International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 9, No.4 (July-2020)

http://dx.doi.org/10.12785/ijcds/090420

Multilayer Software Defined Networking Architecture for the Internet of Things

Hushmat Amin Kar[1, *] and G. M. Rather[1]

[1] *Department of Electronics & Communication, National Institute of Technology Srinagar, India*

Received 5 Mar. 2020, Revised 23 May. 2020, Accepted 24 Jun. 2020, Published 1 Jul. 2020

Abstract: The exponential growth in the devices connected to the internet of things (IoT) has raised lot of challenges for existing network architecture in providing support to time-constrained applications, device mobility, data management, etc. Fog computing enhances network performance by providing features like reduced response time, network and data security, etc. On the other hand, Software-defined networking (SDN), which slices out control and data planes, provides a platform for efficient network management. This strategy provides a flexible control mechanism for implementing policies of data and device management. In this paper, a new IoT architecture has been proposed which combines SDN features with fog computing so as to enhance the overall performance and management of network systems. The proposed framework also introduces an additional layer within the cell, which provides services to the IoT devices, thus reducing the latency and performs load balancing in the network. The new proposed IoT network architecture has been simulated using MININET simulator for evaluation of round-trip time (RTT) of each layer. The simulation results show a remarkable improvement in the latency of the proposed IoT architecture.

Keywords: IoT, Edge computing, SDN, Fog computing.

1. INTRODUCTION

Advances in communication technologies and micro-electro-mechanical systems (MEMS) have resulted in the exponential growth of internet-enabled devices. This evolution inspired the idea about the Internet of Things (IoT)—a large-scale cognitive system in which a wide variety of ‘‘things’’ could be connected. The definition of ‘‘thing’’ is very flexible and may refer to intelligent machines, drones, self-driving cars, sensor nodes, etc. having varying degrees of sensing, processing and actuation capabilities. These have the ability to communicate and interoperate through the internet. IoT is a dynamic infrastructure, providing self-identifiable adaptive capabilities in end devices, in order to make them intelligent. These devices recognize the triggers in the surrounding environment and accordingly react in an appropriate manner. This new environment is an evolving technology that is expanding its horizons in different areas at a very fast rate and is a key enabling technology for new future digital applications. In 2014, approximately 3.9 billion connected devices were in use and this figure is expected to rise to 50 billion connected devices by 2020, with over 200 billion intermediate connections [1]. Once in operation these devices will generate data of the order of petabytes per day that needs to be processed, stored, analyzed, and presented efficiently to the end users. [2].The generic framework of IoT architecture as proposed by the International Telecommunication Union (ITU) consists of 5 layers viz; Perception, Network, Middleware, Application, and Business layers as shown in Figure 1 [3]. The perception layer consists of physical objects and sensor devices. This layer keeps track of the unique identification, status, and management of each device. The next immediate layer is the network layer, which incorporates IP addressing and routing mechanisms. It provides a means for the transfer of collected information from sensor devices to the middleware layer for offering various services. The transmission links can use any wireless technology like 4G, 3G, UMTS, Wi-Fi, Bluetooth, ZigBee, Satellite, etc. or wired technology. Middleware Layer performs functions like Storing of lower-level information in the database, Service management, information processing, ubiquitous computation and information retrieval. This layer has a communication interface with the application layer, which is responsible for application

E-mail:hushmat.kar@ieee.org, ghulammohdrather@yahoo.com

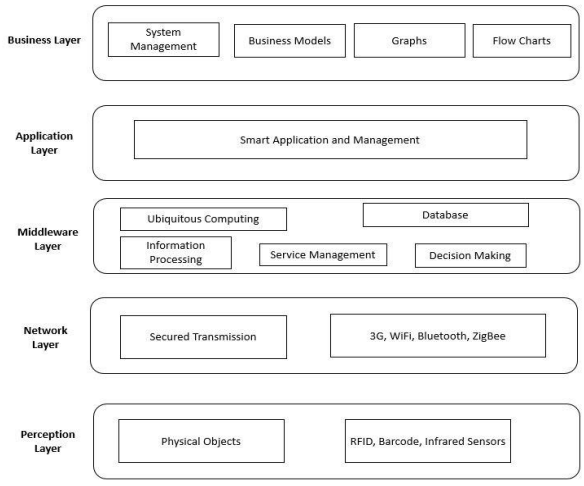


Figure 1. Architecture of IoT

management as per the processed information in the middleware layer. The applications include smart healthcare, smart city, smart transport, etc. [4-7]. The topmost layer of the IoT structure is the business layer. At this layer information received from lower layers is analyzed. The knowledge extracted therefrom is helpful for the executive to take accurate decisions about business strategies for optimizing the benefits of IoT technology.

2. CURRENT STATE AND CHALLENGES

The IoT devices collect information about the environment by using specified sensors. Most of the IoT devices have limited capacity for information storage, data processing, and energy resources. Secondly, every device is designed with specific hardware for a particular application and modifying the functionality of the device in real-time is a complex task. Power management techniques are employed in situations where limited power resources are available[8]. But the volume, variety, and rate of data generation by the IoT devices create a lot of burden on the existing static network infrastructure [9,10]. Although monitoring and sensing technologies have achieved some level of maturity, IPv6 has been proposed as a solution to identify IoT objects independently. But the standards are missing for mapping the sensed data (through 6LoWPAN) into an IPv6 header. Many general routing issues still need to be addressed like how to cope with the heterogeneity of the device capabilities that affect the amount of information that can be stored and used for computing optimal paths. Novel reliable transport layer protocols are needed to cope with the congestion issues that may arise due to the scale of the network, in order to provide end to end reliability. Unavailability of open software solutions for security, privacy, and device management that allows IoT devices to Seamlessly discover and identify the IoT devices manufactured by different manufacturers.

Due to these issues, challenges like handling of huge data, the requirement of links with large bandwidth, high latency, information, and network security and incompatibility among heterogeneous devices [11-15] have arisen while implementing IoT using existing internet infrastructure and protocols.

To overcome these issues, a number of solutions have been proposed by researchers so far which include:

1. Adding more resources in terms of memory, processing power, and communication link capability, but this is not economically feasible in the long run.
2. Efficient traffic management algorithms to handle the transfer of large data across the network entities [16-20].
3. Introducing an intermediate Fog computing layer between devices and cloud [21].
4. Providing security strategies, authorization schemes, etc. to counter Denial of service, wormhole attack, etc. [22-25].
5. Efficient schemes for the handling of data generated by incompatible devices [26-29].

However, all the solutions proposed so far solve a particular challenge and there is a need for a comprehensive strategy that can efficiently utilize network resources. Software-defined networking along with Fog computing can be one such approach.

3. FOG COMPUTING

Cloud computing can provide the requisite services to IoT devices for different applications. Here the IoT device sends the sensed data to the cloud and on receiving the data, cloud processes it and sends the requisite information back to the intended device/system, for taking an appropriate action [30-33]. This approach may prove to be cost effective, provided the number of connected devices is within a reasonable limit. However, due to the increasing number of devices, the approach suffers from longer response time, comprising of propagation and queuing delay. Due to this, the system suffers in providing its services to the time-critical IoT applications. The concept of fog computing has been introduced as shown in Figure 2, wherein an intermediate service layer is introduced between IoT devices and the cloud, to provide the services like data processing, storage, decision making and analytics to the IoT devices at the edge of the network, thereby reducing the propagation delay [34]. The queuing delay can be minimized by using the efficient scheduling algorithms so that traffic load gets uniformly distributed among the network resources.

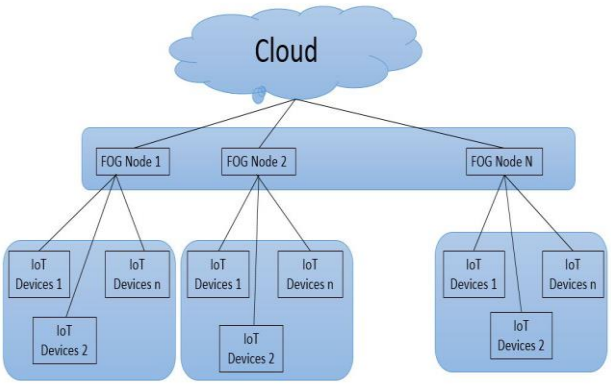


Figure 2. Role Of FOG layer in IoT networks.

To provide services with less response time, the data from the IoT device is sent to the nearest fog node. However, in some situations, the fog node may not be able to serve the request of the IoT device due to the non-availability of resources and service. In that situation, the fog node will redirect the request to the cloud and the cloud provides the requisite services to the devices through the fog layer. The fog node will also filter the important data collected from the IoT devices and send it to the cloud for long term storage and batch analytics.

The introduction of the Fog layer in the cloud-to-thing consortium results in benefits like lesser response time, efficient bandwidth utilization, efficient and cost-effective support for resource-constrained devices, enhanced security environment, and uninterrupted services.

4. SOFTWARE-DEFINED NETWORKING

In the existing internet infrastructure, each network node consists of two layers i.e. control and data plane. The functions of the control plane involve routing, signaling traffic, system configuration, and management. The role of the data plane is to transfer the packets to the desired destination. The network build by these nodes can be viewed as the combination of various heterogeneous and autonomous nodes. Thus, the traditional internet is static and cannot be configured as per requirements of the fast-growing IoT applications because, for a particular application, the device needs to be manufactured with specific hardware architecture and requires preprogramming to achieve the desired task [35,36]. Therefore any change in the forwarding policy requires the reconfiguration of each network node individually. Secondly, the network is complex and very hard to manage as both control and data planes are bundled inside the node which reduces its flexibility and hinders network evolution [37-39].

Software-defined networking is a new paradigm, which can help to address such limitations in the current network infrastructure as it provides a clear separation of control and data planes as shown in Figure 3 [40]. With this separation the data plane simply comprises of data forwarding devices and control is shifted to the main controller which is known as SDN controller. This defines network configuration and ensures policy enforcement.

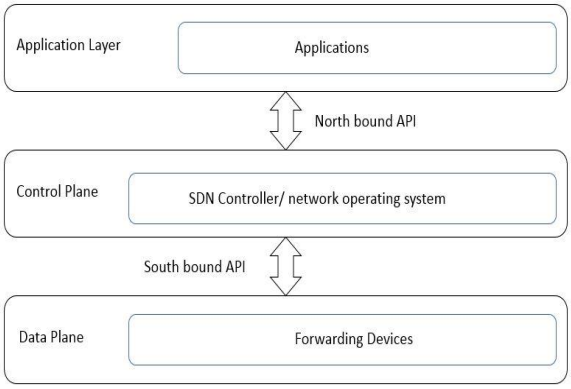


Figure 3. SDN Architecture.

Open Flow protocol [41] is used for achieving communication between the SDN controller and the data plane. Using this protocol, the controller can proactively or reactively instruct the data plane devices about the identification and treatment of different data flows in the network [42]. Once the SDN controller instructs the data plane devices like OpenFlow switches about handling of the traffic flow, then these devices are able to handle the packets belonging to the flow without the controller's intervention until the time expires for that instruction. Thus, making the devices behave differently as per the situation like a router, switch, firewall etc.

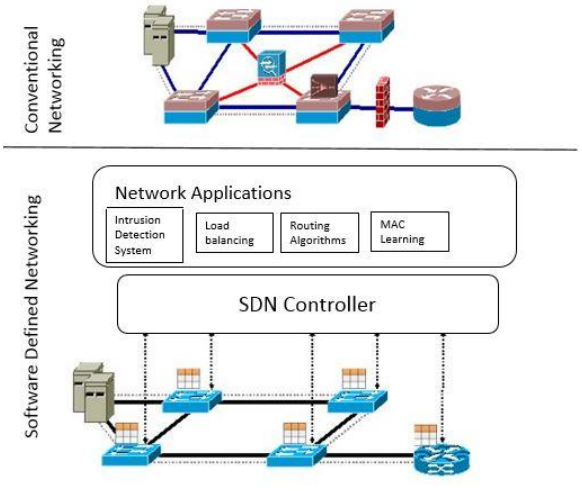


Figure 4. (a) Conventional Networking (b) Software Defined Networking.

Figure 4 shows both Conventional and Software-defined networking. In SDN the management becomes simpler and middleware services can be delivered as SDN controller applications. The benefits that can be achieved by using SDN in IoT are better network management, virtualization of network function, efficient resource utilization, better energy management, dynamic resource allocation, and efficient security policies. [43-54]

Many researchers have analyzed the improvements in the IoT system by using SDN. Caragray et al [55] have analyzed various benefits of using SDN in IoT. Also, they have discussed the need for SDN at various levels viz edge, access, data centers, and core. Jagandeesan et al. [56] analysed the support for OpenFlow protocol in the existing wireless network and discussed various benefits of using SDN in wireless networks. Sood et al [57] have discussed the use of SDN in the IoT system and its enhancements to an optical and wireless network so that they can be effectively used with SDN and IoT. Jararweh et al [58] have proposed an architecture for IoT which incorporates SDN called SDIoT.The architecture mainly focusses on data protection during its transmission and storage. Sahoo et al [59] have provided an architecture for IoT, providing a secure environment for IoT nodes. In this architecture, IoT nodes are required to get authentication from the border controller for communicating with other nodes. Flauzac et al [60] have proposed an architecture for IoT, which supports both infrastructure as well as infrastructure-less networks. Chakarbarty et al [61] have proposed SDN based architecture for smart cities. The architecture keeps track of security, routing, and identity management for IoT communication. Balfour et al [62] proposed SDN based security architecture for IoT using a protocol, named as software defined perimeter (SDP). In this model, whenever an IoT device wants to communicate it has to get authentication by configuring SDP. An approach has been proposed by Tajiki et al [63] known as CECT for time critical IoT applications. CECT ensures efficient traffic management, Better network throughput and good QoS to the end users by using SDN in data centers. Ukil et al [64] have proposed an SDN based architecture for embedded system which uses IoT agents that are running on IoT devices. The SDN controller provides the agents with routing information and authorization for an efficient and secure communication. Mauro et al [65] have proposed a cloud enabled SDN based IoT architecture known as CENSOR. In this architecture also includes the security module which provides security and reliable communications in IoT networks. Hence using SDN in the IoT network will definitely improve the overall performance of the system.

5. PROPOSED ARCHITECTURE

This paper proposes a framework for the Internet of things environment as shown in Figure 5, which consists of 3 layers viz cloud, fog, and devices, having SDN features incorporated. The main features of the proposed architecture are the use of SDNC in conjunction with edge Computing which helps in enhanced resource utilization, improved quality of service (QoS), quality of experience (QoE) especially for time-constrained applications and network function virtualization (NFV) to provide dynamic cost-based routing.

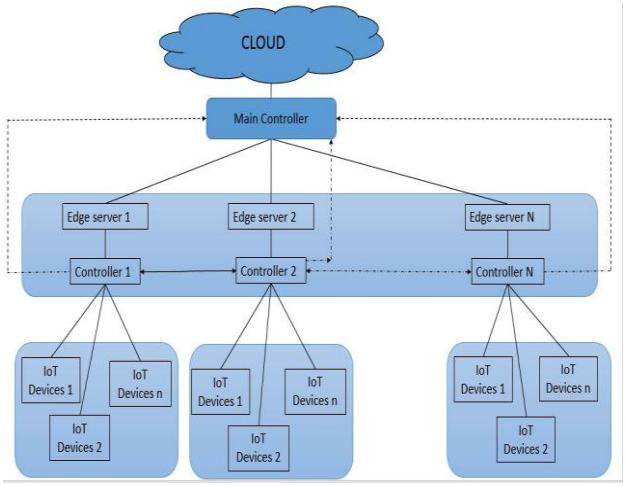


Figure 5. Proposed architecture for IoT networks using Fog Computing in Wide Area Networks using customized SDN Controllers

The local networks in this architecture are termed as cells. The network cells are connected to the edge servers through SDNC's. These SDNC's are having their separate network with each other using east-bound and west-bound interfaces. Also, each SDNC is connected to the main SDNC controller, thus making the proposed architecture distributed as well as centralized. The SDNCs are connected to the devices and cloud using south-bound and north-bound interfaces respectively. Each SDNC in this framework is state aware of its connected devices (all the nodes of the cell) and the service provided by other edge servers, which is available in each SDNC in the form of SYNC STATUS table as shown in Figure 6. The advantage of using SYNC STATUS table is that we can directly send the request to the edge server capable, available, and cost-efficient for handling the request. The SDNC categorizes the devices within a cell into two types namely: consumer devices and service providing devices. This categorization among the cells helps in achieving maximum resource utilization and efficient network management. This logical server defined henceforth can be utilized in addressing the requests of neighboring devices within the cell.

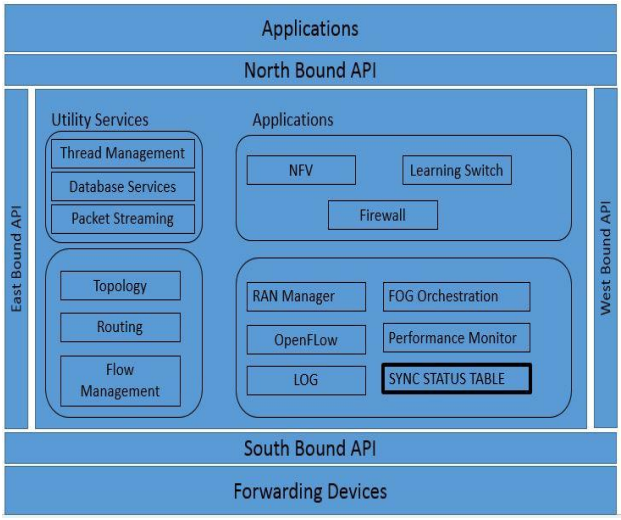


Figure 6. Modified SDN Controller Architecture

A. *Request handling mechanism in proposed Architecture*

The request generated by an IoT device can be served in any layer of architecture depending upon the available resources required, as shown in Figure 5, according to one of the following scenarios:

1. The request is addressed by the immediate Edge Server.
2. The request is addressed by a neighboring node within the cell.
3. The request is addressed by the next Edge Server.
4. The request is redirected to the Cloud if the Fog layer is not able to process the request.

Scenario 1: When IoT device takes services from the Edge Server

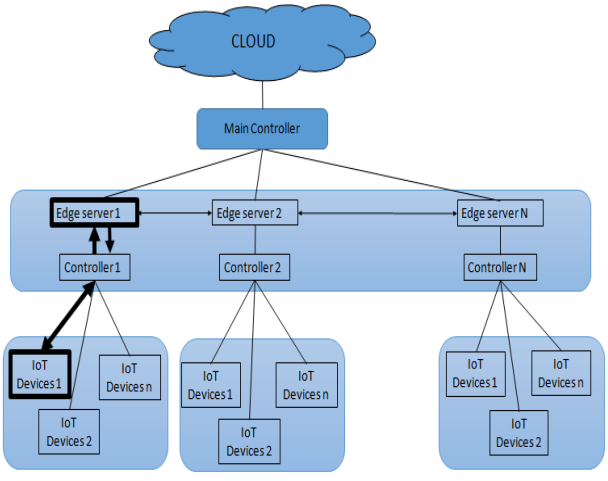


Figure 7. IoT device accessing services from the Edge Server

In this scenario, a node initiates a request for service, the SDN controller determines the nature and category of the request. The controller checks the SYNC STATUS table and determines the status of the edge. If the Edge server is capable of handling the request, the SDNC forwards the request to the immediate edge server for requisite services as shown in Figure 7.

Scenario 2: IoT device takes services from the logical node

In this case, a node initiates a request, the SDN Controller determines the nature and category of a request. If the SDNC finds out that the edge server is unable to serve the request, then it checks the availability of services among its pool of connected devices (within the cell). If the request can be handled, the controller sets up a connection between the source and the target node. The request is forwarded to the target node which then offers its services to the requesting node.

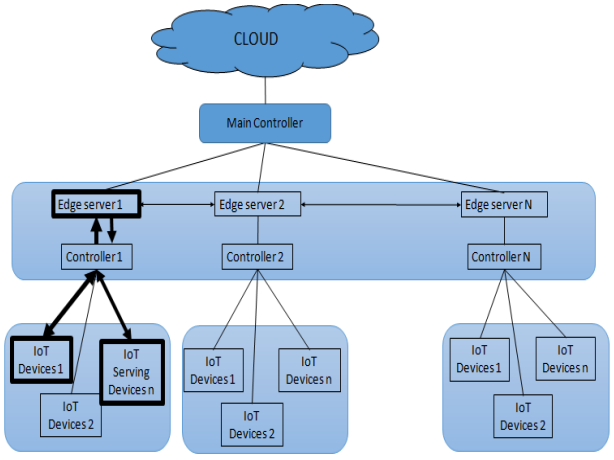


Figure 8. IoT device accessing services of a neighboring device

The choice of a node is dependent upon the type of request generated. Each SDN controller logically labels the devices based on the services a device can provide, therefore, it is SDNC that dynamically chooses the route based on the best routing algorithm (shortest path). The scenario is shown in Figure 8.

Scenario 3: When IoT device takes services from the neighboring Edge Server

When the SDNC finds that the requisite service of the IoT device can be provided neither by the immediate edge server nor by the logical server within the cell, in this case, the request is forwarded to the neighboring edge server, where this service is available as per the SYNC STATUS table. The scenario is depicted in Figure 9.

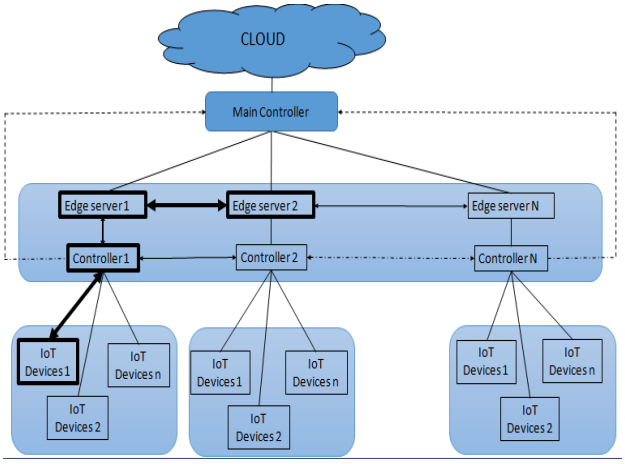


Figure 9. IoT device accessing services from neighboring Edge Server

Scenario 4: The Cloud provides the services to the IoT device

In this case, a node initiates a request, the SDN Controller after determining the nature and category of a request determines that neither the logical server nor the Fog layer is able to process the request. Then, the request is forwarded to the cloud and the IoT device utilizes the services of the cloud. Figure 10 shows the scenario.

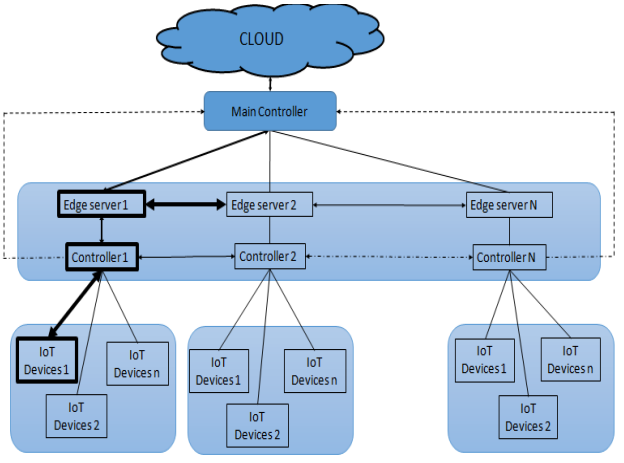
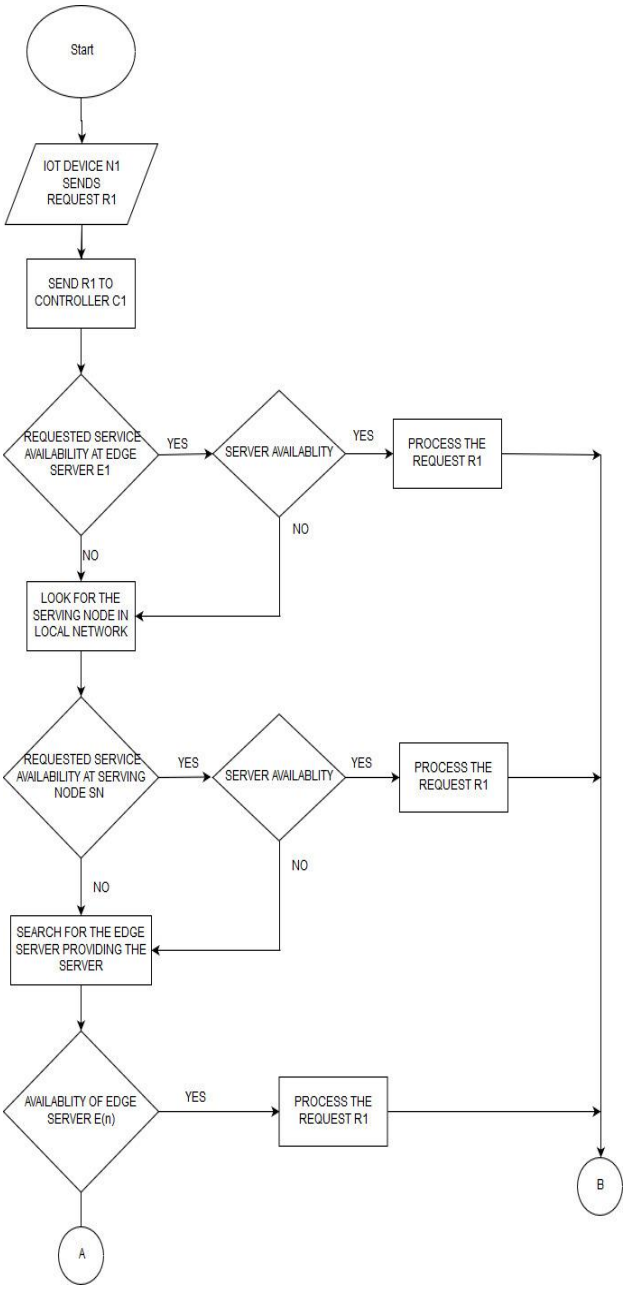


Figure 10. When Fog layer is unable to process the request

The controllers logically have a centralized view of the network and can utilize the resources to the maximum limit. These controllers have all the information required to find the best path for the packets and thus implement efficient traffic engineering and load balancing techniques. These routes in all the above scenarios are dynamically calculated. The cost comparisons for routing will be updated dynamically in real-time and at each of the SDN controllers throughout the network. At the time of routing each SDN controller is configured to calculate the weight of the link to be established and compare with the weight of the link if the requests are routed to the cloud. If at any point in time the weight of the network redirect exceeds the cloud redirect. The request is forwarded to the Cloud and the cloud services are accessed to address the request. The routing process can be explained by the below-shown process.

B. Request Flow Cycle

The request flow cycle can be best represented by the following chart as shown in Figure 11.



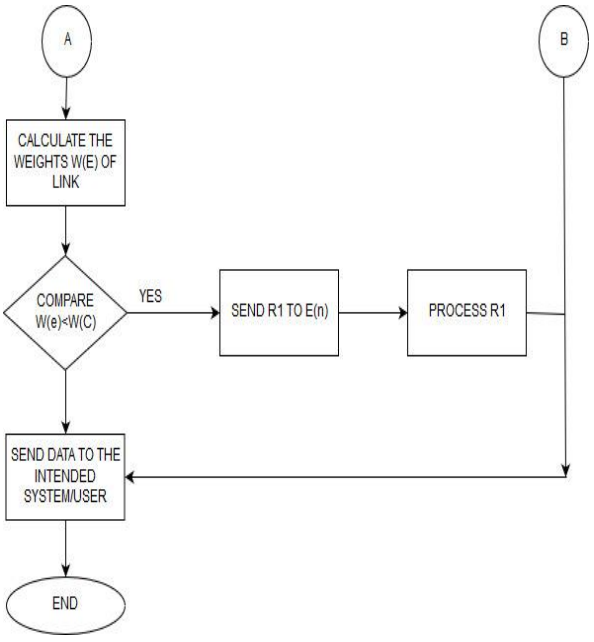


Figure 11. Flowchart of a request lifecycle.

6. RESULTS

The proposed framework is simulated in the MININET simulator using multiple POX controllers and the round-trip time (RTT) of the network for each layer is calculated. During the simulation, different TCP servers are build using Iperf at every layer and a host from one of the cells utilizes the services of these Layers. The configuration of each virtual machine used for simulating the framework is 8GB RAM, i7 processor, and ubuntu 16 Operating system. In the experimental setup, the links to each server are weighted as per the position of the server. The average RTT is shown in Table I and the latencies at different levels are shown in Figure 12.

TABLE I. RTT COMPARISON OF THE PROPOSED FRAMEWORK AT DIFFERENT LEVELS

S. No	Servers	Average Round Trip Time (RTT)
1.	Local Edge Server	8.53ms
2.	Cell (Logical Server)	32.17ms
3.	Neighboring Edge Server	118ms
4.	Cloud	242.82ms

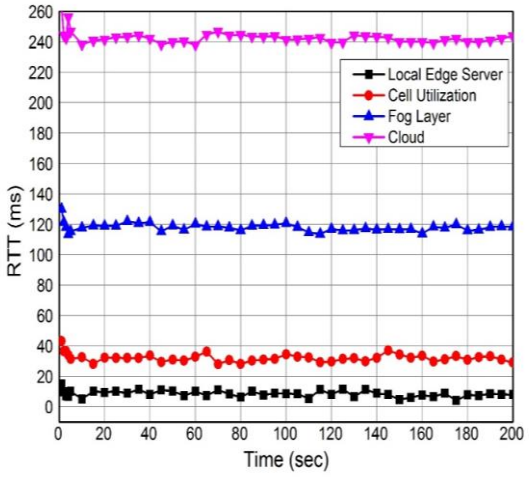


Figure 12. Latencies of different Layers of Architecture.

The use of sync status table in Pox controller shows a 21.67% decrease in the round-trip time, thus reducing the latency of the system when the request is been served by other cells. Figure 13 shows the exchange of messages between the pox controllers and Figure 14 shows the exchange of messages between pox with sync status table. As the modified controllers are state aware of the servers, so it directly sends the request to the cost-effective and available edge server capable of serving the IoT device. which saves time for searching the service. Table II and Figure 15 show the comparison of RTTs between the proposed controller and pox controllers when the requisite services are provided by the neighboring Edge server. Figure 16 shows the comparison between the proposed controller and Pox controller when services are utilized within the cell.

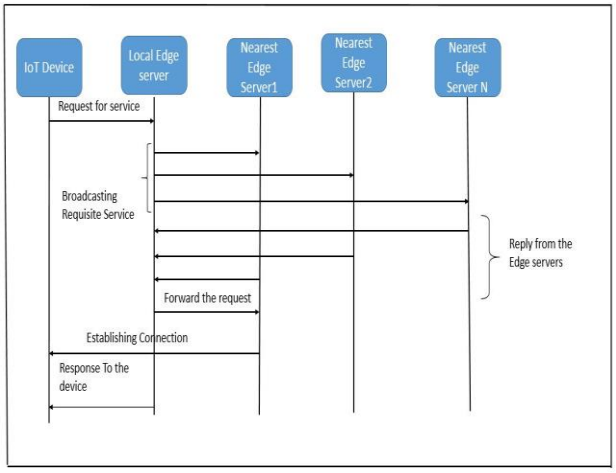


Figure 13. Exchange of messages with pox controllers.

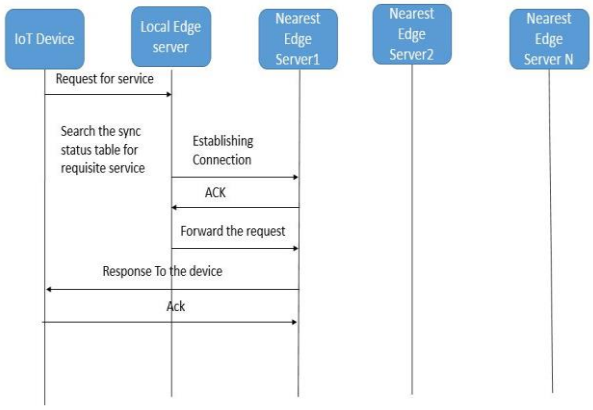


Figure 14. Exchange of message using modified Pox with Sync Status Table.

TABLE II. COMPARISON OF ROUND-TRIP TIME WHILE UTILIZING SERVICES WITH THE HELP OF TWO CONTROLLERS.

S.No	Controller	Avg. RTT (ms)
1	Pox	118
2	Proposed controller	92.41

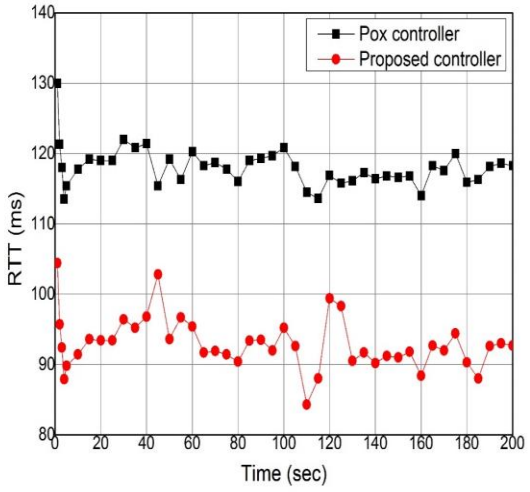


Figure 15. Comparison between the proposed controller and Pox controller.

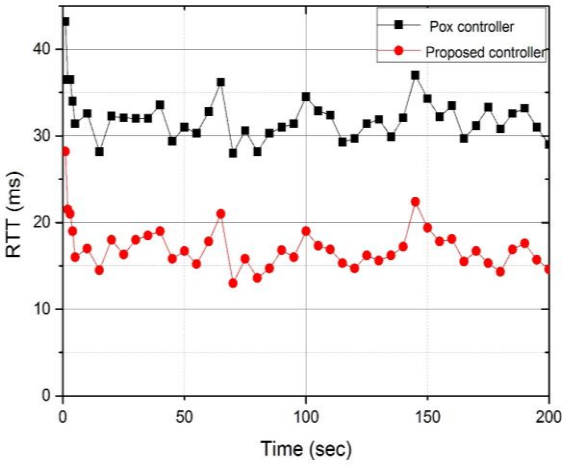


Figure 16. Comparison between the proposed controller and Pox controller when services are utilized within the cell.

Suppose a user makes a TCP request for a simple web page containing 5 assets like images, Script file, and webpage itself. The total time for the web page to load for each layer is calculated as:

1. The establishment of a TCP/IP connection takes 3 instances.
2. The HTTP request from the sender.
3. Response from the server.
4. Response time for five assets.

The resultant time taken to load a simple web page from the server to the client is shown in Table III and illustrated in Figure 17. The results show that there is a remarkable improvement in latency by bringing the services close to the end-user. This approach helps in better utilization of resources and implementing policies in real-time.

TABLE III. TOTAL TIME TO LOAD A SIMPLE WEBPAGE IN DIFFERENT SCENARIOS.

S. No	Servers	Total Time (ms)
1.	Local Edge Server	102.36
2.	Cell (Logical Server)	386.04
3.	Neighboring Edge Server	1416
4.	Cloud	2913

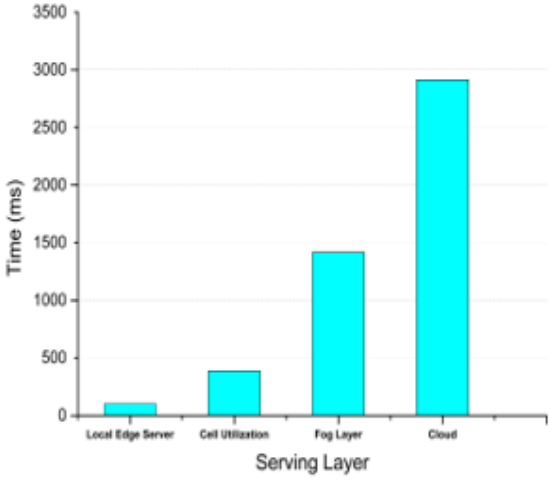


Figure 17. Loading time of webpage from server to client

7. CONCLUSION:

This study proposes an architecture for IoT, which involves the utilization of two technologies viz Software-defined networking and Fog computing. By modifying the architecture of the SDN controller, we can utilize the services within the cell even when Edge servers are unable to provide it. The architecture presented provides a way in which devices can get their requests served at any layer as per the availability of resources and cost incurred. While connecting SDNC's with each other and with the main controller, the architecture becomes distributed as well as centralized. This helps each individual SDNC to have a logically centralized view of the network, which in turn helps in network management. This architecture increases the overall performance of the network and offers the best support for time-constrained applications. Also, this framework increases the availability of the resources in case connectivity to any layer is lost.

REFERENCES

[1] "Gartner: Top 10 Strategic Technology Trends For 2013."0020[Online]. Available: https://www.forbes.com/sites/ericsavitz/2012/10/23/gartner-top-10-strategic-technology-trends-for-2013/#502d3b29b761. [Accessed: 10-Jan-2019].

[2] C. Perera, R. Ranjan, L. Wang, S. U. Khan, and A. Y. Zomaya, "Big Data Privacy in the Internet of Things Era," IT Prof., vol. 17, no. 3, pp. 32–39, May 2015.

[3] R. Khan, S. Ullah Khan, R. Zaheer, and S. Khan, "Future Internet: The Internet of Things Architecture, Possible Applications, and Key Challenges," 2012.

[4] L. Hu, M. Qiu, J. Song, M. S. Hossain, and A. Ghoneim, "Software defined healthcare networks," IEEE Wirel. Commun., vol. 22, no. 6, pp. 67–75, Dec. 2015.

[5] S. M. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, and K. S. Kwak, "The internet of things for health care: A comprehensive survey," IEEE Access, vol. 3, 2015.

[6] H. Arasteh et al., "IoT-based smart cities: A survey," EEEIC 2016 - Int. Conf. Environ. Electr. Eng., pp. 1–6, 2016.

[7] S. Jain, N. V. Kumar, A. Paventhan, V. K. Chinnaiyan, V. Arnachalam, and M. Pradish, "Survey on smart grid technologies-smart metering, IoT and EMS," in 2014 IEEE Students' Conference on Electrical, Electronics and Computer Science, 2014, pp. 1–6.

[8] A. Prasad and P. Chawda, "Power management factors and techniques for IoT design devices," in 2018 19th International Symposium on Quality Electronic Design (ISQED), 2018, pp. 364–369.

[9] W. H. Chin, Z. Fan, and R. Haines, "Emerging technologies and research challenges for 5G wireless networks," IEEE Wirel. Commun., 2014.

[10] P. M. Julia and S. A.F, "Extending-the- Internet-of-Things-to-IPv6-with-Software-Defined-Networking," 2014.

[11] P. Wongthongtham, J. Kaur, V. Potdar, and A. Das, "Big Data Challenges for the Internet of Things (IoT) Paradigm," in Connected Environments for the Internet of Things: Challenges and Solutions, Springer International Publishing, 2017, pp. 41–62.

[12] S. Incorporated ULC, "2015 - Global Internet Phenomena Asia-Pacific & Europe," p. 12, 2015.

[13] S. Yi, C. Li, and Q. Li, "A Survey of Fog Computing: Concepts, Applications and Issues (#16)," Proc. 2015 Work. Mob. Big Data - Mobidata '15, 2015.

[14] F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, "Internet of Things security: A survey," J. Netw. Comput. Appl., vol. 88, pp. 10–28, Jun. 2017.

[15] M. Ammar, G. Russello, and B. Crispo, "Internet of Things: A survey on the security of IoT frameworks," J. Inf. Secur. Appl., 2018.

[16] N. H. B. Halim, N. B. Yaakob, and A. B. A. M. Isa, "Congestion control mechanism for Internet-of-Things (IOT) paradigm," in 2016 3rd International Conference on Electronic Design (ICED), 2016, pp. 337–341.

[17] R. Hassan, A. M. Jubair, K. Azmi, and A. Bakar, "Adaptive congestion control mechanism in CoAP Application Protocol for Internet of Things (IoT)," in 2016 International Conference on Signal Processing and Communication, ICSC 2016, 2016, pp. 121–125.

[18] A. Betzler, C. Gomez, I. Demirkol, and J. Paradells, "CoCoA+: An advanced congestion control mechanism for CoAP," Ad Hoc Networks, vol. 33, pp. 126–139, 2015.

[19] T. Qiu, Y. Lv, F. Xia, N. Chen, J. Wan, and A. Tolba, "ERGID: An efficient routing protocol for emergency response Internet of Things," J. Netw. Comput. Appl., vol. 72, pp. 104–112, 2016.

[20] N. Gozuacik and S. Oktug, "Parent-Aware Routing for IoT Networks," Springer, Cham, 2015, pp. 23–33.

[21] I. Stojmenovic and S. Wen, "The Fog Computing Paradigm: Scenarios and Security Issues," vol. 2, pp. 1–8, 2014.

[22] S. Raza, L. Wallgren, and T. Voigt, "SVELTE: Real-time intrusion detection in the Internet of Things," Ad Hoc Networks, vol. 11, no. 8, pp. 2661–2674, 2013.

[23] H. Perrey, M. Landsmann, O. Ugus, T. C. Schmidt, and M. Wählisch, "TRAIL: Topology Authentication in RPL," Dec. 2013.

[24] P. Pongle, G. C.-I. J. of Computer, and undefined 2015, "Real time intrusion and wormhole attack detection in internet of things," pdfs.semanticscholar.org.

[25] P. Kasinathan, C. Pastrone, M. A. Spirito, and M. Vinkovits, "Denial-of-Service detection in 6LoWPAN based Internet of Things," in 2013 IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2013, pp. 600–607.

[26] W. Kim, "Adaptive Resource Scheduling for Dual Connectivity in Heterogeneous IoT Cellular Networks," Int. J. Distrib. Sens. Networks, vol. 12, no. 4, p. 6036952, Apr. 2016.

[27] M. Surligas, A. Makrogiannakis, and S. Papadakis, "Empowering the IoT Heterogeneous Wireless Networking with Software Defined Radio," in 2015 IEEE 81st Vehicular Technology Conference (VTC Spring), 2015, pp. 1–5.

[28] Le Zhang, "An IOT system for environmental monitoring and protecting with heterogeneous communication networks," in 2011 6th International ICST Conference on Communications and Networking in China (CHINACOM), 2011, pp. 1026–1031.

[29] S. M. A. Oteafy, F. M. Al-Turjman, and H. S. Hassanein, "Pruned Adaptive Routing in the heterogeneous Internet of Things," in 2012 IEEE Global Communications Conference (GLOBECOM), 2012, pp. 214–219.

[30] A. R. Biswas and R. Giaffreda, "IoT and cloud convergence: Opportunities and challenges," in 2014 IEEE World Forum on Internet of Things (WF-IoT), 2014, pp. 375–376.

[31] S. M. Babu, A. J. Lakshmi, and B. T. Rao, "A study on cloud based Internet of Things: CloudIoT," in 2015 Global Conference on Communication Technologies (GCCT), 2015, pp. 60–65.

[32] M. Aazam, I. Khan, A. A. Alsaffar, and E.-N. Huh, "Cloud of Things: Integrating Internet of Things and cloud computing and the issues involved," in Proceedings of 2014 11th International Bhurban Conference on Applied Sciences & Technology (IBCAST) Islamabad, Pakistan, 14th - 18th January, 2014, 2014, pp. 414–419.

[33] S. K. Josyula and D. Gupta, "Internet of things and cloud interoperability application based on Android," in 2016 IEEE International Conference on Advances in Computer Applications (ICACA), 2016, pp. 76–81.

[34] I. Stojmenovic, "Fog computing: A cloud to the ground support for smart things and machine-to-machine networks," in 2014 Australasian Telecommunication Networks and Applications Conference, ATNAC 2014, 2015.

[35] S. Bera, S. Misra, and A. V. Vasilakos, "Software-Defined Networking for Internet of Things: A Survey," IEEE Internet Things J., vol. 4, no. 6, pp. 1994–2008, Dec. 2017.

[36] S. Szott and M. Natkaniec, "Emerging Technologies in Wireless LANs: Theory, Design, and Deployment (Bing, B., Ed.; 2008) [Book review]," IEEE Commun. Mag., vol. 47, no. 4, pp. 18–18, Apr. 2009.

[37] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," Futur. Gener. Comput. Syst., vol. 29, no. 7, pp. 1645–1660, 2013.

[38] S. Tomovic, M. Pejanovic-Djurisic, and I. Radusinovic, "SDN Based Mobile Networks: Concepts and Benefits," Wirel. Pers. Commun., vol. 78, no. 3, pp. 1629–1644, Oct. 2014.

[39] S. Tomovic, N. Prasad, and I. Radusinovic, "SDN control framework for QoS provisioning," in 2014 22nd Telecommunications Forum Telfor (TELFOR), 2014, pp. 111–114.

[40] O. N. F. W. Paper, "Borderline products: new guidance on the classification of food for special medical purposes," 2012.

[41] N. McKeown et al., "OpenFlow," ACM SIGCOMM Comput. Commun. Rev., vol. 38, no. 2, p. 69, Mar. 2008.

[42] Ken Gray and Thomas D.Nadeau, Book -- SDN: Software Defined Networks. O'Reilly Media,Inc, 2013.

[43] H. Kim and N. Feamster, "Improving network management with software defined networking," IEEE Commun. Mag., 2013.

[44] N. Omnes, M. Bouillon, G. Fromentoux, and O. Le Grand, "A programmable and virtualized network & IT infrastructure for the internet of things: How can NFV & SDN help for facing the upcoming challenges," in 2015 18th International Conference on Intelligence in Next Generation Networks, ICIN 2015, 2015.

[45] E. Patouni, A. Merentitis, P. Panagiotopoulos, A. Glentis, and N. Alonistioti, "Network virtualisation trends: Virtually anything is possible by connecting the unconnected," in SDN4FNS 2013 - 2013 Workshop on Software Defined Networks for Future Networks and Services, 2013.

[46] M. Yang, Y. Li, D. Jin, L. Zeng, X. Wu, and A. V. Vasilakos, "Software-Defined and Virtualized Future Mobile and Wireless Networks: A Survey," Mob. Networks Appl., 2015.

[47] Z. Sheng, S. Yang, Y. Yu, A. Vasilakos, J. Mccann, and K. Leung, "A survey on the ietf protocol suite for the internet of things: standards, challenges, and opportunities," IEEE Wirel. Commun., vol. 20, no. 6, pp. 91–98, Dec. 2013.

[48] T. Wood, K. K. Ramakrishnan, J. Hwang, G. Liu, and W. Zhang, "Toward a software-based network: Integrating software defined networking and network function virtualization," IEEE Netw., 2015.

[49] S. M. A. Oteafy and H. S. Hassanein, "Towards a global IoT: Resource re-utilization in WSNs," in 2012 International Conference on Computing, Networking and Communications (ICNC), 2012, pp. 617–622.

[50] C. C. Aggarwal, N. Ashish, and A. Sheth, "The internet of things: A survey from the data-centric perspective," in Managing and Mining Sensor Data, 2014.

[51] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," Comput. Networks, vol. 54, no. 15, pp. 2787–2805, 2010.

[52] M. Dayarathna, Y. Wen, and R. Fan, "Data center energy consumption modeling: A survey," IEEE Commun. Surv. Tutorials, 2016.

[53] Q. Jing, A. V. Vasilakos, J. Wan, J. Lu, and D. Qiu, "Security of the Internet of Things: perspectives and challenges," Wirel. Networks, 2014.

[54] Z. Yan, P. Zhang, and A. V. Vasilakos, "A survey on trust management for Internet of Things," J. Netw. Comput. Appl., 2014.

[55] Á. L. Valdivieso Caraguay, A. Benito Peral, L. I. Barona López, and L. J. García Villalba, "SDN: Evolution and Opportunities in the Development IoT Applications," Int. J. Distrib. Sens. Networks, vol. 10, no. 5, p. 735142, May 2014.

[56] N. A. Jagadeesan and B. Krishnamachari, "Software-Defined Networking Paradigms in Wireless Networks: A Survey," ACM Comput. Surv., vol. 47, no. 2, pp. 1–11, Nov. 2014.

[57] K. Sood, S. Yu, and Y. Xiang, "Software-Defined Wireless Networking Opportunities and Challenges for Internet-of-Things: A Review," IEEE Internet Things J., vol. 3, no. 4, pp. 453–463, Aug. 2016.

[58] Jararweh Y, Al-Ayyoub M, Darabseh A, Benkhelifa E, Vouk M, Rindos A. SDIoT: a software defined based internet of things framework. J Ambient Intell Humaniz Comput. 2015;6(4):453-461.

[59] Sahoo KS, Sahoo B, Panda A. A secure SDN framework for IoT. Paper presented at: 2015 International Conference on Man and Machine Interfacing (MAMI); 2015; Bhubaneswar, India.

[60] Flauzac O, Gonzalez C, Nolot F. New Security Architecture for IoT Network. Procedia Comput Sci. 2015;52:1028-1033.

[61] Chakrabarty S, EngelsDW.Asecure IoT architecture for smart cities.Paper presented at: 13th IEEE AnnualConsumer Communications andNetworking Conference (CCNC); 2016; Las Vegas, NV.

[62] Balfour RE. Building the Internet of Everything (IoE) for first responders. In: Systems, Applications and Technology Conference (LISAT); 2015; IEEE Long Island.

[63] Tajiki MM, Akbari B, Shojafar M, et al. CECT: computationally efficient congestion-avoidance and traffic engineering in software-defined cloud datacenters. 2018. arXiv preprint arXiv:1802.07840

[64] Ukil A, Sen J,Koilakonda S. Embedded security for Internet of Things. Paper presented at: 2nd National Conference on Emerging Trends and Applications in Computer Science; 2011; Shillong, India.

[65] M. Conti, P. Kaliyar, and C. Lal, "CENSOR : Cloud-enabled secure IoT architecture over SDN paradigm," no. August, pp. 1–14, 2018.

Hushmat Amin Kar (corresponding author) has received his B.Tech. degree in Information Technology from the University of Jammu, J&K, India, in 2012, M. Tech. Degree from National Institute of Technology Srinagar, India in 2014. He has worked as Assistant Professor in the Department of Information Technology, National Institute of Technology, Srinagar, India before joining as Senior Research Fellow in the Department of Electronics and Communication Engineering in the same Institute. He is a member of IEEE, IETE, IEI, and IIETA. His research interests include Wireless Sensor Networks, Internet of Things, Big Data, and cloud computing.

Ghulam Mohammad Rather was born in Kashmir, India. He received his B.E. degree in from the Kashmir University, India, in 1981, the M. S. degree (1988) and the Ph.D. degree (1997) from the Indian Institute of Sciences (IISc) Bangalore, India. He is currently Professor in the Department of Electronics and Communication Engineering, National Institute of Technology, Srinagar, India. He has more than 35 years of teaching experience. He is a senior member of IEEE and IETE. His research interests include Communication Systems, Computer Networks, Antennas and Propagation.