Dynamic Mobile Cloudlet Clustering Policy Using Nano Cloudlet Clustering

Ms. Soundarya G, Student, EPCET, Bangalore, India, 27soundarya@gmail.com

Mrs. Bindiya A J, Assistant Professor, EPCET, Bangalore, India, bindiyaaj@gmail.com

Ms. Shubha M, Student, EPCET, Bangalore, India,shubhashubha44@gmail.com

Ms. Tejashwini N, Student, EPCET, Bangalore, India, tejashwini557@gmail.com

Ms. Likitha N, Student, EPCET, Bangalore, India, likithan17@gmail.com

Abstract: In spite servers have obtained too many capabilities such as CPU power, memory and battery they still insatiable the necessity of high resource consuming applications. Fog computing is one of the solution for unloading the task of a mobile. Yet the potential of fog server is limited due to the high deployment cost. A Dynamic Cloudlet Cluster Policy (DMCCP) to use cloudlet as additive for the fog server for unloading. The main plan is that by observing each server resource amount, the DMCCP system clusters the optimal cloudlet to come across the request of different task from the local server. In simulation result, we show that the proposed policy can cluster the exact optimal cloudlets for different resources demands of a task in a real time processing time compared to the optimal method.

Keywords: Cloudlet, Cloudlet Cluster, DMCCP, Fog computing, request, unloading.

I. INTRODUCTION

These days, Servers have become a mandatory part of human life all over the world. . However, the computing power of each server is still not enough to satisfy the high requirement of applications. Cloud computing is one of the solutions. Cloud computing plays a remote server role for sharing the massive amount of on demand services and provides flexible information technology (IT) resources for different applications demand. Lots of enterprises take advantage of cloud computing to allow high-performance services on mobile applications for the users. Despite cloud computing benefits, it faces the challenges of solving delayed response caused by the far-away distance between cloud and users.

II. LITERATURE SURVEY

Fog Computing: Mitigating Insider Data Theft Attacks in the Cloud

Cloud computing assure to seriously change the way computers are used, access and store our personal information. With the new computing and communications pattern arise new data security challenges. Existing data protects the mechanisms such as encryption is unsuccessful in preventing data theft attacks, mostly those immoral action by an insider to the cloud provider. We propose a distinct approach for securing data in the cloud using outrageous decoy technology. We observe the data access in the cloud and detect abnormal data access patterns.

Energy Management-as-a-Service over Fog Computing Platform

Energy management may admit to help us to hold out zero net energy for the eminent domain. Advancement in technology, cost, and feature size have enabled devices all where, to be connected and interactive, as it is called Internet of Things (IoT). The greater complexity and data, due because of growing number of devices such as sensors and actuators, which requires much powerful computing resources which are provided by cloud computing. Anyhow, scalability has become the possible issue in cloud computing.

Fog computing: A cloud to the ground support for smart things and machine-to-machine networks

Cloud services to smart things face latency and intermittent connectivity issues. Fog devices are located between cloud and smart devices. The high speed Internet connection to the cloud, and physical closeness to users, permit real time applications and location based services, and mobility support. Cisco advanced fog computing concept in the areas of smart grid, connected vehicles and wireless sensor and actuator networks. This examine article expands this concept to the decentralized smart building control, recognizes cloudlets as special case of fog computing, and relates it to the software defined networks (SDN) scenarios. Our literature review identifies a handful number of articles.

Fog Computing and Smart Gateway Based Communication for Cloud of Things

With increasing in applications the domains of ubiquitous and context-aware computing, Internet of Things (IoT) is

 DOI : 10.18231/2454-9150.2019.0576

gaining importance. In IoT's, segment of it, it is sensor nodes or dumb objects, so very various types of services can be produced. The amount of data IoTs are going to produce would not be possible for standalone power-constrained IoTs to handle. Cloud computing plays an important role here. Integration of IoTs with cloud computing, known as Cloud of Things (CoT) help us to achieve the goals of envisioned IoT and future Internet. This IoT-Cloud computing integration is not straight-forward. It involves many challenges. One of the main challenge is data trimming. Because of useless communication not only burdens the network, but also the data center in the cloud. For this issue, data can be preprocessed and trimmed before sending to the cloud. This can be done through a Smart Gateway, with a Smart Network or Fog Computing.

III. EXISTING SYSTEM

To solve the weak points of cloud computing, many researchers have proposed their techniques and schemes relate by using fog computing paradigm. Fog computing, known as the mobile edge computing (MEC) defined by European Telecommunications Standards Institute (ETSI), brings the remote services more closely to the users.

Drawbacks: This fog server can run close to users. The deployment of fog server at the edge of the network such as access points (AP). The architecture of fog computing consists of cloud servers, fog servers, and end-user equipment. The architecture, consists each user equipment connects with one of fog servers, and the fog server can communicate with each other through a cloud server. Fog server at the edge of the network provides low latency and location awareness services which increases the quality-of-service (QoS) for real-time demand response applications.

IV. PROPOSED SYSTEM

A dynamic server cloudlet cluster policy (DMCCP) for fog server offloading. We assume that the fog server will dynamically cluster the Cloudlet corresponding to different resource demands of local server task. All available servers, which are potential IT resource for cloudlet, should be pre-connected to a fog server via wireless network. In order to efficiently use the potential IT resource pool, we use the server resource monitor in fog server to observe available resources amount of each server. All the information of the server such as remaining central processing unit (CPU), memory, and battery level will be sent to fog server when fog server requests for the monitoring data through server resource monitor. If a single server requests a task to the fog server, the fog server will cluster a cloudlet by DMCCP based on resource demands of this specific task.

Benefits: The main objective of this system is to propose a dynamic mobile cloudlet cluster policy (DMCCP) to use cloudlets as a supplement for the fog server for offloading. The main idea is that by monitoring each server resource amount, the DMCCP system clusters the optimal cloudlet to meet the requests of different tasks from the local server.

V. ARCHITECTURE

IP Look UP Function Module

IP addresses of the client systems should be maintain in the server database, based on the client's ip addresses, request will be serve by the respective nano data center, if the load status of the nano data center is overload and the ip address is unknown then the request will be redirected to the main data center.

Energy Management Setting

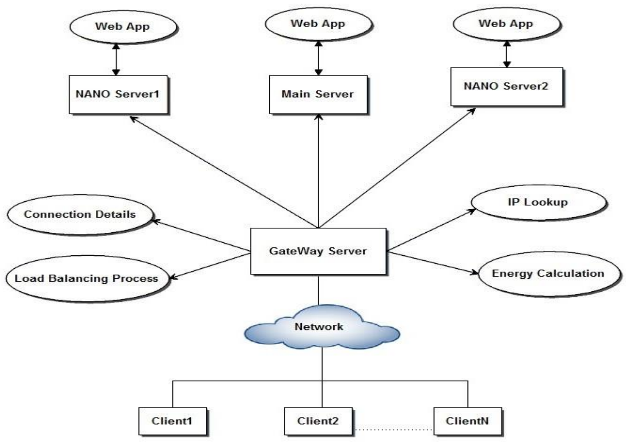
There is a energy variation between Main Server and Nano Servers, for the measurement purpose system has to maintain the following units in this system

- *P*(*C*) Power consumption under load *C*
- *P*idle Idle power consumption of a network equipment
- *P*max Maximum power consumption of a network equipment
- *C*max Maximum capacity/load of a network equipment
- *E* Incremental energy per connection
- *U* Load threshold

Energy Monitoring

Based on the number of connections the energy consumption differs from one server to another server. To monitor and keep track of energy conception this module is used.

SYSTEM ARCHITECTURE



Load Balance Module

When a connection request received in Gateway server, it has to fetch the IP-Address of the client and identify the Server for the IP Address using look-up function, then Energy Threshold test has to be conducted. If the test pass then connection send to corresponding server, else based on

 DOI : 10.18231/2454-9150.2019.0576

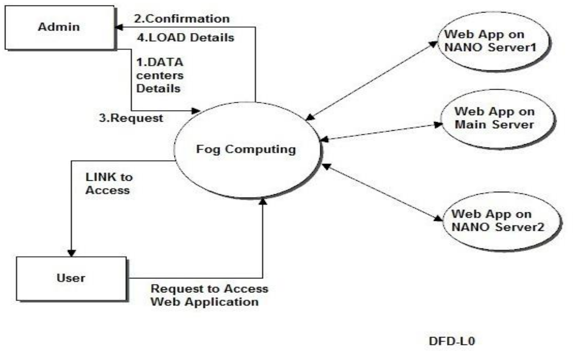
ISSN : 2454-9150 Special Issue - NGEPT - 2019

the loads of all other server, one server is shortlisted and connection send to that shortlisted server.

VI. FEATURE ENHANCEMENT AND APPLICATIONS

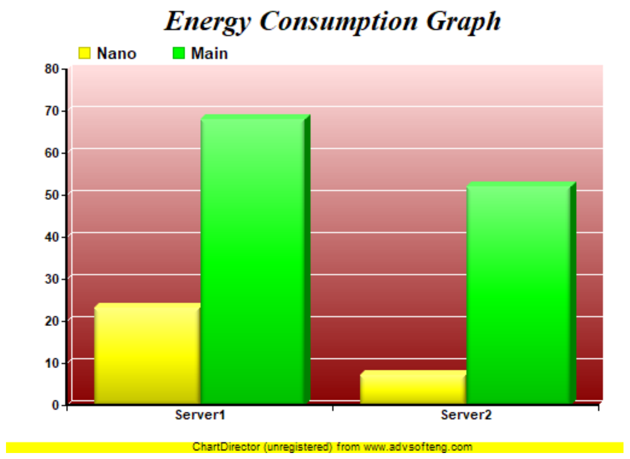
Wide acceptance of mobile phones and their resources applications have highlighted resource limitations of the devices. The placement of the cloudlets has a significant impact on the resource utilization of the cloudlets. Inappropriate placement of cloudlets can cause the severe imbalance in the edge cloud load. Some cloudlets are overloaded, while others are loaded or even idled and reducing the mobile users' response time requirements. In this, cloud computing has provided mobile devices with limitless resources in order to help them overcome their composure and enable them to support broad range of applications. So, mobile devices can outsource their tasks to local clouds. To gratify to exponential growth of requests, users request should be partitioned to different cloudlets and then clearly and actively redirected to the servers according to the latest network and server status. We have considered system performance and user quality of service parameters in this paper. Simulation results that, compared with other server selection schemes, the proposed scheme can achieve higher resource utilization, provide better user sight quality of service, and efficiently deal with network dynamics. Though the use of a cloud datacenter offers various benefits such as scalability and elasticity, its consolidation and centralization lead to a large separation between a mobile device and its associated datacenter. End-to-end communication then involves many network hops and results in high latencies and low bandwidth.

VII. FIGURES



The above diagram indicates the dataflow diagram of the DMCCP using Nano cloudlet clustering. This project is provided for producer side benefit. As the users requests for the servers based on the load status and the capacity the connection is done to its native server. If the native server is overloaded or in busy state, then the connection is redirected to other servers based on their load capacity and status. The admin can view the power and energy consumption the Nano servers. Admin is a super user who maintains the servers configurations, He has the rights Activate and Deactivate the Servers. The number of connections to the servers and the server status can be viewed.

VIII. GRAPHS



The above graph indicates the energy consumption of the Nano servers and the main servers. It describes that the Nano servers consumes less energy than the main servers and except one main server other servers can be considered as the Nano server. So the energy, power and the cost can be consumed.

IX. PESUDOCODE

- Step 1: Start
- Step 2: Network controller receive connection request from users.
- Step 3: Using IPLOOK_UP function get the NATIVE SERVER (NS)
- Step 4: Current_Load = Pidle(NS) + Current_No_Connection(NS) * Incremental Energy
- Step 5: Pick the Load threshold U(NS)
- Step 6: If U <= Current Load then Go to Step 10
- Step 7: Using Load Balance technique Shortlist Server (S) to transfer the connection.
- Step 8: Transfer the Connection to S
- Step 9: Go-to Step 11
- Step 10: Transfer the Connection to NS
- Step 11: Stop

The concept of skewness is to quantify the fluctuation in the utilization of multiple resources on a server. Let n be the number of resources we consider and ri be the adoption of the i-th resource. We define the resource skewness of a server p as

 DOI : 10.18231/2454-9150.2019.0576

$$skewness(p) = \sqrt{\sum_{i=1}^{n}(\frac{r_i}{\overline{r}} - 1)^2}$$

Where, r' is the mean utilization of all resources for server p. In practice, not all types of resources are performance perceptive and hence we only need to consider bottleneck resources in the above calculation. By reducing the skewness, we can combine different types of workloads pleasantly and improve the total utilization of server resources.

X. CONCLUSION

In this paper, we proposed an efficient heuristic DMCCP for fog server offloading. Our simulation results show that the proposed real-time DMCCP can be practically used to overcome limited resources of fog computing while its performance is almost close to the optimal method. For future research, we will try to balance the resource amount of cloudlets under required execution time. To demonstrate this project we need three systems. Based on the number of connections the energy consumption differs from one server to another server. When a connection request received in Gateway server, it has to fetch the IP-Address of the client and identify the Server for the IP Address using look-up function, and then Energy Threshold test has to be conducted. If the test pass then connection send to corresponding server, else based on the loads of all other server, one server is shortlisted and connection send to that shortlisted server.

REFERENCES

[1] Stojmenovic, Ivan, and Sheng-Wen. "The fog computing paradigm: Scenarios and security issues." Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on IEEE, 2014.

[2] Luan, Tom H., et al. "fog computing: Focusing on mobile users at the edge." arXiv preprint arXiv:1502.01815, 2015.

[3] S. Clinch, J. Hawkes, A. Friday, N. Davies, and M. Satyanarayanan, “How close is close enough? Understanding the role of cloudlets in supporting display appropriation by mobile users,” in Pervasive Computing and Communications (PerCom), 2012 IEEE International Conference on IEEE, pp. 122–127, 2012.

[4] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies. “The case for VM-based cloudlets in mobile computing.” IEEE Pervasive Computing, 8(4):14 –23, oct.-dec. 2009.

[5] Kwon K, Park H, Jung S, et al. "Dynamic Scheduling Method for Cooperative Resource Sharing in Mobile Cloud Computing Environments." KSII Transactions on Internet and Information Systems, 10.2: 484-503, 2016.

 DOI : 10.18231/2454-9150.2019.0576