the BDES algorithm, 0.018 to 0.03 for the SES algorithm, 0.04 to 0.07 for the REGRET algorithm, and 0.06 to 0.09 for the Average algorithm. For a trustee with the two-phase behavior pattern, the corresponding ranges are 0.004 to 0.001, 0.012 to 0.017, 0.04 to 0.06, and 0.05 to 0.08, respectively.

AVERAGE MSE FOR EACH BEHAVIOR PATTERN

	Random	Stable	Trend	Jumping	Two-Phase
Average	0.086069	0.0037669	0.026811	0.072545	0.063554
SES	0.093301	0.007613	0.014744	0.021522	0.014272
REGRET	0.08932	0.0075901	0.016907	0.055423	0.046255
BDES	0.12558	0.0056795	0.0062433	0.012282	0.0074293

Table 2: Average MSE for each behavior pattern

VI.CONCLUSION

In this paper, presented a vibrant trust model for user authorization. This model is rooted in findings from social science, and is not limited to trusting belief as most computational methods are. We presented a representation of context and functions that relate different contexts, enabling building of trusting belief using cross-context information. The proposed dynamic trust model enables automated trust management that mimics trusting behaviors in society, such as selecting a corporate partner, forming a coalition, or choosing negotiation protocols or strategies in e-commerce. The formalization of trust helps in designing algorithms to choose reliable resources in peer-to-peer systems, developing secure protocols for ad hoc networks and detecting deceptive agents in a virtual community. Experiments in a simulated trust environment show that the proposed integrity trust model performs better than other major trust models in predicting the behavior of users whose actions change based on certain patterns over time.

VII.REFERENCE

[1] G.R. Barnes and P.B. Cerrito, "A mathematical model for interpersonal relationships in social networks," *Social Networks*, vol. 20, no. 2, pp. 179- 196, 1998.

[2] R Brent, *Algorithms for Minimization Without Derivatives*. Englewood Cliffs, NJ: Prentice-Hall, 1973.

[3] A. Das, and M.M. Islam. "SecuredTru.st: a dynamic trust computation model for secured communication in multiagent systems," IEEE Trans. Dependable Sec. Comput., vol 9, no. 2, pp. 261-274, 2012.

[4] C. Dellarocas, "Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior," *in Proc. 2nd ACM Conference on Electronic Commerce*, 2000, pp. 150-157.

[5] L. Fan et al., "A grid authorization mechanism with dynamic role based on trust model," *Journal of Computational Information Systems*, vol. 8, no. 12, pp. 5077-5084, 2012.

[6] T. Grandison and M. Sloman, "A survey of trust in internet applications," *IEEE Communications Surveys*, vol. 3, no. 4, pp. 2-16, 2000.

[7] J.D. Hamilton, *Time Series Analysis*. Princeton, NJ: Princeton University Press, 1994.

[8] J. Hu, Q. Wu, and B. Thou, "FCTrust A Robust and Efficient Feedback Credibility- Based Distributed P2P Trust Model," *In Proc IEEE Ninth Intl Conf. Young Computer Scientists* (ICYCS '08), 2008, pp. 1963-1968.

[9] B. Lang, "A Computational Trust Model for Access Control in P2P," *Science China Information Sciences*, vol. 53, no. 5, pp. 896-910, May, 2010.

[10] C. Liu and L. Liu, "A trust evaluation model for dynamic authorization," *In Proc. International Conference on Computational Intelligence and Software Engineering* (CiSE), 2010, pp. 1-4.