Received: August 12, 2023. Revised: October 12, 2023.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Gaussian Flower Pollination Optimization Based Trusted Service Selection for Cloud Computing Environment

S. Priya[1]* **R. S. Ponmagal[1]**

[1]*Department of Computing Technologies, SRM Institute of Science and Technology, Kattankulathur-603203, India*
* Corresponding author's Email: *priya03phd@yahoo.com*

Abstract: As cloud computing is a combination of variety of technologies, there are a number of security concerns need to be addressed. To protect from various attacks and to improve the security of cloud users, it is required to analyze the trust relationships among the cloud resources and tasks. Most of the trust management systems did not consider the significance of interactions, which affects the correctness of trust evaluations. This paper proposes Gaussian flower pollination optimization (GFPO) based trusted service selection (TSS) technique. In this technique, each cloud service provider's reputation values are computed from the trust value of its servers. The individual trust values of cloud servers are determined based on interaction success rate (ISR), service success index (SSI) and service response time (SRT) parameters. The optimal weight values of these parameters are adaptively determined using GFPO algorithm. During the data transfer from server to the user, the user checks the server's trust value and ignores the response if its trust value is low. Experimental results from Cloudsim show that GFPO-TSS minimizes the computation overhead and access delay, maximizes the accuracy and success rate. GFPO-TSS has 45% reduced computation cost, 10% increased data correctness.

Keywords: Cloud computing, Trust evaluation, Cloud service provider (CSP), Gaussian flower pollination optimization (GFPO).

1. Introduction

An open standard architecture called cloud computing enables global computing and boasts request-oriented access to a collection of reconfigurable computing hardware. An innovative vision of cutting-edge computing like virtualization and service-oriented framework are crucial to this hopeful outlook. It offers all of its resources as administrative services, including storing, computing, and correspondence. The combination of capabilities and development innovations is unique. It provides flexible and dynamic basis and assessing with little administrative effort from expert co-ops [1].

Cloud computing has emerged as a main application at the enterprise level [2]. Since a cloud may contain different set of resources with varying users in difficult scenarios, proper access control to these resources is essential. The concept of trust will be compared to several relationships among things involved in a social process. These relationships often involve two parties: the specialist cooperative is the trustor, and the party expecting the trustor's actions, is the trustee. The basis for building trust hinges on the knowledge or experiences obtained from previous drug relationships. There is an explicit need for analysis on trust-based security within the cloud environment, since in any relationship, trust comes before approval [3].

Using cloud computing is a contemporary way to access and use computing resources on the Internet, however there are some security risks and vulnerabilities compared to using the traditional Internet, including issues with confidentiality, integrity, and availability. The cloud computing has also raised new issues, such as the possibility of migrating to and storing data in other countries with different legal systems [4]. Even if it had remained within the country's borders, there are many types of rules that would have been encountered if they had been partially or entirely divulged. Moreover, there are chances that the information may be shared

among various outsiders, which could affect the security. Protecting the integrity of the data is really expensive, when the volume of data is high. Apart from this, the stored data can be altered or corrupted [5].

As cloud computing can include a variety of technologies, including networks, databases, operating systems, resource scheduling, concurrent control, transaction management, virtualization, and memory management, there are a number of security concerns that are problems. All of these are crucial for a cloud service provider since they guarantee that consumers won't experience data loss or theft, which can be highly costly depending on how sensitive the data is stored in the cloud. The malicious user may even pose as a legal user in order to infect the cloud and users [6].

Using access controls is one strategy for safeguarding the confidentiality of data kept in the cloud. Over the years, numerous access control methods have been put forth in the literature. A well-known access control architecture, role-based access control (RBAC), can make security administration easier, especially in complex systems. Roles are used in RBAC to link users to resource rights. Instead of giving permissions to specific users, roles are assigned to users, and only those who have been granted membership in those roles can access the permissions linked to those roles and, in turn, the resources [7].

In conventional systems, a central authority with administrative control over all the system's resources will typically specify and enforce access control policies. However, in a distributed system like the cloud, where data may be kept in dispersed data centres that are not under the control of a single authority, such a central authority may not exist. Although the access control policies may occasionally be defined in a centralized manner by the cloud provider authority, there may be additional authorities dispersed throughout the cloud system that are responsible for enforcing these access restrictions. Therefore, it would be necessary to have faith in these authorities to appropriately establish and enforce access control measures [8].

In the quickly developing world of cloud computing, the security of very valuable client/user resources is a critical concern. The concept of multi-cloud architecture was designed and is seen as advantageous for enhancing resource security and privacy. Among the users and service providers, there is a management platform in this architecture. It is in charge of resource management, scheduling, resource allocation, and login features. This centralized platform enables administration tasks to be scheduled in accordance with user needs. A substantial transition toward a multi-cloud environment has recently taken place. This change was made for security reasons, with the thought that one resource may be separated and kept by various cloud service providers. The sharing algorithm known as the Shamir Secret is seen to be helpful in light of these factors. It implies that even if a resource leaks in part, the remaining portions and hence the overall resources are safe. By decreasing the intended trust value for the user-service provider, the multi-cloud provides the advantage of improved security when compared to a single cloud [9].

This paper is organized as follows: In section 2, works related to trust methods on cloud computing are presented. In section 3, rresearch gaps and problem identification are presented. In section 4, the proposed methodology is discussed in detail. Section 5 presents the experimental results and section 6 presents the conclusion of the work.

2. Related works

In this section, literature review on existing trust based attack detection techniques, trust based resource allocation techniques and trust based access control techniques are presented. The research gaps associated with these works are summarized followed by the problem statement.

2.1 Trust based attack detection

The trusted anonymous lightweight attacker detection (TALAD) method is provided by Rajan and Naganathan [10] to recognize and defend malicious nodes in a cloud based WSN-IoT system. Subject to a specified path length constraint, the TALAD technique constructs a routing path to the cloud with highly trustworthy nodes. The original identity of each node is concealed from the other nodes in the network by using the binomial algebraic theorem to create false node identities. The original identities of the nodes are revealed if only the forward key and the reverse key string are matched. A context-free grammar rule is used to map the forward and backward keys. TALAD reliably prevents invasions even when a significant portion of the network fails to forward packets.

Alshammari et al. [11] have designed a trust model system which contains various trust conditions to estimate the trustworthiness of cloud services. Only the responses with good trust scores are accepted from the cloud service providers. The customer’s trust values are computed such that both there is a balance between customer trust and provider’s service.

Soleymani et al. [12] have developed a trust management method for multicloud systems. The trust values of service providers are computed from the subjective and objective trust metrics. This system identifies the fake responses received from others. The confidence level has been estimated using Fuzzy logic. Tang et al. [13] have developed a trust-based route creation system for secure cooperation among the cars. It combines both direct and indirect trust estimation values to compute the trust value of the cooperating vehicles.

Alshammari et al. [14] have presented trust architecture for avoiding various attacks. By detecting malicious and doubtful actions using trust algorithms, it can recognize on/off and collusion attacks. This technology provides effective protection to cloud services. In conclusion, the findings demonstrate that the suggested trust model yields superior security by minimizing the security risks among the cloud data owners and operators.

The study of Sarkar et al [15] has focused on domain related issues of cloud computing systems. It also discussed the requirements for zero-trust architecture and the challenges of cloud systems.

2.2 Trust based resource allocation and scheduling

A trust-based scheduling technique was presented by Yang et al [16]. First, they have formulated the cloud workflow scheduling, and then proposed the algorithm that corresponds to it. In this algorithm, the trustworthy computation service and the trustworthy storage service are chosen based on the S-PSO method and tree search heuristic method, respectively.

Yang et al [17] have designed trust based resource allocation technique, where trust is maintained between customers and service providers. In this technique, the allocation decision fetches the best solution that will provide the maximum trust values, by applying Genetic algorithm.

2.3 Trust based access control

In [18], a subjective trust model is designed depending on the characteristics of users and providers by applying fuzzy logic. Performance and elasticity are taken into consideration while evaluating the resource's level of trustworthiness. Workload and response time are the factors used to determine performance. We also considered scalability, availability, security, and usability while assessing elasticity. To evaluate the trust value of users, fuzzy C-means clustering is applied to parameters like bad requests, fake requests, unauthorized requests, and total requests.

In order to solve this problem, Paul and Raj [19] created a trust-based access control model based on user and server properties. It includes Cyclic shift transposition algorithm for data encryption. This model uses direct trust degrees to assign cloud users trust values. The interaction success and failure rates, service satisfaction index, and the amount of dishonesty are used to evaluate the direct trust degree. The positional each user's access control policy is altered in accordance with his level of trust. Another acceptable server will be chosen if the server doesn't achieve the required degree of confidence.

Zhou et al. [20] have developed trust model based on role based access control (RBAC) to promote the data storage security in cloud network. The trust model monitors the roles of owners and users. In determining whether a position is trustworthy, the trust models consider the inheritance and hierarchy metrics. The authors demonstrate how trust models can be integrated into a system that use cryptographic RBAC techniques by presenting the architecture of a trust-based cloud storage system. Additionally, they took into account real-world application scenarios and showed how trust assessments may be utilized to lower risks and improve the calibre of decision-making.

3. Research gaps and problem identification

Most of the trust management systems did not consider the significance of interactions, which affects the correctness of trust evaluations. When applying for healthcare-related services, advanced security measures are necessary. Numerous scholars have been interested in the problem of cloud service trust, but there are still numerous issues that need to be resolved.

To protect from various attacks and to improve the security of trust models, other reputation attack types of cloud computing systems should be considered. It is required to analyze the trust relationships among the cloud resources and tasks. The resource trust value will assist cloud customers in choosing a cloud provider to handle and store their crucial data. Performance, flexibility, cost, time, and data security are the characteristics that must be combined in order to assess each service provider's level of confidence based on its distinct attributes. A crucial area for research in vehicular cloud computing is security collaboration (VCC). Due to the presence of hostile cars, security cooperation in VCC has become a difficult problem.

There are many meta heuristic algorithms such as Gaussian flower pollination optimization (GFPO)

Table 1. Notations used in this work

Notation	Definition
A_i^t	Positive feedback at time t for i[th] interaction
D_L	Dishonest Level
NF	Number of feedbacks
II	Interaction Importance
μ	Weighting constant
δ	Weighting constant
N_{AV}	Number of Access Violation
T_p	Time of Service Reply
T_q	Time of Service Request
$\Gamma(\delta)$	Gamma term
α^2	Variance of all members
$Gauss(\alpha)$	Gaussian Step Factor
dt_i	Degree of Trust
B^t	Negative feedback at time t
δ	Constant
Z_j^t	Pollen individual position
α	Uniform distribution parameter

algorithm [21], puzzle optimization algorithm (POA) [22], guided pelican algorithm [23], stochastic komodo algorithm [24].

In this work, GFPO is applied which is metaheuristic algorithm that mimics the pollination characteristics of flowers and it includes cross pollination and self-pollination.

The following problems are identified from the research gaps:

- Since cloud computing services are provided through public cloud networks, any unauthorized users can access them.
- The security and privacy of sensitive data on cloud will be questionable due to various issues.
- The on/off attack is common attack in the cloud networks in which the providers gain reputation initially and later become distrustful.

In order to provide solution to these problems, a GFPO-TSS technique is developed.

4. Proposed methodology

This paper proposes GFPO-TSS technique for cloud computing. In this technique, the trust values of cloud servers are determined based on ISR, SSI and SRT parameters. The optimal weight values of these parameters are adaptively determined using GFPO algorithm. During the data transfer from server to the user, the user checks the server's trust value and ignores the response if its trust value is low.

4.1 Trust estimation for cloud server

The trust values of servers are determined from the ISR, SSI and SRT parameters, which are defined below:

4.1.1. Interaction Success Rate (ISR)

By computing the interaction specification of the cloud servers, the trust model determines the ISR which gives an accurate result for confidence level of each user.

The ISR of a cloud user C can be computed as follows [11]:

$$ISR(C) = \sum_{i=1}^{n-1} \frac{A_i^t(C)+P_C}{(A_i^t(C)+P_C)+(B_i^t(C)+N_C)} \quad (1)$$

where A^t and B^t denotes positive and negative feedback at time t, respectively.
P_C and N_C are computed by

$$P_C = \frac{A_i^t(C)^{XII}}{NF} \quad (2)$$

$$N_C = \frac{B_i^t(C)^{XII}}{NF} \quad (3)$$

Where II denote Interaction Importance interaction NF denotes the number of feedbacks.

4.1.2. Number of dishonest attempts

The service provider is able to monitor any rogue user C_i's access rights. Then, using Eq, (4), the dishonest level (D_L) of user C_i is determined [10].

$$D_L(C_i) = \mu N_{AV} \quad (4)$$

where N_{AV} denotes the number of access violations done by the user and μ is a weighting constant.

4.1.3. Service success index (SSI)

When one entity uses numerous services provided by another entity, the operation domain assigns a service satisfaction index (SSI). Following k interactions, the following Eq. outputs the SSI that user C_j assigned to user C_i [10].

$$SSI(C_i, C_j)^{k-1} = \delta . SSI(C_i, C_j)^{k-1} + (1-\delta) . TR_{in}(C_i, C_j)^k \quad (5)$$

Where δ indicates a weighting constant.

4.1.4. Service response timey (SRT)

The service provider is able to monitor any rogue user U_i's access rights. Then, using the following Eq, the SRT of user C_i, (i=1,2….m) is determined.

$$SRT = \frac{\sum_{i=1}^{n}[T_p(C_i) - T_q(C_i)]}{m} \quad (6)$$

where T_p *and* T_q indicate time of service reply and the service request respectively, by users C_i.

4.2 Trust evaluation process

Many servers are selected as possible members to provide the user-demanded services. To evaluate the confidence level of the user-requested services, an initial value is assigned for all these parameters by the users. The trust table is accessed to determine the trust value of the services.

Let *p* servers are selected as members, based on the highest dependability value defined.
The degree of trust dt_i is determined by

$$dt_i = \sum_{i=1}^{n}\sum_{j=1}^{4} wp_j Xp_{j,i} \quad (7)$$

Here the 4 parameters IE, DL, SSI and SRD are considered as $p_{1,i}$,$p_{2,i}$.$p_{3,i}$,$p_{4,i}$

The corresponding optimal weight values wp_1, wp_2, wp_3 and wp_4 are adaptively estimated using GFPO algorithm, such that their sum = 1

4.2.1. Gaussian flower pollination optimization (GFPO) algorithm

It is the metaheuristic algorithm [21] that mimics the pollination characteristics of flowers and it includes cross pollination and self-pollination. In the cross pollination, the flight characteristics of the butterflies follow the levy's flight distribution. In the self-pollination, mature antigens of plants provided to the own or various flowers of similar kinds of plants. The FPO algorithm is described in the following section:

- The constancy of the flower is set as the reproduction probability and it is the ratio of similarity between two flowers.
- The trade-off among the global and local pollination is managed by the conversation probability value.
- In the biological cross pollination, flower's flight follows Levy flight for global pollination. It is represented as:

$$Z_j^{(t+1)} = Z_j^t + Levy(\delta)(Z_j^t - h^*) \quad (8)$$

where $Z_j^{(t+1)}$ and Z_j^t indicates the pollen individual position j in the $t+1$ generation, h^* is the optimal flower position and $Levy$ is the step term that follows $Levy$ distribution. It is represented as:

$$Levy = \frac{\delta\Gamma(\delta)\sin(\pi\delta/2)}{\pi}\frac{1}{S^{1+\delta}} \quad (9)$$

where $\Gamma(\delta)$ is the gamma term. Abiotic self-pollination is considered as the local pollination process and it is given as:

$$Z_j^{(t+1)} = Z_j^t + \alpha(Z_k^t - Z_l^t) \quad (10)$$

where Z_k^t *and* Z_l^t are the two pollen's position in the similar type of plants and α is the uniform distribution parameter.

The Gaussian mutation operator enhances the searching capacity of the optimization. The Gaussian density function is described as:

$$H_{Gaussian(0,\alpha^2)}(\sigma) = \frac{1}{\sqrt{2\pi\alpha^2}} exp\left(-\frac{\alpha^2}{2\alpha^2}\right) \quad (11)$$

where α^2 is the variance for all members of the population.

The random parameter obtained is provided to the Eq. (11) and it is expressed as:

$$Z_j^{(t+1)} = Z_j^t + Gauss(\alpha)(Z_k^t - Z_l^t) \quad (12)$$

where $Gauss(\alpha)$ is the Gaussian step factor and the value of α ranges from 0 to 1.

4.3 Trusted server selection method

In this work, the best and most suitable service provider is chosen using a roulette wheel mechanism. This approach is used to distribute the load among all cloud service providers. The quantity of user requests (in percentage) assigned to each cloud service provider is determined by the following formulae. Each parameter's weight is calculated from the GFPO algorithm. The value of tv_i that is calculated using Eq. (13) is stored in the *tv* array *(tv_1, tv_2, tv_3, . . . , tv_n)*.

$$tv_i = \sum_{i=1}^{n}\sum_{j=1}^{m} wpXp_{j,i} \quad (13)$$

The % of user requests allocated to the server (s_i) is estimated by

Table 2. Experimental settings

Settings	Value
No. Servers	4
No. registered users	10
No. dishonest users	4
No. requested services	10-50
Average size of each request	200 KB to 500 KB
Desired Response Time	DRT = 1000ms=1s
Configuration of VMs	Medium and Large
Maximum On-demand VM Limitation	*Max*VM=10VM

Table 3. Results of computation cost

No of service Requests	FRTM	RAD	GFPO-TSS
10	2750	2175	1270
20	5322	4683	2275
30	7434	6967	4647
40	9234	7889	5307
50	10480	9284	6870

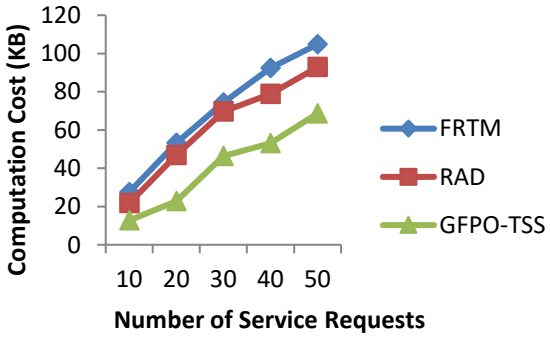

Figure. 2 Computation cost for varying service requests

$$s_i = \frac{tv_i}{\sum_{i=1}^{n} tv_i} \tag{14}$$

The value of S_i is stored in the S array $(s_1, s_2, \ldots, s_n)$. The optimum server is chosen in the previous stage using SP and the roulette wheel mechanism.

4.3.1. Estimating global trust values (GTV)

By aggregating all local trust values, the global trust value (GTV) of each server is obtained. The GTV is estimated from the average of local trust values.

The GTV of server j at time t is given by

$$GTV_j^t\{U\} = \begin{cases} gtv_j^t\{T\} = \frac{\sum_{i=1}^{n} ltv_{i,j}^t(T)}{n} \\ gtv_j^t\{T\} = \frac{\sum_{i=1}^{n} ltv_{i,j}^t(-T)}{n} \\ gtv_j^t\{T\} = 1 - gtv_j^t\{T\} - gtv_j^t\{-T\} \end{cases} \tag{15}$$

Where T, $-T$ and U denote the trust, distrust and Uncertain status of obtained services, respectively, gtv_j^t denotes the global trust values in time t and $(ltv_{i,j}^t)$ denotes the local trust value of server j evaluated by user i at time t.

Then the Reputation of the server is computed by summing the GTVs of all the servers $S_j \in S_j$, i=1,2…m as

$$R_i = \sum_{i=1}^{m} GTV_j \tag{16}$$

If the GTV is less than a threshold for a server, then the reputation of that server is reduced by a factor f. During the data transfer from a server to the user, the user checks the server's reputation and ignores the response if its reputation is low.

5. Experimental results

The proposed GFPO based trusted service selection (GFPO-TSS) technique is implemented in Cloudsim. Table 2 presents the experimental settings used in the implementation.

The attack model includes distributed denial of service (DDoS), malicious and selfish attacks, abuse of cloud services, data loss etc.

5.1 Results and discussion

The performance comparison of GFPO-TS, fuzzy rule-based trust management (FRTM) [12] and reputation attack detector (RAD) [14] techniques have been conducted and the results are presented in this section.

The experiments are conducted by increasing the service requests from 10 to 50. The computation cost, data correctness, service success rate, service access delay and detection accuracy metrics are measured. Table 3 and Fig. 2 show the results of computation cost.

From Fig. 2, it can be seen that GFPO-TSS has 45% reduced computation cost, than FRTM and 37% lesser than RAD.

Table 4 and Fig. 3 show the results of data correctness.

From Fig. 3, it can be seen that the data correctness of GFPO-TSS increases by 10% and 7% when compared to FRTM and RAD, respectively.

Table 4. Results of correctness of data

No of Service Request	FRTM (%)	RAD (%)	GFPO-TSS (%)
10	0.86	0.88	0.91
20	0.83	0.82	0.90
30	0.74	0.79	0.84
40	0.71	0.74	0.81
50	0.68	0.71	0.80

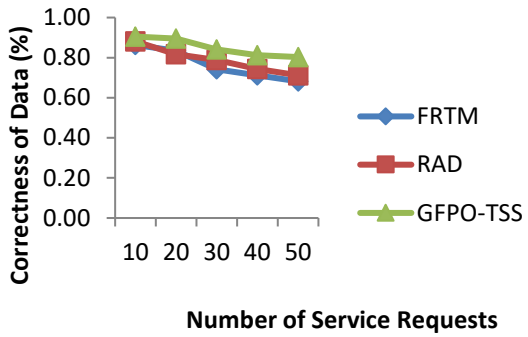

Figure. 3 Correctness of data for varying service requests

Table 5. Results of service success rate

No of Service Requests	FRTM (%)	RAD (%)	GFPO-TSS (%)
10	92.1	96.5	98.1
20	91.7	95.3	97.4
30	91.2	94.8	96.7
40	90.4	94	96.5
50	90.1	93.6	95.8

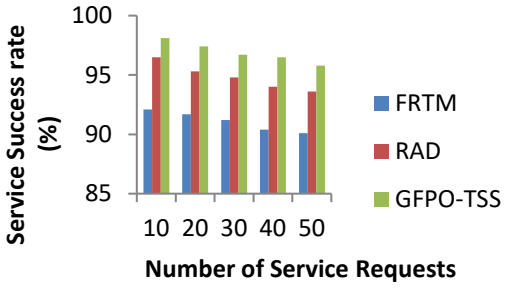

Figure. 4 Service success rate for varying service requests

Table 5 and Fig. 4 show the results of service success rate.

From Fig. 4, it can be seen that the service success rate of GFPO-TSS increases by 6% and 2% when compared to FRTM and RAD, respectively.

Table 6 and Fig. 5 show the results of service access delay.

Fig. 5 shows that GFPO-TSS has 26% and 36% lesser access delay, when compared to FRTM and RAD, respectively.

Table 6. Results of access delay

No of Service Requests	FRTM (sec)	RAD (sec)	GFPO-TSS (sec)
10	1.32	1.75	0.79
20	1.73	1.92	1.17
30	2.04	2.13	1.45
40	2.31	2.62	1.81
50	2.79	3.33	2.55

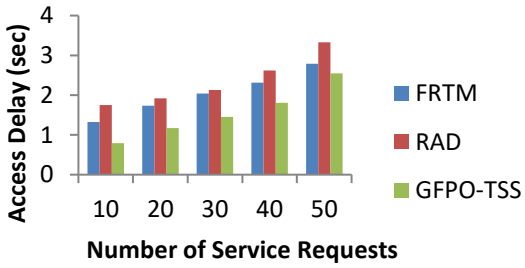

Figure. 5 Access delay for varying service requests

Table 7. Results of detection accuracy

No of Service Requests	FRTM (%)	RAD (%)	GFPO-TSS (%)
10	92.78	93.70	96.70
20	92.25	93.11	96.10
30	92.30	92.59	95.88
40	91.48	92.07	95.34
50	90.74	91.36	94.61

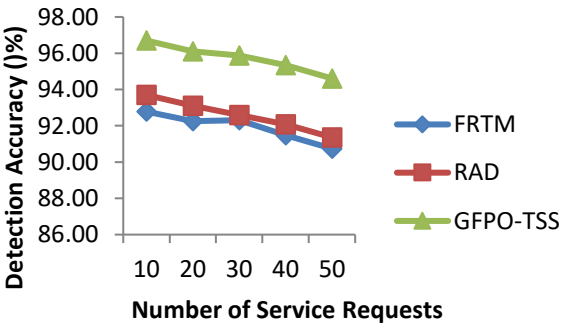

Figure. 6 Detection accuracy for varying service requests

Table 7 and Fig. 6 show the results of detection accuracy.

From Fig. 6, it can be seen that the detection accuracy of GFPO-TSS is 4% is higher than FRTM and 3% higher than RAD.

Table 8 and Fig. 7 show the results of false positive rate.

Table 8. Results of false positive rate

No of Service Requests	FRTM	RAD	GFPO-TSS
10	0.382	0.345	0.279
20	0.423	0.392	0.317
30	0.454	0.413	0.345
40	0.512	0.462	0.381
50	0.569	0.523	0.465

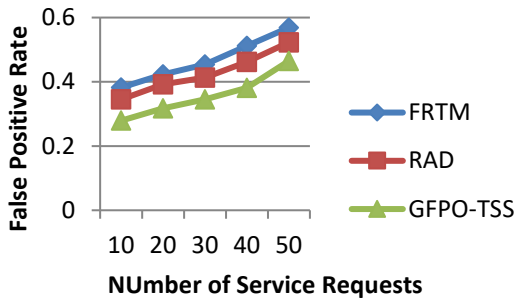

Figure. 7 False positive rate for varying service requests

From Fig. 7, it can be seen that the false positive rate of GFPO-TSS is 24% is lesser than FRTM and 16% lesser than RAD.

6. Conclusion

In this paper, GFPO-TSS technique is proposed. In this technique, each cloud service provider's reputation values are determined from the trust value of its servers. The trust values of cloud servers are determined based on ISR, SSI and SRT parameters. The optimal weight values of these parameters are adaptively determined using GFPO algorithm. During the data transfer from server to the user, the user checks the server's trust value and ignores the response if its trust value is low. Experimental results from Cloudsim show that GFPO-TSS minimizes the computation overhead and access delay, maximizes the accuracy and success rate. GFPO-TSS has 45% reduced computation cost, 10% increased data correctness.

Conflicts of interest

The authors don't have any conflict of Interest

Authors contribution

In this manuscript preparation author 1 prepared the concept and author 2 prepared the implementation and both the authors together corrected the English grammatical errors.

References

[1] M. K. I. Rahmani, M. Shuaib, S. Alam, S. T. Siddiqui, S. Ahmad, S. Bhatia, and A. Mashat, "Blockchain-Based Trust Management Framework for Cloud Computing-Based Internet of Medical Things (IoMT): A Systematic Review", *Hindawi, Computational Intelligence and Neuroscience*, Vol. 2022, Article ID 9766844, 14 pages, 2022.

[2] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility", *Future Generation Computer Systems*, Vol. 25, No. 6, pp. 599–616, 2009.

[3] S. N. Mthunzi, E. Benkhelifa, T. Bosakowski, C. G. Guegan, and M. Barhamgi, "Cloud computing security taxonomy: from an atomistic to a holistic view", *Future Generation Computer Systems*, Vol. 107, pp. 620–644, 2020.

[4] M. S. Mushtaq, M. Y. Mushtaq, M. W. Iqbal, and S. A. Hussain, "Security, integrity, and privacy of cloud computing and big data", *Security and Privacy Trends in Cloud Computing and Big Data*, pp. 19–51, 2022.

[5] X. Li, J. He, and Y. Du, "Trust Based service Optimization selection for cloud computing", *International Journal of Multimedia and Ubiquitous Engineering*, Vol. 10, No. 5, pp. 221–230, 2015.

[6] S. T. Siddiqui, S. Alam, Z. A. Khan, and A. Gupta, "CloudBased E-learning: using cloud computing platform for an effective E-learning", *in Smart Innovations in Communication and Computational Sciences*, Vol. 851, pp. 335–346, 2019.

[7] S. Harbajanka and P. Saxena, "Security issues and trust management in cloud computing", In: *Proc. of the ACM Symposium on Women in Research 2016 - WIR '16*, Indore India, pp. 1–3, 2016.

[8] E. F. Rawashdeh, I. I. Abuqaddom, and A. A. Hudaib, "Trust models for services in cloud environment: A survey", In: *Proc. of the 2018 9th International Conference on Information and Communication Systems (ICICS)*, Nagoya, Japan, pp. 175–180, 2018.

[9] F. N. Nwebonyi, R. Martins, and M. E. Correia, "Reputation Based Security System For Edge Computing", In: *Proc. of ARES 2018: International Conference on Availability, Reliability and Security*, Hamburg, Germany, 2018.

[10] D. A. J. Rajan and E. R. Naganathan, "Trust based anonymous intrusion detection for cloud assisted WSN-IOT", *Global Transitions Proceedings*, Vol. 3, pp. 104–108, 2022.

[11] S. T. Alshammari, A. Albeshri, and K. Alsubhi, "Building a trust model system to avoid cloud services reputation attacks", *Egyptian Informatics Journal*, Vol. 22, pp. 493–503, 2021.

[12] M. Soleymani, N. Abapour, E. Taghizadeh, S. Siadat, and R. Karkehabadi, "Fuzzy Rule-Based Trust Management Model for the Security of Cloud Computing", *Hindawi, Athematical Problems in Engineering*, Vol. 2021, Article ID 6629449, 14 pages, 2021.

[13] Z. Tang, A. Liu, Z. Li, Y. J. Choi, H. Sekiya, and J. Li, "A Trust-Based Model for Security Cooperating in Vehicular Cloud Computing", *Hindawi Publishing Corporation Mobile Information Systems*, Vol. 2016, Article ID 9083608, 22 pages, 2016.

[14] S. T. Alshammari and K. Alsubhi, "Building a Reputation Attack Detector for Effective Trust Evaluation in a Cloud Services Environment", *Appl. Sci.*, Vol. 11, p. 8496, 2021.

[15] S. Sarkar, G. Choudhary, S. K. Shandilya, A. Hussain, and H. Kim, "Security of Zero Trust Networks in Cloud Computing: A Comparative Review", *Sustainability*, Vol. 14, p. 11213, 2022.

[16] Y. L. Yang, X. G. Peng, and J. F. Cao, "Trust-Based Scheduling Strategy for Cloud Workflow Applications", *Informatica*, Vol. 26, No. 1, pp. 159–180, 2015.

[17] J. Yang, H. Zhu, X. Zhu, Y. Liu, L. Liu, and T. Liu, "Resource Allocation Policy Based on Trust in the Multi-Cloud Environment", In: *Proc. of 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Banff, Canada, 2017.

[18] A. Kesarwani and P. M. Khilar, "Development of trust based access control models using fuzzy logic in cloud computing", *Journal of King Saud University – Computer and Information Sciences*, 2019.

[19] N. R. R. Paul and D. P. Raj, "Enhanced Trust Based Access Control for Multi-Cloud Environment", *Computers, Materials & Continua, CMC*, Vol. 69, No. 3, 2021.

[20] L. Zhou, V. Varadharajan, and M. Hitchens, "Trust Enhanced Cryptographic Role-Based Access Control for Secure Cloud Data Storage", *IEEE Transactions on Information Forensics And Security*, Vol. 10, No. 11, 2015.

[21] E. Nabil, "A Modified Flower Pollination Algorithm for Global Optimization", *Expert Systems with Applications*, Vol. 57, pp. 192–203, 2016.

[22] F. A. Zeidabadi and M. Dehghani, "POA: Puzzle Optimization Algorithm", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 1, 2022, doi: 10.22266/ijies2022.0228.25.

[23] P. D. Kusuma and A. L. Prasasti, "Guided Pelican Algorithm", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 6, 2022, doi: 10.22266/ijies2022.1231.18.

[24] P. D. Kusuma and M. Kallist, "Stochastic Komodo Algorithm", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 4, 2022, doi: 10.22266/ijies2022.0831.15.