ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

Trustworthy Cloud Service Discovery for Large Scale applications - An Approach to Cloud and Big Data Environment

[1] T.Vivekanandan, [2]A.Srinivasan
[1][2] Associate Professor, CSE Department, SITAMS, Chittoor

Abstract: Cloud computing emerged as a paradigm to deliver various services through web. This technology enables virtualization to separate physical computing resources in to scalable system of multiple independent computing resources. This facility offers to speed up the operations on-demand and greatly reduces the cost as well. Cloud adopts service oriented architecture to provide massive IT resources in heterogeneous environment. Service oriented computing is a solution for solving large scale applications. These applications utilize various services from heterogeneous resources offered by different service providers. Effective service discovery in cloud is highly complex at different levels of quality of service offered by different providers. Ensuring its accessibility, availability, trustworthiness and relevance matching are challenging tasks. To identify the relevance matching between the published cloud services by a publisher and the requested details by the end user, we proposed Trustworthy cloud service discovery for large scale applications. In this paper, we focused on the above issues for effective service publication and service discovery in the cloud. We have addressed about effective way of discovering the cloud services and accessing the resource

Index Terms — Cloud services, Service publication, Service discovery, Trustworthiness.

INTRODUCTION

Cloud computing is an evolving paradigm that facilitates ubiquitous access to on demand computing environment. This computing environment enables seamless integration of computing resources such as networks, servers, storage, applications, software and services. These resources can be rapidly provisioned with minimum management effort over web. The characteristics like on-demand service, hardware virtualization, broad network access, resource policy, rapid elasticity, Service oriented architecture (SOA), pay-as-you-go model, etc. has put forth the growth in cloud computing[3][14][24]. The effective adoption of service-oriented architecture offers robustness, scalability, extensibility and interoperability in cloud through different models like Infrastructure as service (IaaS), Platform as a service (PaaS), Software as service (SaaS), etc. The services through these models emerged as a solution for huge computational and data requirements of many applications [18].

In the recent times, cloud computing matured to a state where organizations can progress on their effectiveness, technical advancements, and usability in implementing Cloud Computing services[19][21] [22]. This is due to the blending of established technologies such as virtualization, on-demand computing, highly scalable data processing, improved web services, data visualization, huge data storage and backup, high performance computing, and APIs on portable devices and Cloud Computing. As a result, it greatly helps to understand the services offered in the context of existing and value added services. Among the available services, some offer new and innovation based serviced [1][10][25].

The computational intensive application requires massive amount of computing resources and data storage. In cloud computing environment, different types of services offered by service providers are published independently. In a heterogeneous environment like cloud, thousands of service instances [2] will be hosted with large number of different services from different service provides with different levels of quality of service [23].

The main objective of this paper is to search and discovery exact cloud services. This is implemented by generating accurate results for web based end user queries. The another objective is to efficiently access the data from the data resources like huge databases, data warehouses, large file systems like NTFS, FAT32. These results are optimized for large scale applications that need data dynamically at the time of execution.

This paper further includes the filters based on category and description. The end user results depend on the past history and user's feedbacks. The cloud service discoveries depend on the four factors as follows:

The trust worthiness of a resource can be assessed by calculating the rating that depends on the feedback given by end users who have registered through the web user interface. The relevance matching of a document or any other data resources can be assessed based on the three filters which they are category filter, description filter and connective words filter. The access cost is based on the average Round Trip Time (RTT) from regular checks of type's instances; this sampling improves the accuracy of accessibility measure. The availability is evaluated based on probability calculated by using previous history of the resource availability and availability of resource at current instance.

Connective words filter removes English grammatical connecting words from the user query to improve the quality of search. Category filter helps to match data resource with related category, so the resources set that need to be searched will reduce optimally. The description filter tries to locate data resource based on its description given by its publisher so that user can get accurate results from the resource set that is already optimized by other two filters and previous four factors.

II. LITERATURE SURVEY

Currently many cloud service discovery techniques exist for different requirements. String algorithms are the major searching techniques used in current system, string algorithms try to find a place where one or several strings (also called patterns) are found within a larger string or text[5][6][11].

Parhi, Pattanayak, and Patra developed a framework for service description and discovery using ontology based on multi-agent cloud environment [4]. The different search engines such as Google, yahoo, MSN, bing, etc. fail to perform effective the service discovery. To overcome this they proposed to address the non-standardized specification terminology used by different cloud providers [7][8][9].

Liu, Xing, Che, and Bao proposed agent-based cloud computing environment with four agents, which are resource agents (RA), service provider agents (SPA), broker agents (BA) and consumer agents (CA). The agent-based computing is applied for robust service discovery with local and global service discovery capabilities.

Ontology based technique [17][26][20] has been implemented many researchers. In particular, Rajendran and Swamynathan proposed to discover efficient cloud services. To retrieve information about services multi-broker agent based discovery system was adopted with for ontology based cloud service discovery. Due to heterogeneous nature of the cloud systems, client needs to discover the cloud services. The cloud service and service providers may have varied levels of quality. Identifying the appropriate is highly difficult task and also great change for vulnerability. In this paper [5], a model was proposed for trust based reliable cloud discovery [12][15].

Service discovery is a subdivision of resource discovery in such a way that service discovery should be seen as specifically as capability to find specific services such as applications or well defined network services that are not pure abstractions [10][13][16].

III. PROPOSED APPROACH

The proposed system includes the following steps as mentioned below:
1. Service publication.
2. Service discovery
3. Estimation of Trust ranking

Service Publication

Service publisher publishes the service. First, the service publisher needs to register. After the successful registration and login, the publisher is allowed to publish services. For the service publication, the service publisher has to enter the fields like service category, service description and URL of the service. While entering the details like service category and the service description, the publisher has to take care about the words present in those fields. Because, the unknown end user can request for this service.

Publisher is allowed to enter any number of services. But it is the responsibility of the publisher to check for the working of the service published by him. The service publisher has the option for managing the services. i.e., he is able to modify the service details and also delete the services published before

Service Discovery

Service discovery is done by the end user. End user also needs to register first. After successful registration and login, the end user is provided with the web user interface, where he has to enter the fields like service category and the service descriptions of the service he is looking for. Then, the matching of the fields entered by the end user and the fields that are present in the profile of the service is done. If any service matches, then the URL of the matched service is displayed in the web user interface of the end user.

Along with the URL, the Effective Service Discovery is provided to the end user. If its value is less than the threshold set by the end user, then he has to reenter the words in the fields of service category and service description in different combinations, until he can get the matching degree equal to or greater than the threshold limit.

In the service discovery process, four QoS factors are implemented to improve the user satisfaction and for better search results.

These are mentioned below,

1. Trustworthiness.
2. Relevance Matching.
3. Access cost.
4. Availability.

The trust can be defined as firm belief in the proficiency of a resource to act as expected. This firm belief is not fixed value associated with the resource. It is subject to dynamic behavior of the resource and applies only within a specific context at a given time. Trustworthiness of a resource can be calculated on the basis of feedback given by the end user. Trust rating will be given by the end user after utilizing the service. The trust rating value must lie between 1 and 10 in three different trust levels. If the trust value is between 1 and 3, it is considered as untrustworthy. If the trust value is between 4 and 7, it is considered to be average trustworthy. If the trust value is between 7 and 10, it is considered to be very trustworthy.

By providing the trust rating, end user can affect the future service discovery. Initially, the trust value for the resource is registered as 10. With the dynamic behavioral changes in the resource, the end user can rate the trust value in that instance of time, after experiencing the data service.The average trust of the resource over a period of time can be established as,

$$Avg\_trust = \frac{\sum_{i=0}^{N} T_R}{N} \qquad (1)$$

Where,

T_R is the trust rating given by the end user and $0 < T_R < 10$.

N is the total number of the end users who gave rating.

The **Relevance matching** factor of a document or any other data resource can be calculated based on the two filters which are category filter and description filter. For calculating this value, count the total number of words entered by the end use in category field and description fields. These values are assumed as α and β. Now, calculate the number of words matched in the category field of the end user and the category field of the publisher and its value is assumed as CMW. The number of words matched in the description field of the end user and the description field of the publisher, whose value is assumed as DMW. These values are also used in calculating the Effective Service Discovery as follow,

$$RMFC = Avg\_trust \times \frac{CMW}{\alpha} \qquad (2)$$

$$RMFD = RMFC \times \frac{DMW}{\beta} \qquad (3)$$

The **Access cost** factor is measured at regular interval based on average Round Trip Time (RTT). This sampling improves the accuracy of accessibility measure. The RTT value is measured in ms (milliseconds) and is divided with 1000 to get the RTT value in terms of seconds. For calculating the RTT value, we have used Ping command. Then, the Efficient Access Cost (EAC) is calculated as follows,

$$EAC = RMFD \times \frac{RTT}{1000} \qquad (4)$$

The **Availability** factor is estimated based on the probability of resource available in the past and at present. For this, total number of attempts (A) made to the service and the total numbers of failures (F) are considered. Then, the Available Service Matching (AVM) is calculated as follows,

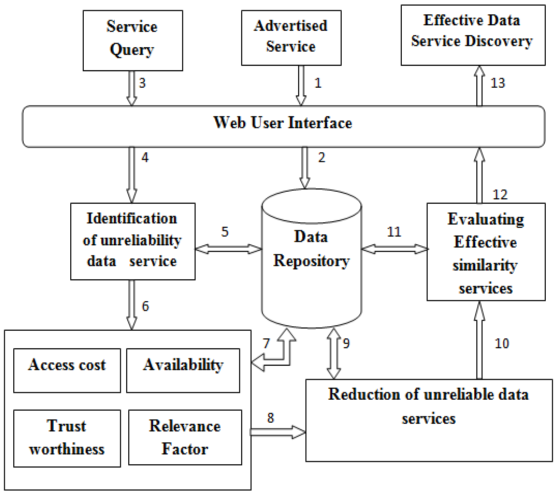
$$ASM = EAC \times (1 - \frac{F}{A}) \qquad (5)$$

Thus, by implementing these factors, the Quality of Service (QoS) related to the service discovery is achieved.

Estimation of Trust ranking for cloud services

End user provides rating to the service after utilizing the service based on his satisfaction. This Rating value should lie between 1 and 10. For providing rating, the end user has to click on the button 'Rating', which is present at web user interface. After clicking this button, the user is allowed to enter the rating value (current_Trust) and also the complete URL of the service. In this, existing_Trust_Value is the average trust value stored in the services.xml file. Visits value is the total count of end users who have given rating till that time. After entering the rating, the rating value gets updated in the service.xml file as follows (visits get incremented after this calculation),

$$Average\_Trust = \frac{current\_Trust + (existing\_Trust\_Value \times Visits)}{Visits + 1} \qquad (6)$$


Service Query
Advertised Service
Effective Data Service Discovery
3
1
13
Web User Interface
4
2
12
Identification of unreliability data service
5
Data Repository
11
Evaluating Effective similarity services
6
7
9
10
Access cost
Availability
Trust worthiness
Relevance Factor
8
Reduction of unreliable data services

First the publisher publishes the services through web user interface in step 1 and they can be stored in the repository in step 2. The end user can login into the system through web user interface in step 3. Then, the user enters a query in Web User Interface. That entered query will be passed to the identification of unreliability data service module in step 4. The results will be communicating with the data repository using step 5. The corresponding outputs will be passed to the factors module in step 6. Then, the outputs will be generated by communicating with the Data Repository. These results will be passed to the data repository in step 7 and the results will also be passed to the reduction of unreliable data services module in step 8. The results of this module will be passed to the data repository in step 9 and those results will be then passed to the evaluating effective similarity services module in step 10. The results of evaluating effective similarity services module will be passed to the data repositories in step 11 and the corresponding results are passed to the web user interface in step 12. Then the web user interface will show the results in the browser in step13 to the end user.

IV. SIMULATION, EXPERIMENTAL RESULTS AND DISCUSSION

Simulation experiments were conducted to examine the effect of our proposed approach to achieve a trustworthy cloud service discovery using the following, Trustworthiness, Relevance Matching, Access cost and Availability. In this paper, we created cloud computing environment using SQL SERVER as the database. Whenever a new publisher or end user is registered, the registration values are stored. When a new service is published by the publisher, then those service profile values are stored. These service profile details include service category, service description, URL of the service, rating, no_attempts and no_failures. These service profile details like service category, service description and URL of the service are allowed to be updated by the publisher. Rating will be updated when the new rating is entered by the end user. The value of the no_attempts will be incremented when the particular service is searched. The value of the no_failures will be incremented when the particular service is not available to the end user at that particular time. The experiment examined the cloud service discovery with and without transactions carried out without involving Four attributes. The reliability of service discovery in our approach has outperformed compared with some of the cloud computing model available for cloud based service discovery.

CONCLUSION

In this paper, we proposed a service discovery mechanism for huge data sets in cloud and big data environment. The end user is allowed to make the service discovery by entering the category and description of the service in which he is interested. If the category and description fields did not match with the fields in the service profile, then the more appropriate similar service details will be given as the result of the service discovery. In the service discovery, we have implemented 4 factors namely Trustworthiness, Relevance matching, Access cost and Availability. These factors have increased the effectiveness of the service discovery. For extension of service discovery, we have also implemented Trust ranking option to the end user. End user can share his opinion on the service by providing rating which may have an impact on future results. Here, we conclude that our model for cloud service discovery mechanism is more effective to the end users.

FUTURE ENHANCEMENTS

In this paper, we have provided the search results as a single service, which has the highest Effective Service Discovery. As a future enhancement, the search results need to be displayed as a list of services in the decreasing order of their Effective Service Discoveries. In our service discovery, the search results are based on words entered by the end user (like service category and service descriptions). In future, the search results are not only based on the exact words but also the meaning of the words entered at the fields by the end user.

REFERENCES

[1] Zhang, Qi, Lu Cheng, and Raouf Boutaba. "Cloud computing: state-of-the-art and research challenges." Journal of internet services and applications 1.1 (2010): 7-18.

[2] Buyya, Rajkumar, et al. "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility." Future Generation computer systems 25.6 (2009): 599-616.

[3] Regalado, Antonio. "Who coined 'cloud computing'? Technology Review 31 (2011).

[4] Parhi, M., Pattanayak, B. K., & Patra, M. R. , A multiagent- based framework for cloud service description anddiscovery using ontology. In Intelligent Computing, Communication and Device, Springer India, 2015,(pp. 337–348).

[5] Akinwunmi, A. O., Olajubu, E. A., & Aderounmu, G. A, A trustworthy model for reliable cloud service discovery.International Journal of Computer Applications, 87, 2014, 23–30.

[6] Meshkova, E., Riihijärvi, J., Petrova, M. and Mähönen, P. 2008. " A survey on resource discovery mechanisms, peer-to-peer and service discovery frameworks" Computer Networks: The International Journal of Computer and Telecommunications Networking ,Vol. 52 No.11, pp. 2097-2128, August, 2008.

[7] Rajendran, V., & Swamynathan, S. , A novel approach forsemantic service discovery in cloud using broker agents.In International Conference on Advances in Computing,Communication and Information Science, 2015(pp. 242–250.

[8] Han, T., & Sim, K. M . An ontology-enhanced cloud service discovery system. In Proceedings of International Multiconference of Engineers and Computer Scientists (IMEC,2010,) (Vol 1, pp. 644–649)

[9] Kang, J. and Sim, K.M., Cloudle: A Multi-criteria Cloud Service Search Engine 2010IEEE Asia-Pacific Services Computing Conference pp., 2010, 339-346

[10] Boroujerdi, M.M. and Nazem, S. , "Cloud Computing: Changing Cogitation about Computing" World Academy of Science, Engineering and Technology Vol.58,,2009, pp1112-1116,

[11] Cortázar, G.O., Zapater, J.J.S. and Sánchez, F.G. ,"Adding Semantics to Cloud Computing to Enhance Service Discovery and Access", Conference Proceedings EATIS'12, May 23-25, 2012, Spain pp. 231- 236, 2012.

[12] Naseera, Shaik, T. Vivekanandan, and Dr KV Madhu Murthy. "Trust Based Data Replication Strategy in a Data Grid." Proc. of 2nd International conference on Information Processing ICIP. 2008.

[13] Perkins, Corey J. Cloud Computing Implementation Organizational Success in the Department of Defense. No. AFIT-ENV-14-M-48, 2014.

[14] Koulopoulos, Tom, and Jim Champy. Cloud Surfing: A New Way to Think About Risk, Innovation, Scale & Success. Routledge, 2016.

[15] Naseera, Shaik, T. Vivekanandan, and KV Madhu Murthy. "Data replication using experience based trust in a data grid environment." International Conference on Distributed Computing and Internet Technology. Springer, Berlin, Heidelberg, 2008.

[16] Kang, J., & Sim, K. M. , Towards agents and ontology for cloud service discovery, 2010 In International Conference onCyber-Enabled Distributed Computing and Knowledge Discovery , 2010, (pp. 483–490).

[17] Hasan, Samer, and V. Valli Kumari. "Generic-distributed framework for cloud services marketplace based on unified ontology." Journal of Advanced Research 8.6 (2017): 569-576.

[18] Zarrin, Javad, Rui L. Aguiar, and João Paulo Barraca. "ElCore: Dynamic elastic resource management

and discovery for future large-scale manycore enabled distributed systems." Microprocessors and Microsystems 46 (2016): 221-239.

[19] Upadhyay, Nitin. "Managing Cloud Service Evaluation and Selection." Procedia Computer Science 122 (2017): 1061-1068.

[20] Di Martino, Beniamino, and Antonio Esposito. "Semantic Techniques for Multi-cloud Applications Portability and Interoperability." Procedia Computer Science 97 (2016): 104-113.

[21] Akinwunmi, A. O., E. A. Olajubu, and G. A. Aderounmu. "A multi-agent system approach for trustworthy cloud service discovery." Cogent Engineering 3.1 (2016): 1256084.

[22] Sambasivam, G., V. Ravisankar, T. Vengattaraman, R. Baskaran, and P. Dhavachelvan. "A normalized approach for service discovery." Procedia Computer Science 46 (2015): 876-883.

[23] Elgazzar, Khalid, Hossam S. Hassanein, and Patrick Martin. "Daas: Cloud-based mobile web service discovery." Pervasive and Mobile Computing 13 (2014): 67-84.

[24] Sun, Le, Hai Dong, Farookh Khadeer Hussain, Omar Khadeer Hussain, and Elizabeth Chang. "Cloud service selection: State-of-the-art and future research directions." Journal of Network and Computer Applications 45 (2014): 134-150.

[25] Parhi, Manoranjan, Binod Kumar Pattanayak, and Manas Ranjan Patra. "A multi-agent-based framework for cloud service description and discovery using ontology." Intelligent Computing, Communication and Devices. Springer, New Delhi, 2015. 337-348.

Alfazi, Abdullah, et al. "Ontology-based automatic cloud service categorization for enhancing cloud service discovery." Enterprise Distributed Object Computing Conference (EDOC), 2015 IEEE 19th International. IEEE, 2015.