ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)
Vol. 2 Issue 1 Ver 2 Jan-March 2014
International Journal of Advanced Research in Computer Science & Technology (IJARCST 2014)

Efficient Mapping of Resources Over Networked Cloud through ILS

[I]Sindhu.S, [II]Ramachandran.A
[I,II]Srinivasan Engineering College, Perambalur
Email : [I]sindhu300691@gmail.com, [II]msg2chandran@gmail.com

Abstract

Cloud computing is a delivery of services rather than a product over a network whose development intention refers to access of information with help of internet. Mapping techniques are applied to retrieve the resources from Networked Cloud Computing Environment (NCE) in an efficient manner. The purpose of NCE is to Partition the user details, and resources mapping from Virtual Nodes and Virtual Links to the Substrate network. While doing so, security problems are faced in retrieving the required data from cloud which is solved by the efficient key generation in ILS algorithm. The user can't get the needed resources in the requested time in case large number of VN requests is being sent to NCE. This project focuses on intra cloud server where mapping of requested virtual to physical resources is performed, which can be realized through the use of an Iterated Local Search Technique. Further, user request are partitioned in two phase. (i.e., Node Mapping and Link Mapping).Additionally a Mapping Server is created by applying the mapping techniques to generate keys for authentication of each user so that each node can be mapped with cloud server in the Node Mapping phase which are used to access the nodes in the Link mapping phase to map the user requested data. Finally, a thorough evaluation of proposed framework is done on a networked cloud computing environment to evaluate its performance and to yield an optimized result.

Keywords

Networked cloud mapping (NCM), resource mapping with security, node mapping, link mapping, cloud computing

I. Introduction

In cloud computing, the cloud service providers (CSPs), such as Amazon, are able to deliver various services to cloud users with the help of powerful datacenters [1]. Cloud can provide a pool of resources to many user and organization without investment of infrastructure. Cloud computing also includes online applications, such as those offered through the configurable resources for Microsoft Online Services.

Such us Hardwar services, and Internet-based clusters are also examples of cloud computing. Among the cloud service models defined by the US National Institute of Standards and Technology [2] is the cloud Infrastructure as a Service (IaaS).Infrastructure as a Service is a provision model in which an organization out sources of the virtual resources used to support operations, and including storage, hardware, servers and networking intra components. So the clients buy an application for pay per usages. Looking into the domain of a network virtualization environment the problem of mapping substrate (i.e., physical) resources (computing and communication) to Virtual Network (VN) requests is known as Virtual Network Embedding (VNE) problem. In intra cloud server, mapping of requested virtual to physical resources is performed, which can be realized through the use of an Iterated Local Search

Technique. User requests are partitioned in two phase followed by security provisioning. In this paper determined and, we explore inter domain resource mapping in a networked cloud environment via the introduction of a hierarchical framework.

In order to provide cloud IaaS with minimal management effort it is essential to address efficiently the resource mapping problem. Looking into the domain of a network virtual environment the problem of mapping substrate (i.e., physical) resources (computing and communication) to Virtual Network (VN) requests is known as Virtual Network Embedding problem. The resource mapping solution includes mapping the virtual nodes to the physical host nodes and routing the virtual links over the physical links. However, solving the Virtual network embedding problem across multiple administrative domains has been only recently addressed within the related research community. In brief, it introduces additional challenges related to 1) assigning the requested resources among the various physical resources belonging to different cloud service providers and 2) establishing the interconnection of these resources via appropriate virtual network services. The request partitioning problem is related to the graph. Graph partitioning or bisection is a common NP-complete problem with a large number of applications such as load balancing, mapping, etc. The proposed heuristic integrates a minimum k-cut algorithm followed by sub graph isomorphism. Virtual resource provisioning costs are defined according to resource availability within the cloud. The cost of an inter cloud virtual link is defined as a function of the transit network providers it traverses. In the context of network environment, split a request partitioning across multiple Infrastructure Providers (InPs) using both max-flow min-cut algorithms and linear programming techniques. The cost of peering links is also randomly set satisfying the constraint that it exceeds the cost of corresponding candidate virtual domain links of the two endpoint peers In this paper, we explore inter domain resource mapping in a networked cloud environment via the introduction of a hierarchical framework.

II. Related Work

Client receives the services from networked cloud with high traffic through Resource mapping which is not flexible and also difficult to maintain VNE problem. Security problems are faced while retrieving data's from cloud. Resources are not obtained in the requested time (network delay). Resource mapping problem for virtual network to assign the node details, and enables users to access stored data and applications on an on-demand basis.

Graph partitioning or bisection is a common NP-complete problem

ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

with a large number of applications such as load balancing, mapping, etc. Meta heuristic approaches for graph partitioning include the use of simulated annealing search [10]. In the case of the k-cut problem the goal is to find a minimum weight set of edges whose removal separates the graph into k disconnected components [11].

The proposed heuristic integrates a minimum k-cut algorithm followed by sub graph isomorphism. Intra cloud resource provisioning costs are defined according to resource availability within the cloud. The cost of an inter cloud virtual link is defined as a function of the transit network providers it traverses. VN request partitioning across multiple Infrastructure Providers (InPs) using both max-flow min-cut algorithms and linear programming techniques. Intra provider provisioning costs are randomly set. The approach described in [5] applies a request partitioning Meta heuristic, an initial request partitioning and reallocates the different parts of the VN via local improvements.

Intra domain resource provisioning costs are randomly generated. The cost of inter domain links is assumed to be an order of magnitude higher than the average intra domain cost. It should be noted that in most of these approaches either fixed or randomly generated provisioning costs are taken into partitioning. In our proposed framework more realistic intra- and inter domain resource.

Virtual network embedding within a cloud network, with conditions on virtual nodes and virtual links for the virtual environment, it can be reduced to the NP-hard multi way separator problem [17]. Several of them decompose the problem into the node mapping phase and the link mapping phase, to reduce the overall complexity (e.g., [18]). Recent approaches tend to solve the two problems either simultaneously (e.g., [12]) or providing some type of coordination between the two phases (e.g., [7]).

The starting point for the search may be a randomly create a candidate solution by a greedy construction heuristic method. Normally a greedy solution results to better performance on mapping the resources over minimum iteration steps. Local Search refers to the local improvements of the solution produced from the perturbation based on user attributes. Usually, local search consists of moving from one solution to another solution according to some well-defined rules related to a optimal solution, accepted moves within the cloud network and termination criterion [25, 26], [10]. Concerning acceptable movements in the search space, best improvement or first improvement can be used. Finally, common stopping condition for local search algorithms are definition of a maximum number of iteration, while terminating the search when no improvement has been obtained for a number of iterations.

Perturbation generates new starting points of the local search by modifying some local optimum solution. Perturbation guides the algorithm to accomplish by randomly remapping nodes from local search. Hence to investigate other parts of the search space. The strength of the algorithm is plays a significant task on cloud resource in the efficiency manner, where strength is generally defined as the change the number of solution components. For a very strong perturbation, Iterated Local Search (ILS) behaves like random key generation. For providing privacy to restart while for a weak perturbation of the algorithm's behaviour is greedy technique.

In most cases, VN requests are handled upon arrival (e.g.,[7], [19]). Dynamic (e.g., [18, 20]) approaches support remapping of resources during the life-time of the request as opposed to static ones (e.g., [20, 7]). VNE algorithms suffer from scalability issues and hence request partitioning has been studied for mapping each part of a request to a different part of the substrate network (e.g., [20]). Distributed algorithms have been proven to enhance overall network resiliency as well as overcome scalability limitations and delays imposed by maintaining information centrally (e.g., [21]). Moreover, the proposed intra domain resource mapping formulation and approach takes into account quality of service (QoS).

III. System Architecture

System design is the process of defining the architecture, components, modules, and data for a system to satisfy specified requirements. Design could see it as the application of systems theory to product development. There is specification with the disciplines of systems analysis, and systems architecture and systems engineering. DFDs can also be used for the visualization of data processing. In this diagram it shows that mapping in a networked cloud environment. The key provisions of this work are as follows the providing privacy. Specifically, to deal with the inherent complexity and Iterated Local Search. Many applications retrieved from cloud server for pay per usage. After the registration client details are mapped with mapping server. Then mapping server response to the client and what user needs that data can be mapped with mapping server. Mapping server mainly used to deliver the resources in security manner.

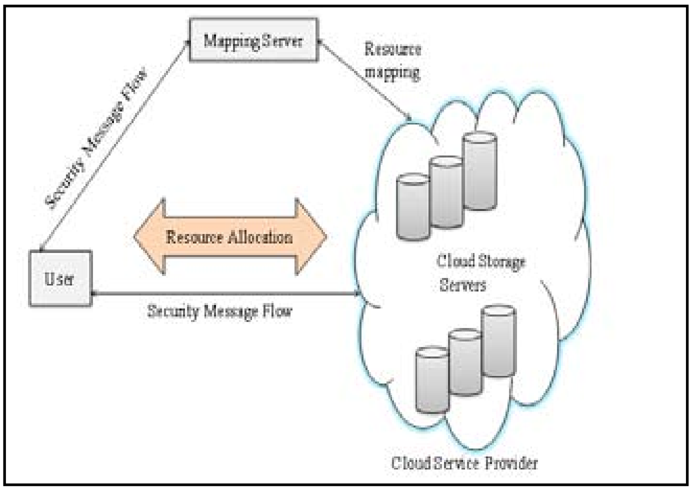

Fig.1 : Overall System Architecture

Each user having the two keys. Those keys are generated by only mapping server. In intra cloud server, mapping of requested virtual to physical resources is performed, which can be realized through the use of a Iterated Local Search Technique. User requests are partitioned in two phase followed by security provisioning. Client receives the services from networked cloud with high traffic through Resource mapping which is not flexible. Difficult to maintain the Virtual Network Embedding (VNE) problem.

A. System Process

Trusted Boot

A kernel module, which call trust agent, is loaded on boot can be compiled in the Kernel node. Module loading after boot to prevent any poundering with the agent, a user-space trust client can concurrently opens the device and socket to the trusted server, then waiting for communication with the server. The trust agent is to identify the signature of each client program and also verifies the integrity of the client. This verification process also prevents any other client program from opening the device in our security model.

Initial Authentication

When the remote server gets a connection from a client, it requests the initial confirmation or attestation. The trust client uses the system call whether to specify the read mode or write mode, then request the required information from the trusted agent. The trust agent forwards. The trust client forwards the information to the server which verifies the information. The trust agent and the remote server set up a shared key through an RSA key exchange protocol based on the keys. At the trust agent, each keystroke event is encrypted and generate to their corresponding signature. Both the encrypted event and its signature are covered in a packet. The trust client forwards the packets to the remote server which verifies the completeness of events by checking the signatures using a keyed hash function. If the signature associated with mapping server events do not pass the server's verification, then this process notified by the trust agent.

Node Mapping Phase

In this node mapping phase, this is mapping the user details with server. Initially client request to cloud server for purpose of access the number of resources and storing the shared resources, then the mapping server can generate the key after valid client registration. Client access the cloud application through security key. Mapping server is create two keys for each user, where first key is used for node mapping phase and second key is used for link mapping phase. Attribute mapping that is IP address, user name and host name these are mapped with mapping server using first security key.

Link Mapping Phase

Once the node mapping procedure has been successfully completed, then link mapping procedure is initiated in mapping server by user through second key. In this phase the non-functional attributes are mapped with mapping server, where mapping server contains all the functional and non-functional attribute details. Link mapping achieved by solving the flow allocation problem which is allowing traffic bifurcation (multi commodity flow problem). On other hand, a shortest path algorithm can be applied in order to reduce cost for each flow to a single path.

Performance Evaluation

Mapping server to verify integrity, those users can access the resources or application from the cloud server. In order to prevent the privacy leakage. Then the mapping server to mapping the functional and non-functional attributes. After the node mapping and link mapping procedure, mapping sever can deliver the resources according to their requested client based on security.
This module to evaluate the malware actively makes the outside connections for command and control the activity or attacks. For example, malware may attempt to log user request inputs, while introduce the traffic bypassing the host's firewall, forge input events, and tamper with network traffic. In the particular study an iterated descent algorithm for Local Search was applied employing a first improvement pivoting rule.

IV. Proposed Framework

In intra cloud server, mapping of requested virtual to physical resources is performed, which can be realized through the use of an Iterated Local Search Technique. User requests are partitioned in two phase. 1. Node Mapping Phase, 2.Link Mapping Phase.

Node Mapping Phase - The each node details (user name, IP address, host name) are mapped with mapping server. After the registration the mapping server to generate two keys for each user.
Link Mapping Phase - Once the node mapping procedure has been successfully completed, to map the data what the user needs. Resources are obtained in the requested time (network delay).

Request Partitioning Through ILS

1. Generate Initial Solution

To select an appropriate initial solution for the particular ILS implementation,
Three cases where examined:
(a) Minimum cost node mapping
(b) Single path mapping with security
(c) Random key generation for ILS
A random initial solution was selected; hence no significant improvement was noticed, both in terms of provisioning cost and computing time.

2. Perturbation

In terms of perturbation, different strengths have been proven, where perturbation is practiced by randomly remapping nodes within the set of interconnected clouds or intra cloud. The best performance was achieved when as many as 80 percent of the solution components were altered by the perturbation.

3. Acceptance Criterion

In order to select an appropriate memory less acceptance criterion two different choices were examined: (a) only better solutions are accepted and (b) a simulated annealing type acceptance criterion. The first approach proved to be the most suitable. This choice of the acceptance condition has the advantage that it implements a randomized descent in the space of locally optimal solutions.

4. Local Search

In the particular study an iterated descent algorithm for Local Search was applied employing a first improvement pivoting rule. A simple type of move is used to define a neighbourhood, at each iteration a node is remapped to the various remaining CPs. A maximum number of iterations are used as a stopping criterion.
Request partitioning approach based on ILS (using mapping server).In NCE Partitioning techniques are performed in two phases. (i.e., Partitioning and embedding). To increase the average utilization rate of resources. Security problems are faced while retrieving data's from cloud and to overcome this problem to generate the keys. That key is mainly used to verify the integrity of the client. The mapping server to map the node details and the user request. If the user is authorized means to deliver the resources otherwise to block the client.

Algorithm 1. Iterated Local Search

Si = GenerateInitialSolution ()
S = Local Search (Si)
while Stopping criterion is not met do
S' = Perturbation(s, history)

ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

Sc = Local Search (s')
S = Acceptance Criterion(s, Sc, history)
end while

ILS Meta heuristic is adopted for request partitioning in this paper, mainly due to its intrinsic simplicity and general applicability, overcoming at the same time performance/ scalability issues that arise when addressing the resource mapping problem in real time and for large incoming requests.

Regarding the embedding phase, a NCM approach based on the principles set by Chowdhury et al and Papagiani et al, was adopted. The node mapping phase and the link mapping phase are defined as follows:

1. Node mapping phase:

The physical network graph is augmented by introducing one pseudo node for each virtual node and connecting these pseudo nodes to the physical ones. In the acuminated substrate, each virtual link with bandwidth constraints is treated as a commodity having as source and destination, a pair of pseudo nodes. Hence, the problem is formulated and provided as a security MIP problem. While solving the flow allocation problem, taking into consideration virtual nodes and virtual link those results in substrate node mapping process can be created. Specifically, the optimal fractional solution is computed for the random key generation and problems of Linear Programming relaxation. While relaxed problem can be solved by any suitable linear programming method and provide security, for providing the iteration using polynomial condition. Hence a rounding technique is applied [29] to obtain the integer solution of the MIP problem. Then, the substrate node is selected then the mapping process applied, while taking process with into consideration virtual link demands.

2. Link mapping phase

Once the node mapping procedure has been successfully completed, then link mapping procedure is initiated in mapping server by user through second key. In this phase the non-functional attributes are mapped with mapping server, where mapping server contains all the functional and non-functional attribute details. Link mapping achieved by solving the flow allocation problem which is allowing traffic bifurcation (multi commodity flow problem). On other hand, a shortest path algorithm can be applied in order to reduce cost for each flow to a single path.

V. Experimental Evaluation

In this section, the efficiency of the proposed approach is evaluated via simulation. A discrete event java-based simulator (Simulator for Controlling Virtual Infrastructures) was used for that purpose [8]. To allow for a thorough evaluation in the following, compare four different approaches, that are varying on the request partitioning and intra domain VNE methodology maintained.

Table 1: Evaluated Approaches

APPROACHES	REQUEST FOR PARTITIONING	INTRA-CLOUD VNE
1	ILS	GSP
2	Exact	GSP
3	ILS	NCM
4	Exact	NCM

The sets of approaches are presented in Table 1. The proposed Iterated Local Search ILS heuristic is compared against an accurate solution where the request partitioning problem is formulated as a quadratic programming problem and through appropriate transformation a linear integer program is extracted [12]. The linear integer program is solved using the branch and bound algorithm. To successfully added the virtual resources within a cloud two different approaches have been followed: 1) a Greedy node mapping followed by a Shortest Path algorithm for the link mapping phase [7], and 2) a NCM approach based on the methodology

The objective is to evaluate the request splitting algorithm performance in terms of VN provisioning cost and computing time required to split a VN graph request. For that purpose Approach 1 (ILS-GSP) and Approach 2 (Exact-GSP) have been proved. To estimate the Provisioning cost is according to the objective function 1. There are two approaches are compared in terms of the percentage of the incremental provisioning cost.

The impact of the request partitioning algorithm employed is important on the embedding cost. In combination with the embedding revenue graph, it is mining from the application of the exact partitioning algorithm leads to increased embedding cost due to longer paths employed in the link mapping process, when G-SP is occupied, then the same difference is not reflected in the case that NCM is applied as the intra domain embedding algorithm due to the fact that the NCM selects an appropriate node mapping that target toward physical path minimization.

VI. Conclusion

This paper focuses basically on the cloud resource mapping problem over a networked cloud computing environment. Additionally, high importance is provided towards high-performance algorithms in terms of embedding effectiveness and runtime complexity. In the proposed framework, an ILS-based heuristic is employed to provide a cost-efficient resource allocation that realizes the partitioning of the user request. Empirical and Numerical results obtained through modelling and simulation demonstrates the efficiency of the proposed approach. The proposal is compared against an exact method to provide minimal cost partitioning in networks. Detailed evaluations have shown great cost efficiency over a large number of VN request instances and networked cloud sizes to reduce computational time. Subsequently, VN partitioning is proceeded with a sophisticated intra VNE algorithm based on linear programming. This will be utilized in cloud so that each sub-VN is properly allocated .The target is to minimize the cost of embedding a request by performing load balancing and enhancing the scalability in cloud. Finally, resource provisioning cost based on node scarcity and average utilization is verified. A challenging and interesting approach would be to derive utilization based on more enhanced queuing models that could better reflect potential nonlinear behaviours.

References

[1] *Aris Leivadeas, C. Papagianni, and S. Papavassiliou, "Efficient Resource Mapping Framework over Networked Clouds via Iterated Local Search-Based Request Partitioning", IEEE Transactions On Parallel And Distributed Systems, Vol. 24, No. 6, June 2013.*

[2] *R. Buyya, C.S. Yeo, and S. Venugopal, "Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering It Services as Computing Utilities," Proc. IEEE Int'l Conf. High Performance Computing and Comm. (HPCC '08), pp. 5-13, Sept. 2008.*

[3] *X. Cheng, S. Su, Z. Zhang, H. Wang, F. Yang, Y. Luo, and J. Wang, "Virtual Network Embedding through Topology-Aware Node Ranking," SIGCOMM Computing Comm. Rev., vol. 41, no. 2, pp. 38-47, Apr. 2011.*

[4] *I. Fajjari, N. Aitsaadi, G. Pujolle, and H. Zimmermann, "Ether: Malware Analysis via Hardware Virtualization Extensions," Proc. IEEE Int'l Conf. Comm. (ICC), pp. 1-6, June2011.*

[5] *I. Houidi, W. Louati, W.B. Ameur, and D. Zeghlache, "Virtual Network Provisioning Across Multiple Substrate Network," ELSEVIER Computer Networks, vol. 55, no. 2, pp. 1011-1023, 2011.*

[6] *H. Khazaei, J. Misic, and V.B. Misic, "Blade: An Attack-Agnostic Approach For Preventing Drive-By Malware Infections," IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 5, pp. 936-943, May 2011.*

[7] *J. Lischka and K. Holger, "Peer-To-Peer Botnets: Overview and Case Study," Proc. ACM Workshop Virtualized Infrastructure Systems and Architectures (SIGCOMM '09), pp. 81-88, Aug. 2009.*

[8] *C. Papagianni, A. Leivadeas, S. Papavassiliou, V. Maglaris, C. Cervello-Pastor, and A. Monje, "On the Optimal Allocation of Virtual Resources in Cloud Computing Networks," Submitted 2011 (currently under revision), 2011.*

[9] *L.A. Sanchis, "An Efficient Black-Box Technique For Defeating Web Application Attacks," IEEE Trans.Computers, vol. 38, no. 1, pp. 62-81, Jan 2011.*

[10] *F. Zaheer, J. Xiao, and R. Boutaba, "Multi-Provider Service Negotiation and Contracting in Network Virtualization," Proc. IEEE Network Operations and Management Sump (NOMS), pp. 471- 478, June 2010.*