Available online at http://www.journalcra.com

International Journal of Current Research
Vol. 9, Issue, 09, pp.58089-58094, September, 2017

INTERNATIONAL JOURNAL
OF CURRENT RESEARCH

ISSN: 0975-833X

RESEARCH ARTICLE

CLOUDLET ROUTING: INVESTIGATING THE DATA LATENCY IN CLOUD COMPUTING ENVIRONMENTS USING MULTIMEDIA CONTENT

*,[1]Abdullah Saleh Al-Saleh and [2]Omar Abdullah Al-Shaya

[1]Department of Information Engineering, Florence University, Via di S. Marta, 3, Florence, Italy
[2]Electronic Media Department, Saudi Electronic University, Abu Bakr Street, Riyadh, Saudi Arabia

ARTICLE INFO

Article History:
Received 19th June, 2017
Received in revised form
27th July, 2017
Accepted 15th August, 2017
Published online 30th September, 2017

Key words:
Cloudlet Routing,
Weight Based Routing,
Cost Based Routing,
Routing Techniques.

ABSTRACT

With popularity of smartphones and the explosion of mobile applications, it has become necessary for the mobile devices to function with high performance processing tasks in comparison to the static clients along with the servers. The computational resources of a device can be limited due to the weight, memory, battery life, and heat dissipation. To overcome the limitations in the mobile devices, studies have realized cloud computing as one of the best solutions. However, cloud computing may express few issues, like long-latency and expensive roaming charges for the access of cellular radio. Cloudlets can provide benefits over the distance cloud; for example faster data transfer, efficient application processing, and reduced utilization of mobile resources. This paper has examined the routes among cloudlets in the cloud computing environment. It has been concluded that the combination of cloudlets can provide significant support to the cloud computing infrastructure environment. It can further enable the model to have access to more reliable and faster connections that may also suggest new routes for faster and more efficient data transfer.

Citation: Abdullah Saleh Al-Saleh and Omar Abdullah Al-Shaya. 2017. "Cloudlet Routing: Investigating the Data Latency in Cloud Computing Environments Using Multimedia Content.", *International Journal of Current Research*, 9, (09), 58089-58094.

INTRODUCTION

Cloud computing is a paradigm shift emanating from the increased demand for elastic, efficient, and scalable systems that can meet the growing data volume. Cloud systems deal with real-time user demand and large-scale data processing, indexing, and extraction. Users of cloud-based systems such as enterprise resource planning, sales, and supply chain perform frequent read and write operations that increase resource demand in the cloud. Mobile computing and the Internet of Things increase the demand for cloud resources, which led to the development of cloudlets (Armbrust *et al*., 2010). A cloudlet is a small-scale data center that is found at the edge of the Internet whose role is to support mobile applications that require powerful resources (Quwaider, 2013). The presence of cloudlets reduces data latency between applications running in the cloud. They achieve this role by extending the cloud infrastructure, thereby, forming a 3-tier architecture. However, the effectiveness of cloudlets lies on the routing techniques that reduce data latency as information flows from one point to another. This paper examines cloudlet routing and the associated data latency in the cloud environment.

***Corresponding author:* Abdullah Saleh Al-Saleh**
Department of Information Engineering, Florence University, Via di S. Marta, 3, Florence, Italy.

Literature Review

Mobile cloud computing is a technology that combines mobile technologies and cloud computing (Fernando *et al*., 2013). Users access services hosted on the cloud without having to invest in expensive hardware that is needed to meet the extensive computing power needed to process hosted applications (Huang, 2011). Mobile devices overcome the limitation of computing resources by offshoring intensive computations to cloud servers. A device sends requests to the cloud, data is processed in the cloud, and the results sent back to the mobile device. In this way, mobile users can utilize software hosted in Software-as-a-Service models and developers can access Platform-as-a-Service model to market or deploy their applications (Asrani, 2013). In addition, users can stream high definition video and play mobile games with heavy graphics without crippling their memory and CPU resources. Cloud computing processes the bulk data and stores it while the mobile device acts as a client. Data latency is a significant problem in mobile cloud computing due to network congestion. Overloaded networks increase the time taken to request data from an application in the cloud to the mobile device (Leavitt, 2009; Shiraz, 2013). These delays are costly for enterprises running their business from the cloud.

Cloud computing is distributed and clients can request data from applications hosted in the farthest end of the world. In this case, data latency depends on the Internet traffic between the client and server and the bandwidth allocated to those services. In addition, virtualization increases latency since it adds another layer of hypervisors running virtual machines and communicating over virtual networks. As a result, there are packet delays as data travels over multiple layers before getting to the network to content with other Internet traffic.

Wireless networks have frequent intermittent data transfers and varying data rates due to challenges of coverage. Unlike wired broadband that has fixed physical links, mobile networks can experience longer latency due to weather and dynamic resource demand (Tandon, 2016). Further, processing time in the cloud and on the device can add data latency. As a result, it becomes difficult for mobile devices to access the cloud infrastructure directly. The problem is solved by deploying small-scale data centers at the edge of the Internet. These data centers are known as cloudlets and they allow mobile devices to access cloud applications without having to send request to the main cloud network. This reduces data latency and provides powerful resources for these devices by reducing the distance that data packets have to travel from the cloud infrastructure to the client. Nonetheless, cloudlets do not address the issue of data latency entirely despite their efficient routing protocols.

A cloudlet is an architectural element that occupies the middle section of a 3-tier hierarchy consisting of mobile devices, cloudlet, and the cloud (Jararweh *et al*., 2016). It serves the role of a data center that brings the cloud closer to the end user. Cloudlets have four primary attributes namely soft state, powerful, logical proximity, and standard technology. They exist in a soft state to avoid the management burden associated with a hard state. Further, they contain sufficient power that helps them offload intensive computations from mobile devices. Logical proximity refers to the ease of accessibility in a few hops compared to ordinary cloud applications. Nonetheless, cloudlets use standard cloud computing technologies presence in EC2 or OpenStack clouds (Ha, 2015). These characteristics allow mobile resource-intensive applications such as Apple's Siri to perform speech recognition in cloudlets rather than in the cloud that experiences network latency. In addition, cloudlets can be easily deployed as virtual machines in a container and they can be automated to autoscale and launch new containers when there is a hardware or software failure.

Cloudlets also support rapid provisioning since speed is a critical element when launching new machines or applications (Lewis *et al*., 2014). Cloudlets, unlike data centers, support agile deployment of virtual machine images. In this way, users do not have to upload custom images over the network to deploy a new cloudlet since they can deploy new virtual machines easily. In addition, fast hand-off in cloudlets increases transparency in mobile users. Services are handed over from one cloudlet to another as a way of maintain service consistency. The virtual machine hand-off capability avoids long response times as the logical distance between nodes increases. Resource constraints in the cloud lead to latency that affects business activities. Jitter and latency degrade services, which makes then unusable or they have poor performance (Satyanarayanan, 2013). In this way, speed in cloudlets is an admirable quality to the growing dependence of internet of things and mobile devices. Cloudlets reduce data latency by caching content, which reduces the calls made to cloud servers to fetch data (Tawalbeh, 2015). They also provide overlay routing and diverse paths that can be used to transfer data. Overlay routing is creating new routes on top of existing ones to connect cloudlets and service provides. This process provides better paths that data can use from one cloudlet to another and to the device. In this way, cloudlets help mobile devices avoid network latency that strains resources such as phone batteries. Satyanarayanan *et al*. argue that latency hurts user interaction by creating delays that interfere with the usability of a system (Satyanarayanan *et al*., 2009). In the cloud, minimal interactions with a user application can increase the execution time as a result of WAN delays and jitter. These parameters are difficult to control at the WAN level and they mainly affect intense applications such as augmented reality. Lagar-Cavilla *et al*. conducted a study that shows that latency leads to a negative impact on user interaction even on networks that have sufficient bandwidth (Lagar-Cavilla, 2007). In a different study, Tolia, David, and Satyanarayanan concluded that the perceived quality of thin-client depends on the end-to-end latency and application interactivity (Tolia, 2006). Despite the current innovation, WAN latency might not improve since companies mainly focus on security, energy efficiency, resource management, and bandwidth. In this way, techniques such as overlay networks and energy-saving modes in mobile devices increase end-to-end latency. As a result, network has increased tremendously over the years, but latency is a persistent problem.

Researchers propose multiple ways of addressing data latency in cloudlets. Elbamby, Bennis, and Saad propose the use of proactive networks that have content caches whereby the system anticipates the type of data that users might access and prefetch it during off-peak periods (Elbamby *et al*., 2017). This approach reduces the burden on backhaul links during peak periods when there is a high demand for data. On the other hand, Chamola, Tham, and Chalapathi propose the use of load balancing and latency-aware task assignment systems (Chamola, 2017). This deployment minimizes latency by avoiding overloaded networks.

Routing in Cloudlets

There are multiple routing techniques employed to transmit data in cloud computing. The type of routing has an impact on the latency since it determines the amount of hops that data travels before reaching the end user. The Distributed Hash Table (DHT) is a popular routing popular used in cloudlets and point-to-point networks (Tsu-Shiuan, 2012). In a DHT network, every file name is hashed and assigned to a unique key using a hash function. These hashes contain a key and the location of the file associate with that key. Every node in the network maintains a routing table that contains the location and files are retrieved from the peer closest to the requesting node. Although it is an effective protocol, DHT relies on broadcast to maintain consistency in the routing tables (Dabek *et al*., 2004). When a node leaves the network, the remaining ones have to update their routing tables to maintain the integrity of the network. In this way, there is a scalability issue and protocol is slow since it uses both network and application layers. In addition, network latency occurs due to regular updates as nodes join and leave the network.

Improvements in the traditional DHT protocol implement a DHT algorithm in the network and application layers. This approach reduces the size of routing tables and the time taken to update them (Tsu-Shiuan, 2012; Kuppusamy, 2013). However, the tables in the improved implementation increase latency as they require frequent updates in both layers. Further, merging the tables in the routing and application layers into one improves the performance of this protocol. The combined table contains a key and route entry, which reduces scalability by eliminating the need to maintain two tables. The time taken to retrieve the data and update the tables is also significantly reduced.

Cloudlets compute resource utilization and optimize their parameters to sustain their quality of service by analyzing metrics such as cost and latency on various routes. In some cases, these cloudlets are located close to each other and they support mobile users within a small geographical area. An example of these cloudlets is the mobile cloud hybrid architecture (MOCHA) that can be deployed in an office complex (Mouftah, 2013). This cloudlet can route data to other cloudlets using broadcast techniques over a peer-to-peer network. Broadcast involves sending messages to all nodes in the application and network layers. Every peer in the network has to listen for request from other nodes, which leads to higher data latency compared to DHT protocols. As a result, broadcast is best suited for cloudlets deployed in small networks whose purpose is to test components or transmit data to a limited number of nodes. Wireless technologies improve broadcast routing protocols by limiting it to the application layer (Macharapu, 2017). By default, wireless networks have to broadcast data over the network layer to maintain network consistency. However, limiting cloudlet data routing to the application layer increases the interaction between the network and applications, but there are challenges when the network is significantly large. The networks relying on broadcast routing are flooded by broadcast messages leading to competition with user data. According to Mouftah, cloudlets can use distributed or centralized routing (Mouftah, 2013). In the former, the cloudlets create and maintain the routing table and they periodically broadcast their information to neighboring cloudlets and nodes. Mobile devices also broadcast their user IDs to the available cloudlets and they store the strongest cloudlet ID in their tables. In this way, devices minimize data latency by querying the closest cloudlets. In centralized routing, a central server creates and maintains the routing table. Each cloudlet sends its ID, that of mobile users, and those of neighboring cloudlets to the central server. Simulations conducted on both architectures indicate that distributed routing has lower data latency than centralized routing.

Apart from broadcast and DHT, Sarddar proposed a weight-based routing protocol (Sarddar, 2015). The author proposes a weight-based routing algorithm that selects routes depending on the demand. Improper routing increases data latency in cloudlets and, therefore, the optimal route should be selected depending on the cost and capacity of available links. In this protocol, base stations have link managers whose work is to communicate with the switching center to access link characteristics. The link manager computes the weight factor of every link depending on the cost and capacity. Links with the highest weight are selected as the optimal routes for transmitting data (Prasanna, 2015). The routes are computed every time a mobile device joins the network and selects a base station. However, they are activated when there is data to be transmitted over the network and terminated when the communication ends. This process provides resources on-demand and frees them up for use by other transmitting devices. Traditionally, networks choose the shortest path, but this model can overload a link when there is extensive demand. In the weight-based route selection, the link is not necessarily the shortest path, but it is selected based on the parameters of cost and the available capacity. In this way, a longer path that has the capacity to carry data can reduce data latency compared to selecting the shortest path. Prasana and Keshava argue that data latency in cloudlets can be minimized by deploying cognitive data routing (Prasanna, 2015). The algorithm mainly solves bottleneck problems that increase the time that data packets take to traverse the networks. Cloudlets are heterogeneous and they contain multiple infrastructures, hardware, and technologies. The authors propose a routing protocol that runs in an LTE network that contains a base station and devices. LTE has a high bandwidth and it supports both time division and frequency division duplexing. In this ad-hoc network, the nodes avoid bottlenecks by switching the mobile devices from one infrastructure to another. In this case, the access model changes from cloudlet, to client-server, or to the ad-hoc model depending on the resources available. The outcome is an improved quality of service, reduced data latency, and increased network capacity.

Macharapu states that data latency can be addressed by reducing the number of data packets to be transmitted over a network (Macharapu, 2017). The solution proposes a routing protocol that combines a set of packets into a single one to be transmitted to the end device. The aim is to reduce the energy and delay experienced when transmitting multiple packets over a network. Combining them into one packet reduces transmission energy in both Wi-Fi and cellular communications. The data collection protocol is suitable for users in motions who need to access the enterprise cloud system through Wi-Fi or cellular mobile networks. These networks have varying capacities and combining data packets effectively minimizes the energy and time taken to transfer data.

DISCUSSION

Cloudlets provide a 3-tiered architecture that allows mobile devices to access data and cloud computing resources. However, data latency is still an issue in mobile cloud computing despite the development of numerous routing protocols that aim at streamlining the flow of data from one device to another and between cloudlets. Routing across multiple networks requires protocols that can select the fastest and most economical path. This requires algorithms that can compute multiple characteristics of the available network and select the path that is most economical. In this way, the weight-based routing protocol proposed by Sraddar has the highest probability of generating the least data latency in the cloudlets (Sarddar, 2015). The proliferation of Internet of things has increased the number of devices seeking to access cloudlets and some of them are highly mobile. This makes it necessary to compute the resources and cost needed to maintain a link and transmit data over the most economical path. The fact that this protocol provisions these links on-demand increases its applicability since cloud services are traditionally accessed on demand.

On the other hand, distributed routing protocols perform better than centralized ones. Research conducted by Lin (2012) indicates that distributed hash tables are faster than centralized ones. In addition, DHT has lower latency compared to broadcast routing since the latter relies on constantly sending notifications to other nodes to maintain network consistency. However, a distributed hash table protocol maintains consistency over a shorter distance since the nodes only send the keys and the locations of data files. Mouftah reiterates the importance of distributed routing in cloudlets over centralized routing due to increased overhead in the latter (Mouftah, 2013). Hosting information in one location increases data latency as every node has to query the same source. However, distributing the routing tables creates multiple locations that nodes can obtain the necessary data.

The ideal solution to data latency is to deploy a routing protocol that maintains distributed tables and sends data over the most economical link. Mobile devices have limited resources which makes the issue of power and path cost necessary. Saddar states that link managers continuously compute routes depending on demand (Sarddar, 2015). In this case, data packets from distributed locations take the most economic route that is then made available once complete. This approach overcomes the shortcomings of broadcast protocols that are only effective in small networks and send data using the shortest link. Consequently, links have different capacities based on the available bandwidth, which creates the need to balance data transmitted across a network. In moving devices, cloudlets need to perform hand-off as the user moves across different networks. The parameters of these networks might change depending on the infrastructure, which can affect the quality of service. Transmitting data over a weight-based link provides a more potent ways of adjusting capacities to meet the data requirements as users move across cities and institutions.

Data latency hugely depends on the routing protocol deployed in a cloudlet and researchers have proposed numerous algorithms that can reduce the time taken to transfer data packets over a network. The issue of bottleneck in routing introduced by Prasana and Keshava can be avoided when cloudlets transmit data over a weighted link that has the capacity to carry that data without increasing the overall cost (Prasanna, 2015). Therefore, cloudlets can maintain their logical proximity features by maintaining a distributed routing table and address resource constraints and latency by transmitting data over links that have the necessary capacity as well as the lowest cost.Cloud computing companies mainly focus on security, trust, bandwidth, energy use, and resources. However, improving routing can reduce the time taken to access data in the cloud and create satisfactory user interactions. Therefore, companies and academia need improved routing protocols that are secure, energy efficient, and are intelligent enough to deploy links when needed to avoid latency experienced when retrieving data.

Implementation and evaluation

In accordance with the main objective of this study, the communication between the cloud servers and clients via cloudlets has been observed. Cloudlets function as in intermediary between the clients and cloud distance servers. In order to simulate a real cloud environment and to obtain as accurate results as possible, the implementation deployed 5 virtual machines (1 cloud VM server, 3 VM cloudlets and 1 cloud VM client) over a public network between USA, Italy and Saudi Arabia. The remote cloud VM server was installed in Seattle (USA) with IP address of 67.161.106.95. The model possessed following characteristics: 2.67 GHz Intel Core i5, 1 GB RAM and 20 GB disk, and ran Windows Web Server 2008 and VMware Workstation 9.0. 3 VM cloudlets were installed in Florence, Italy, with shared IP address (79.52.120.186). They had the following characteristics: 1.70 GHz Intel Core i3-4005U, 512 RAM and 10 GB disk, and ran Windows Web Server 2008 and VMware Workstation 9.0. The VM client was installed in Medina, Saudi Arabia, with IP address of 95.187.39.3. It had the characteristics of 2.93 GHz Intel Core 2 Duo CPU, 512 RAM and 5 GB disk, and ran Windows 7 and VMware Workstation 9.0. The high-level architecture of the experiment is shown below in Figure 1.

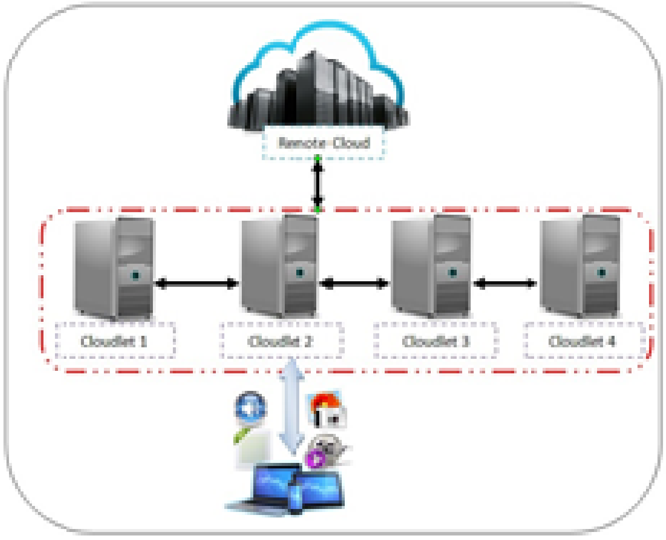

Fig. 1. The High-Level Architecture of the Experiment

The work has explored the details of implementation for in-depth study of the model. The remote cloud server has been established within a specific machine and at a specific port. The cloudlets have been started in the position where each cloudlet was aware of the location and port of the remote cloud server. Likewise, cloudlets were also aware of the port and location of the other cloudlets. The work has tested the routes between cloudlets and the remote cloud server. It compared the elapsed times and selected the fastest route. The model performed in a following manner:

- The client asks the cloudlet 1 for the route for some multimedia files.
- If the cloudlet 1 has the requested files, it returns the route (cloudlet 1 IP address).
- If it does not have that files, the cloudlet 1 tries to download the files from many other cloudlets, compares and sends the fastest path to the client.
- If the cloudlet 2 has that files, it sends files to the cloudlet 1.
- Then, the cloudlet 1 returns the route "(cloudlet 1 IP address), (cloudlet 2 IP address)" to the client.
- If cloudlet 1 asks the remote cloud server, and it is the fastest route, then the remote cloud server sends that files to the cloudlet 1 and the cloudlet 1 returns route for the client "(cloudlet 1 IP address), (remote cloud server IP address)".

The results were collected from 5 virtual machines: 1 as a remote cloud server, 3 as cloudlets, and 1 as client. Screenshot of fastest available route to the client has been obtained for the observation, which has been displayed in Figure 2.

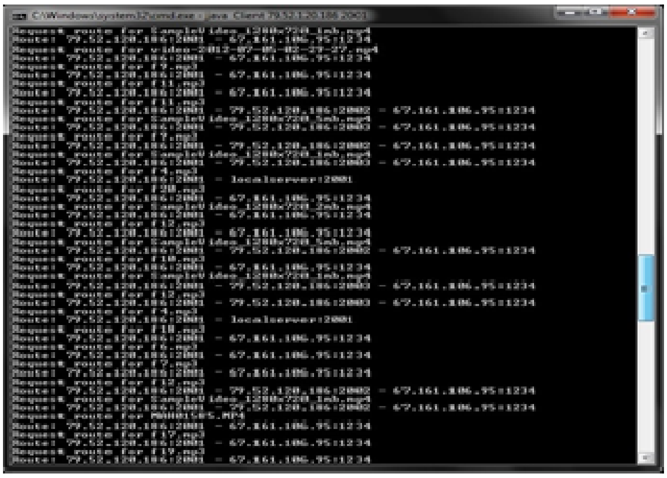

Fig. 2: Capture of a Client Session, Getting the Fastest Route

Conclusion

Cloudlets sit at the edge of the Internet and provide mobile devices with a faster way of accessing cloud computing resources. They reduce the hops needed to access the cloud by occupying the middle layer of the 3-tiered architecture. However, data latency in cloudlets is still an issue mainly due to the routing protocols employed. Broadcasting protocols are suitable for small networks, but the constant updates among nodes throttle the bandwidth needed to transmit data. DHT overcome the problem of broadcasting by maintaining key values in every node that identify the location of data files contained in that network. Further, a weight-based routing protocol selects routes depending on the capacity of a link and the costs of the path needed to transmit data. This overcomes the shortcomings of relying on the shortest path that can be overwhelmed by a high data flow. However, combining distributed routing tables and weighted routing can reduce data latency in cloudlets further. Applications would send data over links with the necessary capacity and distributed tables reduce calls to the same location.

REFERENCES

Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I. and Zaharia, M. 2010. “A view of cloud computing,” *Commun. ACM*, vol. 53, no.4, pp. 50-58, 2010.

Asrani, P. 2013. “Mobile Cloud Computing”, *International Journal of Engineering and Advanced Technology*, Vol. 2, no. 4.

Ch. S. Prasanna, and M. C. Keshava, 2015. “Data Routing Algorithm in Mobile Cloud Computing Network” in *International Journal of Engineering Research & Technology (IJERT)*, vol. 4, no. 4, pp. 915-918.

Chamola, V., Tham, C. K. and Chalapathi, G. S. S. 2017. "Latency aware mobile task assignment and load balancing for edge cloudlets," *IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pp. 587-592.

Dabek, F., Li, J., Sit, E., Robertson, J., Kaashoek, M. F. and Morris, R. 2004. "Designing a DHT for Low Latency and High Throughput." in *NSDI*, vol. 4, pp. 85-98.

Elbamby, M. S., Bennis, M. and Saad, W. 2017. "Proactive Edge Computing in Latency-Constrained Fog Networks," in *arXiv, preprint*, vol. 1704.06749.

Fernando, N., Loke, S. W. and Rahayu, W. 2013. "Mobile cloud computing: A survey," *Future generation computer systems*, vol.29, no. 1, pp. 84-106.

Ha, K. and Satyanarayanan, M. 2015. "Openstack++ for cloudlet deployment," *School of Computer Science Carnegie Mellon University Pittsburgh*.

Huang, D. 2011. "Mobile cloud computing." In *IEEE COMSOC Multimedia Communications Technical Committee (MMTC)* E-Letter 6, no. 10, pp. 27-31.

Jararweh, Y., Doulat, A., AlQudah, O., Ahmed, E., Al-Ayyoub, M. and Benkhelifa, E. 2016. "The future of mobile cloud computing: Integrating cloudlets and Mobile Edge Computing," *23rd International Conference on Telecommunications (ICT)*, pp. 1-5.

Kuppusamy, K. and Mahalakshmi, J. 2013. “A Survey on Routing Algorithms for Cloud Computing”, in *International Journal of Computer Applications (IJCA), Proceedings on International Conference on Computing and information Technology (IC2IT)* (4): 5-8.

Lagar-Cavilla, H. Andrés, N. Tolia, E. Lara, M. Satyanarayanan, and D. O'Hallaron, 2007. "Interactive resource-intensive applications made easy," In *Proceedings of the ACM/IFIP/USENIX International Conference on Middleware*, pp. 143-163.

Leavitt, N. 2009. "Is Cloud Computing Really Ready for Prime Time?," in *Computer*, vol. 42, no. 1, pp. 15-20.

Lewis, G., Echeverría, S., Simanta, S., Bradshaw, B. and Root, J.2014. "Tactical Cloudlets: Moving Cloud Computing to the Edge," *IEEE Military Communications Conference*, pp. 1440-1446.

Macharapu, M. 2017. “Survey on Efficient Routing Techniques in Cloudlets,” in *International Journal of Advanced Engineering Research and Applications (IJA-ERA)*, vol. 3, no. 3, pp. 117-124.

Mouftah, H. T. 2013. “Communication Infrastructures for Cloud Computing.

Quwaider, M. and Jararweh, Y. 2013. "Cloudlet-based for big data collection in body area networks," *8th International Conference for Internet Technology and Secured Transactions (ICITST-2013)*, pp. 137-141.

Sarddar, D., Bose, R. and Sahana, S. 2015. "A novel approach on weight based optimized routing for mobile cloud computing," in *Brazilian Journal of Science and Technology*, vol. 2, no. 1, pp. 1-12.

Satyanarayanan, M. 2013. "Cloudlets: at the leading edge of cloud-mobile convergence,"In *Proceedings of the 9th international ACM Sigsoft conference on Quality of software architectures*, pp. 1-2.

Satyanarayanan, M., Bahl, P., Caceres, R. and Davies, N. 2009. "The Case for VM-Based Cloudlets in Mobile Computing," in *IEEE Pervasive Computing*, vol. 8, no. 4, pp. 14-23.

Shiraz, M., Gani, A., Khokhar, R. H. and Buyya, R. 2013. "A Review on Distributed Application Processing Frameworks in Smart Mobile Devices for Mobile Cloud Computing," in *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1294-1313.

Tandon, R. and Simeone, O. 2016. "Cloud-aided wireless networks with edge caching: Fundamental latency trade-offs in fog Radio Access Networks," *IEEE International Symposium on Information Theory (ISIT)*, pp. 2029-2033.

Tawalbeh, L., Jararweh, Y. and Dosari, F. 2015. "Large scale cloudlets deployment for efficient mobile cloud computing," *Journal of Networks*, vol. 10, no. 1, pp. 70-77.

Tolia, N., Andersen, D. G. and Satyanarayanan, M.2006. "Quantifying interactive user experience on thin clients," in *Computer*, vol. 39, no. 3, pp. 46-52.

Tsu-Shiuan. Lin, "File sharing in Mobile Cloudlet networks," 2012.

****