IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 4, April 2016

An Authentication Protocol for Managing Dynamic Resources in a Massive Cloud Platform

Suganya .D[1], Sowmya .S[1], Ragini .S[1], Harikrishna Pillutla[2]

Student, CSE Department, Rajiv Gandhi College of Engineering and Technology, Puducherry, India[1]

Assistant Professor, CSE Department, Rajiv Gandhi College of Engineering and Technology, Puducherry, India[2]

Abstract: Cloud Computing is designed as an event to anticipate the applications supported by the new paradigm and validate the mechanisms and the techniques .We address the complication of dynamic resource management for a massive cloud environment. We incorporate outlining distributed middleware architecture and performing one of its key elements: a gossip protocol that (a) establishes a generous resource allocation among sites/application, (b) dynamically adapts the allocation to load changes, and (c) rates both in number of physical machine sites/application. We exemplify the resource allocation problem as that of dynamically augmenting the cloud service under CPU and memory constraint. We present a protocol that figure out a solution without considering memory constraint and prove correctness and convergence properties. Hence we extent that protocol to provide an analytical solution for the entire problem that contains reducing the cost for accommodating an allocation. This protocol continuously executes on dynamic, local input and does not need global synchronization, as all other gossip protocol do. This paper also includes a Remote Authentication Dial-in User Service Protocol. RADIUS is a protocol that facilitates Network Access Server to utilize shared authentication server for authorization and authentication. We appraise the heuristic protocol through simulation and find its conduct to be well coordinated with our design goals.

Keywords: Cloud Computing, RADIUS Protocol, Dial-in User Service Protocol, Remote Authentication.

I. INTRODUCTION

We consider the obstacle of resource management for an extensive cloud environment. Such an environment comprises the substantial framework and correlated functionality that empowers the provisioning and authority of cloud services. As long as our input is admissible in more general context, we conduct the discussion from the aspect of the Platform-as-a-Service (PaaS) concept, with the explicit use case of a cloud service provider that hosts sites in a cloud environment. The collaborators for this use case are illustrated in figure. The cloud server provider owns and governs the physical infrastructure, on which cloud services are granted. It offers hosting services to site owners through a middleware which executes on an infrastructure owner furnishes services to their corresponding users via sites that are hosted by the cloud service provider. Our contribution can also be applied to Infrastructure-as-a-service (IaaS) concept. A use case for this concept could include a cloud holder running a set of virtual appliances which are hosted on the cloud infrastructure, with services provided to end users via the public internet. For both prospect this paper proposes resource allocation protocol that dynamically places site modules on servers within a cloud. The authentication and authorization is done using RADIUS protocol. This protocol is a client/server protocol. The client is the Network Access Server (NAS) and the server is a daemon process running on a machine. The server acts as a proxy client to other RADIUS servers. RADIUS allows various clients to use a single authentication and authorization server for user authentication. Password of the user was encrypted that is being transmitted to the server and the client can automatically authenticate the server from reply. Here the replies are also secured from alteration. The demand for centralized user authentication and authorization is very high in case of networks having users and resources shared via networks. The network security has been increased and the demand for administrative support has been reduced by centralization. The security issues are more important in case of users who use the network and the resources from a different network. To solve such problems the RADIUS protocol was developed.

II. SYSTEM ARCHITECTURE

Data centers functioning a cloud environment often encompasses a large number of machines that are linked by a high speed network. Users access spot hosted by the cloud environment via the public internet. A site generally accessed through a global directory services, such as DNS. A request to the site is routed via the internet to a machine inside datacenter either processes request or forwards it.

The cloud middleware architecture is shown in figure 1. The component of the middleware layer runs on all machine. The resources of the cloud are mainly consumed by module instances by which the functionality of the site is made up of one or more modules. In middleware, a module either encompasses part of service logic of a site (denoted by m_i in figure 1) or a site manager (denoted by SM_i). Each machine runs a machine manager component that figure out the resource allocation policy, that includes deciding the modules instances to run. The resource allocation policy is evaluated by a protocol which runs in the resource manager component. This component takes the estimated demand for each module as input that the

DOI 10.17148/IJARCCE.2016.5448

machine runs. The computed allocation policy is directed to the module scheduler for implementation, as well as the site managers for making decisions on request forwarding.

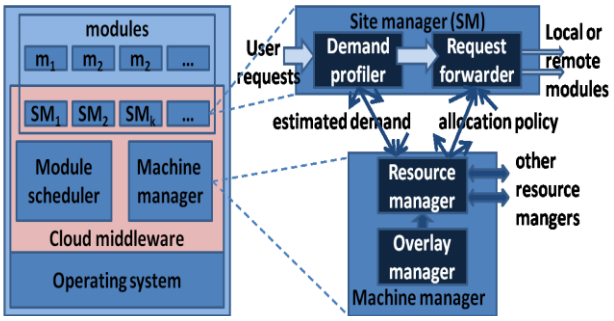

Fig 1 Architecture for cloud middleware (left) and components for request handling and resource allocation (right)

Our architecture accomplices one site manager with each site. A site manager handles user requests to a particular site. The two components present in it are: a demand profiler and request forwarder. The demand profiler estimates the resource demand of each module of the site based on request stats, QoS targets, etc. The demand estimate is dispatched to all machine managers that run instances of module attached to the site. Similarly, the request forwarder dispatches user request for processing to instances of module that belong to the site. Request forwarding decisions takes resource allocation policy and constraint into account such as session affinity. Figure 1 (right) show the components of a site manager and how they related to machine managers. The above architecture is not pertinent for the case where a single site manager could not handle the approaching request stream for a site. However, a plan for a site manager to scale can be conceptualized. For instance, a layer 4/7 switch could be introduced splits the load among lot of instances of site managers, through which each such instance would function like a site manager associated single site. The rest of this paper focus on functionality of resource manager component. For other components, such as overlay manager demand profiler we depend on known solutions.

III. FORMALIZING THE ISSUES IN RESOURCE ALLOCATION BY THE CLOUD MIDDLEWARE

For this work, we consider a cloud has having computational resources (ie.CPU) and memory resources that are applicable on the machine in the cloud infrastructure. As interpreted earlier we inhibit the discussion to the case where all machines belong to a single cluster and conspire as peers in the function of resource allocation. The specific problem we address is that of placing modules on machine and allocating cloud resources t this module, in such a way that a cloud service is magnify under constraints as cloud utility we choose the minimum utility achieved by any sites, which we describe as the minimum utility of module instances. We codify the resource allocation issue as that of maximizing the cloud service under CPU and memory constraint. The result to this problem is that configuration matrix which regulates the module scheduler and the request forwarded components. At distinct point in time events occur such as, demand changes removal an addition of site, machine, etc. In return to such an event, the optimization problem is fixed again, in order to keep utility maximized. We add a trivial objective to optimization problem which states that the cost of change from the ongoing a new configuration must be diminished.

IV. A PROTOCOL FOR AUTHENTICATION

A. Radius Protocol

The flow of RADIUS protocol initially involves authentication. The user initiates PPP authentication to the Network Access Server. Network Access Server asks for username and password. Then the user replies. RADIUS client sends username and encrypted password to the RADIUS server. RADIUS server responds according to the user whether to accept, reject or challenge. Challenge takes place using a Challenge Handshake Authentication Protocol (CHAP). Then finally the RADIUS client acts upon services and service parameter bundled with Accept or Reject.

B. Authentication and Authorization

Authentication refers to confirmation that a requesting user is valid or not. It is based on the identity and credentials. Authorization refers to the granting of service to the users based on the authentication. The RADIUS server can support a variety of methods to authenticate a user such as PPP, PAP or CHAP, and other authentication mechanisms. The user login consists of an Access request from NAS to the RADIUS server. The corresponding response is given from the server. The request packet contains the username, password in encrypted form, NAS IP address and port. The server receives the request from the NAS and searches the database. If the username does not exist in the database then an Access reject message is sent from server. If the username and password matches the server returns an accept response. The response message includes the parameters to be used in the session.

C. Accounting

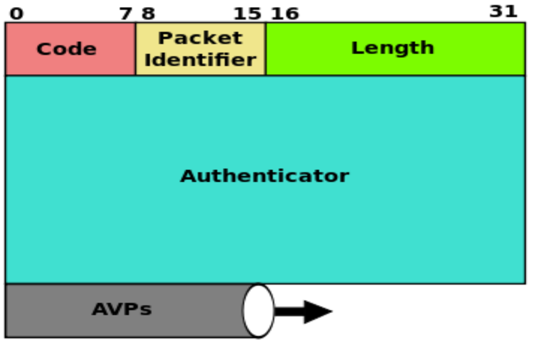

Fig2. RADIUS packet data format

The feature of accounting is independent from authentication and authorization. Accounting involves the tracking of the consumption of network resources by users. Accounting allows data to be sent at the start and end of the sessions. It indicates the amount of resource which includes time, packets, bytes etc. some information which is assembled in accounting involves user's identity, the nature of the delivered services whenever the service id initiated and ended.

[

D. Packet Structure

The RADIUS data packet format includes code, the identifier, the length, the authenticator and the attributes. Figure 2 gives the structure of RADIUS data format. The identifier field is used for request and reply. The length field denotes the length of the packet. Authenticator is used to authenticate the reply.

E. Background Information

User authentication relies on the communication between a Network Access Server (NAS) and RADIUS Server. In General, RADIUS is considered as a connectionless service. Rather than the transmission protocol RADIUS enabled devices are used to manage the issues related to server availability, timeouts and retransmission. RADIUS is a client/server protocol. The user information is passed by the client to the servers and acts on the response that is returned. Then the user connection requests are received by the servers, user gets authenticated and configuration information is returned to the user. For other kind of authentication servers, a RADIUS client can act as a proxy client to all other RADIUS servers.

- PPP authentication is initiated by the user to the NAS.
- Username and password (if Password Authentication Protocol [PAP]) or challenge (if Challenge Handshake Authentication Protocol [CHAP]) is prompted by NAS.
- Reply from the user. Username and encrypted password is sent by the RADIUS client to the RADIUS server.
- Response from RADIUS server with Accept, Reject Or challenge.

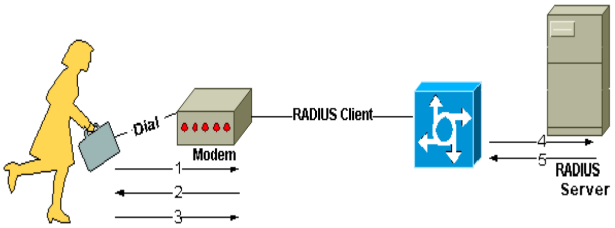

Fig3. Interaction between dial-in user and the RADIUS client and server

F. Network Access Server

Network Access Server is a service that client dial to get access to the network.NAS is situated at the internet provider's point of view to provide Internet access to its users. To a remote resource it is a single point of access. It allows remote access to a network and so called a Remote Access Server. It is an Initial entry point to a Network and acts as a gateway to guard to a protected resource.

V. RELATED WORK

The two main problems in this paper that is related to resource management are application placement and load balancing in processor networks**.** Through mapping a set of applications onto a set of machines, the application placement in datacenters is always modeled in such a way that some function is maximized under resources .The goal of the work in this paper is to magnify a cluster service beneath changing demand, even though a different concept of service is used .The service differentiation works very well under certain workload conditions with the choice of utility function .Every module receives the stake of the CPU resources of the cloud and is guaranteed to have satisfied demand Distributed load balancing have been studied broadly for homogeneous and heterogeneous systems. It is applicable for both divisible and indivisible demands.

VI. DISCUSSION AND CONCLUSION

In this paper, we make an important grant towards the resource management middleware for cloud environments. Here, a key component of such a middleware is identified and a protocol was presented that can be used to meet our resource allocation design goals. A gossip protocol P* was presented which computes in a continuous and distributed fashion and a heuristic solution to the resource allocation problem for changing the resource demand. RADIUS protocol widely used authorization and authentication protocol. It was designed originally to allow Network Access Server to authenticate dial-in users. RADIUS doesn't have error messages and retransmission is used in case of any issues

REFERENCES

[1] R. Yanggratoke, F. Wuhib, and R. Stadler, "Gossip-based resource allocation for green computing in large clouds," International Conference on Network and Service Management,2011.

[2] M. Jelasity, A. Montresor, and O. Babaoglu, "Gossip-based aggregation in large dynamic networks" ACM Trans. Computer Syst.,(vol. 23, no. 3,pp. 219–252), 2005.

[3] F. Wuhib, R. Stadler, and M. Spreitzer, "Gossip-based resource management for cloud environments," International Conference on Network and Service Management,2010.

[4] F. Wuhib, M. Dam, R. Stadler, and A. Clem, "Robust monitoring of network-wide aggregates through gossiping," IEEE Trans. Network and Service Management, on(vol. 6, no. 2, pp. 95–109), June 2009.

[5] G. Pacifici, W. Segmuller, M. Spreitzer, and A. Tantawi, "Dynamic estimation of CPU demand of web traffic," International Conference on Performance Evaluation Methodolgies and Tools, 2006.

[6] S. Voulgaris, D. Gavidia, and M. van Steen, "CYCLON: inexpensive membership management for unstructured p2p overlays," J. Network and Systems Management,.on(vol. 13, no. 2, pp. 197–217), 2005.

[7] R. L. Graham, "Bounds on multiprocessing timing anomalies," SIAM J. Applied Mathematics. on (vol. 17, no. 2, pp. pp. 416–429), 1969.

[8] C. Tang, M. Steinder, M. Spreitzer, and G. Pacifici, "A scalable application placement controller for enterprise data centers," International Conference on World Wide Web, 2007.

[9] H. Shachnai and T. Tamir, "On two class-constrained versions of the multiple knapsack problem," Algorithmica, on (vol. 29, no. 3, pp. 442–467), Dec. 2001.

[10] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: evidence and implications," in Proc. Annual Joint Conference of the IEEE Computer and Communications Societies, 1999, on (vol. 1, pp. 126–134).

[11] C. Adam and R. Stadler, "Service middleware for self-managing largescale systems," IEEE Trans. Network and Service Management,on(vol. 4, no. 3, pp. 50–64), Apr. 2008.

[12] J. Famaey, W. De Cock, T. Wauters, F. De Turck, B. Dhoedt, and P. Demeester, "A latency-aware algorithm for dynamic service placement in large-scale overlays," International Conference on Integrated Network Management, 2009.

[13] Y. T'Joens, R. Ekstein, M. De Vries, O. Paridaens. "Framework for the extension of the RADIUS(v2) protocol"june 2000.

[14] G. Zorn, B. Aboba, D. Mitton." RADIUS Accounting Modifications for Tunnel Protocol Support". Available: http://www.ietf.org/rfc/rfc2867.txt.

[15] D. Mitton." Network Access Servers Requirements: Extended RADIUS Practices", 2000. Available: http://www.ietf.org/rfc/rfc2882.txt.