Transmission Control Protocol (TCP) Incast and Outcast Issues in Cloud Computing

Ban Jawad Khadhim[1]*, Shaymaa Taha Ahmed[2], Rasha Mahdi Abdulqader[3]

[1]*Department of Computer Science, College of Basic Education,University of Diyala, Diyala, Iraq
[2]Department of Computer Science, College of Basic Education,University of Diyala, Diyala, Iraq
[3]Department of Computer Science, College of Basic Education,University of Diyala, Diyala, Iraq

Abstract--- Nowadays, cloud computing has rapid development over the Internet, especially in areas that involve high demands on a large scale of network computing such as IT. Sharing of resources is the main aim of cloud computing where processing service and network bandwidth needed to access it should be allocated concurrently. A general overview of the cloud computing environment and its significance is presented. The paper discussion on the shortcomings of the fundamental design and performance issues of Transmission Control Protocol (TCP) Incast, Outcast in cloud networks.

1. Overview of Cloud Computing Environment

 In the rapid development of the Internet and the rise of customer service registering and have prompted distributed computing condition which turns into an essential piece of the future Internet. With respect to the execution of the cloud server farm systems, there are three basic prerequisites high burst movement resilience, low deferral, and high throughput (Kadhim, Yusof and Mahdi, 2018). These execution prerequisites are forced by the previously mentioned movement composes that exist together in the cloud computing systems where each activity compose some of the nature of the management application, which contrasts from the other requires (S, R. Patil and C, 2018). The core issue that has been identified with the implementation of the organization is to use the cloud of Transmission Control Convention/Web Convention (TCP/IP) stack that was particularly intended for the Internet. TCP/IP experiences a few execution issues when it is customarily and truly utilized in cloud server farm systems without adjustment as indicated by the assorted necessities of a few exist together activity sorts of the cloud condition (Barona and Anita, 2017). As these cloud data centre networks deal with and process a huge amount of data at any given time, it is important to have an efficient and reliable cloud networking environment, considering performance, scalability, resilience and sustainability. Furthermore, since the demand for cloud computing is increasing, it is essential to meet the requirements of the scaling cloud users by designing fast, reliable, and efficient. Networks for communication within cloud data centres to manage the increasing load of cloud users' traffic (Winkler, 2011). While the reliability of the cloud networks relies on their transport layer, it is worthy to understand the performance related issues in Cloud Computing. In regards to the network topology, the two-tier network architecture is common in cloud nowadays, where access switches/routers depend on switching and routing functionalities provided by aggregation switches/routers. This architecture has many advantages; the design becomes simpler as only a few switches/routers can be sufficient, and thus, the network latency and aggregate power consumption can be reduced (Kanaan, Mahdi and Ail, 2018). Nevertheless, this architecture presents limited scalability. Once the input and ports on a particular aggregation switch/router are fully used, then adding another aggregation switch/router would present a high complexity (Borcs, Nagy and Benedek, 2017). A mesh connection between a pair of aggregation switches/routers should be established with high bandwidth; thus, avoiding having bottlenecks in the cloud network.

*Corresponding Author: Ban Jawad Khadhim, Email id: *Fulastar12@gmail.com*

But, these aggregation switches/routers perform routing protocols, this means that the more switches/routers connected in the network the more routing interfaces and information are provided; hence mesh design would increase the complexity inside the cloud networks. Figure 1 network cloud with tiered addresses.
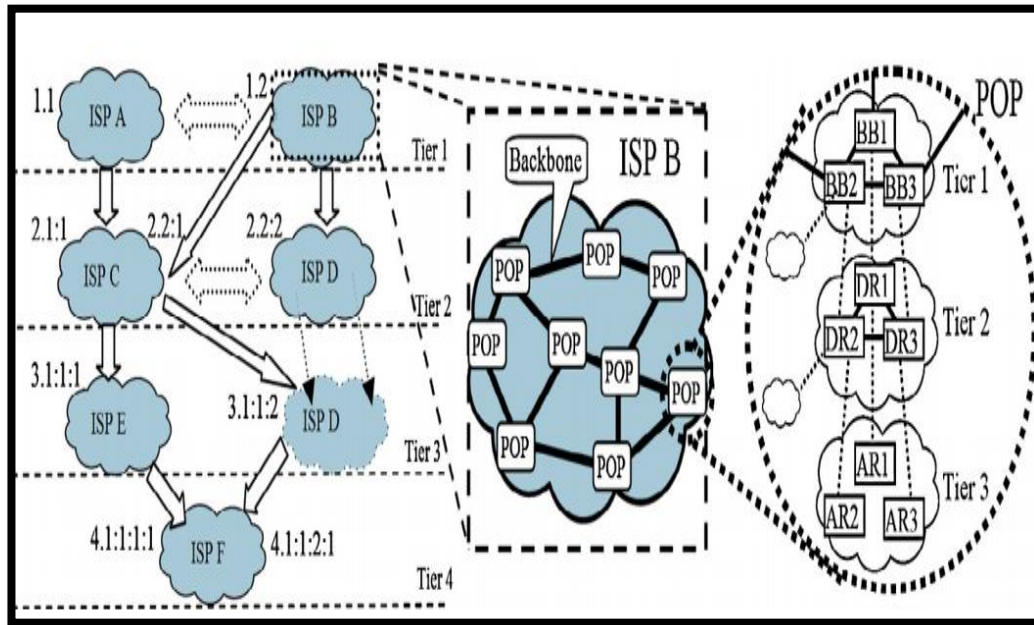
Fig. 1: Network cloud with tiered addresses (Nozaki *et al.*, 2014)

2. TCP Issues in Cloud Computing

Transmission Control Protocol (TCP) is the most common transport layer that has been used widely by Internet Protocol applications (Elezi and Raufi, 2015) (Wang and Matta, 2018). And it dominates the majority of traffic in the cloud data centre networks (Mahdi, Qusay Kanaan and Khudhair, 2018).Due to its congestion control mechanisms, TCP has recognized to be scalable, robust, and capable of dealing with different network conditions. Nonetheless, it has been realized that TCP fails to fulfil the previously mentioned three basic necessities on account of the attributes an assorted variety of the cloud server farm systems and the assortment of prerequisites of their applications. Such assorted variety and prerequisites forced numerous difficulties to TCP blockage control instruments and make TCP experiences a few lacks (Saleh, Qadir and Ilyas, 2018). Such as TCP in the cast, TCP outcast, queue accumulation, buffer pressure, and the effect of false congestion (Kadhim, Yusof and Selamat, 2018).These issues are discussed in the following subsections, highlighting their causes and some proposed solutions to alleviate.

3. TCP Incast

TCP Incast, also known as TCP Throughput Collapse, is widely existed in the cloud computing environment (Ren *et al.*, 2016),(Qin *et al.*, 2016),(Hwang, Yoo and Choi, 2014). It is defined as the uncontrolled TCP behavior that causes considerable link underutilization in many-to-one communication paradigms (such as partition/aggregate application model illustrated in Figure .2) that are the reason for a few vast scale applications like web look, informal organization, and commercial (Lu, Fan and Qian, 2018),(Ghazisaeedi and Huang, 2017). To name a few.

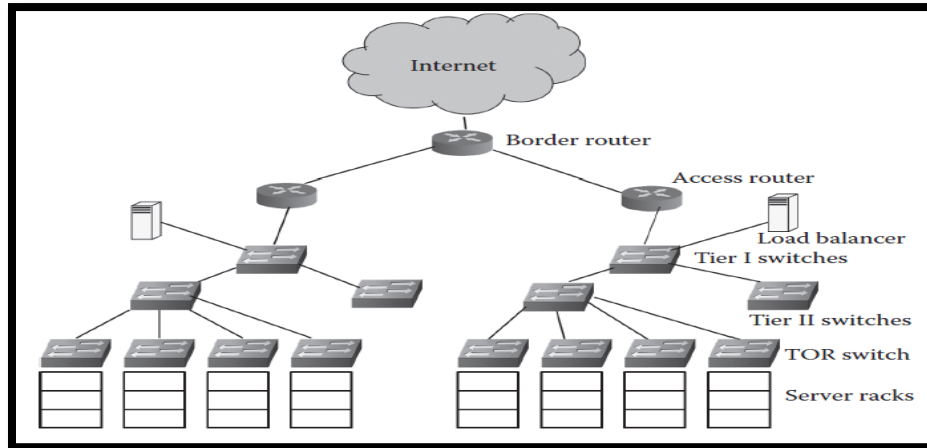*Corresponding Author: Ban Jawad Khadhim, Email id: Fulastar12@gmail.com

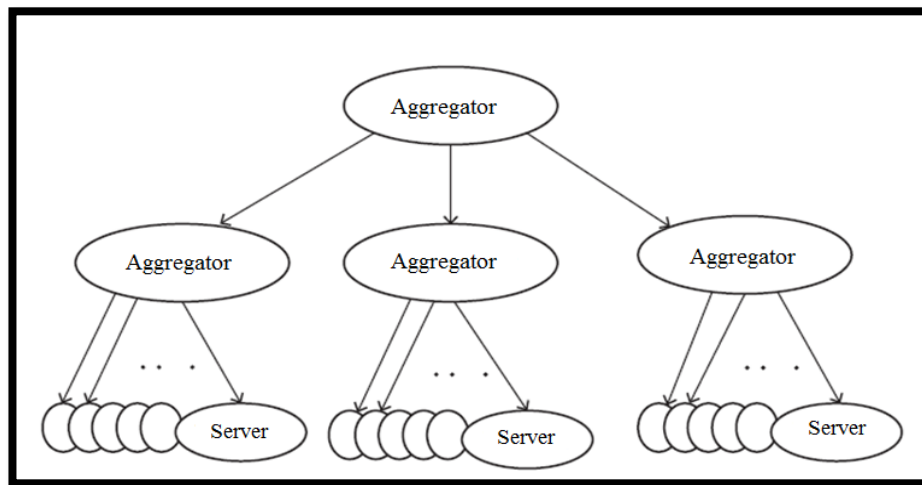Fig. 2: Partition/aggregate application model (Ren *et al.*, 2016)

Fig. 3: TCP Incast in Cloud Computing Environment

In such applications, an aggregator (requester) sends solicitations to a few servers (additionally called labourers) that host the required information (see Figure .3). Once the demand is gotten, the servers all the while reaction with immense information that goes through a bottleneck connect to the comparing aggregator. The concurrent answer to the solicitations is because of the tightly constrained reaction time required by the earth. In this manner, it is exceedingly likely that came about information parcels would flood the lines of top-of-the-rack (ToR) switches, which subsequently causes bundle misfortune. Such circumstance is called the effect of synchronized mice (Teli, Thomas and Chandrasekaran, 2016), in addition, where there are obstacles to the simultaneous transmission of (J. Wang *et al.*, 2015). In which the servers are aborted from transmitting following information square however sitting tight to all server to send their current information. Thus, with such circumstance, the quantity adoption of parallel servers, moving output is seen by the compound mainly because of the ordinary castings expulsion (Dong *et al.*, 2014). The retransmission of missing packets is activated only once the retransmit timeout (RTO) is terminated. In this manner, as brief associations are deferred delicate require short and narrow reaction time, the execution is corrupted impressively by the continuous timeouts caused TCP incast (J. Wang *et al.*, 2015). Several solutions(Zhang *et al.*, 2016). Were proposed to mitigate TCP incast problem, where some are focusing on revising TCP, while others try to find resolutions from other TCP/IP layers.

*Corresponding Author: Ban Jawad Khadhim, Email id: Fulastar12@gmail.com

4. TCP Outcast

TCP Outcast phenomenon is that when traffic that consists of connections of many incoming flows and few flows at two ports switch (or router) and compete to one common output port, throughput the few flows is affected implicitly (Qin *et al.*, 2016). It occurs in the cloud environment where routers and switches of the data centers operate Tail-Drop (TD) queue management mechanism as it has less implementation complexity and less computational overhead compared to other queue management mechanisms(Anelli, Diana and Lochin, 2014). However, TD does not guarantee the proportional packet drop ratio of each flow to the flow's share of the bandwidth. TD drops packets from the end of the queue when the packet arrival reaches the maximum queue length. TD leads to frequent and multiple successive packet drops from the queue of a specific port, which contributes, to frequent timeouts and severe global synchronization occasions. This case is known as port blackout, which causes a noteworthy reduction in the performance of few flows (in terms of response times) due to high delay, resulting from frequent timeouts.   Figure .4 illustrates the port blackout situation, where data packets are arriving at input ports (A and B) and competing for one output port (C). As shown in the figure, data packets that have arrived at port B are buffered successfully, while the ones arrived at the port and are being dropped successively.
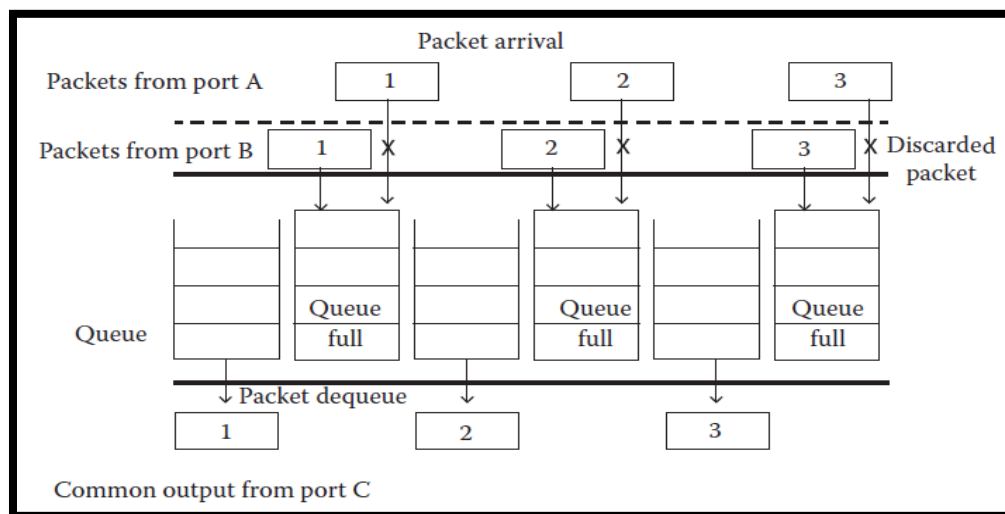

Fig. 4: Port blackout case (Prakash *et al.*, 2012).

As throughput of a TCP connection is symmetrical to its RTT in an inverse manner, could lead to RTT bias. This means that TCP connections that have low RTT can attain a higher share of bandwidth than other connections. Nonetheless, it has been perceived low RTT TCP connections are overtaken by high RTT TCP connections as TCP outcast drives TCP to present inverse RTT bias (Prakash *et al.*, 2012). Since many-to-one communication model is the groundwork of several applications in the cloud, and that networks of the data centres utilize TD queue management mechanism in their routers and switches, TCP Outcast is also the common exist problem the cloud computing environment.TCP outcast problem can be mitigated by either (i) by using other queue management mechanisms such as Random Early Detection (RED) (Misra *et al.*, 2010) (and its variants like Stabilized RED (SRED) Adaptive RED , Double Slop RED (DSRED) (Alkharasani *et al.*, 2017), Hyperbola RED (HRED) (Zhou, Yeung and Li, 2006)) or Early Random-Drop (ERD) (Lin and Morris, 1997), BLUE , Random Early Marking (REM) (Xu, Tian and Ansari, 2004) and Stochastic Fair Queue (SFQ) (Joy and Nayak, 2015) to name a few; or (ii) by designing effective congestion control mechanism at TCP end-hosts that could minimize the queue occupancy level on routers and switches networks of data centres.

*Corresponding Author: Ban Jawad Khadhim, Email id: Fulastar12@gmail.com

5. Queue Build-up

As specified before, cloud applications produce distinctive compose and size of movement; in this manner, fleeting activity, medium-lived movement, and extensive activity regularly coincide in server farms of the distributed computing condition. The idea of seemingly perpetual activity propels the system to floods the lines and blockage fall. Henceforth, the execution of the brief activity is impressively influenced when this movement goes through a similar connection alongside the enduring movement (Kadhim *et al.*, 2017).The execution of brief activity because of that seemingly perpetual movement involves much space of the line. In this way, it exceptionally likely more parcels the transient activity will be dropped (Belamfedel Alaoui, Tissir and Chaibi, 2018). This like the impact of TCP incast issue the execution brief movement is influenced by timeouts come about by visit bundle misfortune (Zhang and Ansari, 2011). Additionally, regardless of whether there were no bundle loses (common case), the late lined parcels of brief activity would encounter higher lining delay, sitting tight for the bundles of the seemingly perpetual movement to be prepared first. This marvel is known as line develop, which can be alleviated by just keeping up little line length in the keys and keys cloud systems at means of proactively diminishing TCP senders sending rate. Bundles drops can be utilized as the flag of clog warning signs to illuminate TCP senders about blockage (Xu *et al.*, 2017). The TCP sender would just understand the nearness clog and lessens its transmission rate if just gets three copy affirmations (ACKs) or after the timeouts; subsequently, the TCP sender would neglect to neglect to decrease the line inhabitancy to a sensible level. In this manner, to conquer the line develop an issue, a proactive strategy is required to limit the line length.

6. Buffer Pressure

The greedy nature of long-lived traffic also causes another deficiency called buffer pressure. As mentioned previously, the coexistence of both short-lived traffic and long-lived traffic on the same link results in those packets of long-lived traffic of occupying more space of the queue; leaving small spaces to house the bursty short-lived traffic produced by many-to-one communication form (Tran *et al.*, 2016). Therefore, short-lived connections would suffer from high packet loss. Furthermore, as the large part of data traffic in a cloud computing environment is naturally busty, packets of short-lived traffic would face frequent drops due to the long-last long-lived traffic that engage the queue space for long time (B. Wang *et al.*, 2015). Mitigating this problem is similar to that of queue build-up problem, which may be alleviated reducing queue occupancy level in routers and switches the cloud networks.

7. Pseudo-congestion Effect

Congestion happens when the network load is greater than its capacity. Congestion results in large delays, frequent buffer overflows, and high packet loss. Contemporary data centres in the cloud environment are called virtualized data centres as they use virtual machines (VMs) to provide access for on-demand cloud computing services (Ahmed, Science and Education, 2018) . Although virtualization technology has many advantages such as efficient utilization of the servers and low cost of the system maintenance, it worsens the performance of TCP and UDP dramatically. It was observed that virtualization causes throughput fluctuation and high end-to-end even in conditions where the network is not overloaded (Zhang, Ren and Lin, 2013). The larger the number of running VMs is high standby time for VM to access the wizard from the physical machine (Flores, Wenzel and Kuzmanovic, 2016). Consequently, unexpected network delays take place which affects throughput constancy notably. Additionally, with increasing delay, VM sender may suppose that there is congestion occurs in the network severely and significantly reduces its transmission rate as a reaction. This situation is known as pseudo-congestion effect as the VM sender perceives the congestion detected falsely (Satapathy *et al.*, 2017).Two possible solutions to overcome pseudo-congestion effect problem; (i) by developing an efficient scheduler for virtual machine monitor (hypervisor) in order to reduce the scheduling delay, or (ii) by modifying TCP to detect and react to the pseudo-congestion intelligently.

*Corresponding Author: Ban Jawad Khadhim, Email id: Fulastar12@gmail.com

8. Conclusion

The TCP could not endure very burst movement; it has neglected to offer low defer and high throughput for the cloud applications because of the assorted variety of the attributes and necessities of such applications. The conventional TCP clog control instruments make TCP experiences a few execution inadequacies, for example, TCP incast, TCP outsider, line develop, cushion weight, and pseudo-blockage impact in cloud server farm networks. The reasoned that neither TCP variations nor the present line administration and blockage control systems gives a persuading answer for the previously mentioned execution issues of TCP. In this manner, there is a high need for additional research to enhance the execution of cloud server farm systems, which centres on building up a dynamic line administration and clog control component that communicates with an effective variant of ECN-able TCP.

9. References

[1] Q. K. Kadhim, R. Yusof, and H. S. Mahdi, "A Review Study on Cloud Computing Issues A Review Study on Cloud Computing Issues," in *Journal of Physics*, 2018, pp. 1–10.

[2] S. S, R. R. Patil, and P. C, "'Cloud computing an overview,'" *Int. J. Eng. Technol.*, vol. 7, no. 4, pp. 2743–2746, 2018.

[3] R. Barona and E. A. M. Anita, "A survey on data breach challenges in cloud computing security: Issues and threats," in *2017 International Conference on Circuit ,Power and Computing Technologies (ICCPCT)*, 2017.

[4] V. (J. R.) Winkler, "Chapter 2 - Cloud Computing Architecture," in *Securing the Cloud*, 2011.

[5] Q. Kanaan, H. S. Mahdi, and H. K. Ail, "Storage Architecture for Network Security in Cloud Computing," *Diyala J. Pure Sci.*, vol. 14, no. 1, pp. 1–17, 2018.

[6] A. Borcs, B. Nagy, and C. Benedek, "Instant Object Detection in Lidar Point Clouds," *IEEE Geosci. Remote Sens. Lett.*, 2017.

[7] Y. Nozaki, P. Bakshi, H. Tuncer, and N. Shenoy, "Evaluation of tiered routing protocol in floating cloud tiered internet architecture," *Comput. Networks*, 2014.

[8] M. Elezi and B. Raufi, "Conception of Virtual Private Networks Using IPsec Suite of Protocols, Comparative Analysis of Distributed Database Queries Using Different IPsec Modes of Encryption," *Procedia - Soc. Behav. Sci.*, 2015.

[9] Y. Wang and I. Matta, "Multi-layer Virtual Transport Network management," *Comput. Commun.*, vol. 130, no. March, pp. 38–49, 2018.

[10] H. S. Mahdi, Qusay Kanaan and, and I. Y. Khudhair, "Empirical Investigation of TCP Incast Congestion in Wireless Cloud Computing Networks," *J. Comput. Sci.*, vol. 14, no. 5, pp. 663–672, 2018.

[11] S. Saleh, J. Qadir, and M. U. Ilyas, "Shedding Light on the Dark Corners of the Internet: A Survey of Tor Research," *J. Netw. Comput. Appl.*, vol. 114, no. June 2017, pp. 1–28, 2018.

[12] Q. K. Kadhim, R. Yusof, and S. R. Selamat, "The Cloud Computing Control in the Government Services," *Jour Adv Res. Dyn. Control Syst.*, vol. 10, no. 04, pp. 1136–1147, 2018.

[13] Y. Ren, J. Li, S. Shi, L. Li, G. Wang, and B. Zhang, "Congestion control in named data networking - A survey," *Computer Communications*, vol. 86. pp. 1–11, 2016.

[14] Y. Qin, W. Yang, Y. Ye, and Y. Shi, "Analysis for TCP in data center networks: Outcast and Incast," *Journal of Network and Computer Applications*, vol. 68. pp. 140–150, 2016.

[15] J. Hwang, J. Yoo, and N. Choi, "Deadline and Incast Aware TCP for cloud data center networks," *Computer Networks*, vol. 68. pp. 20–34, 2014.

[16] Y. Lu, X. Fan, and L. Qian, "Dynamic ECN marking threshold algorithm for TCP congestion control in data center networks," *Comput. Commun.*, vol. 129, no. 200, pp. 197–208, 2018.

[17] E. Ghazisaeedi and C. Huang, "GreenMap: Green mapping of MapReduce-based virtual networks onto a data center network and managing incast queueing delay," *Comput. Networks*, vol. 112, pp. 345–359, 2017.

[18]   P. Teli, M. V. Thomas, and K. Chandrasekaran, "Big Data Migration between Data Centers in Online Cloud Environment," *Procedia Technol.*, vol. 24, pp. 1558–1565, 2016.

[19]   J. Wang, J. Wen, C. Li, Z. Xiong, and Y. Han, "DC-Vegas: A delay-based TCP congestion control algorithm for datacenter applications," *J. Netw. Comput. Appl.*, vol. 53, pp. 103–114, 2015.

[20]   J. K. Dong, H. B. Wang, Y. Y. Li, and S. D. Cheng, "Virtual machine placement optimizing to improve network performance in cloud data centers," *J. China Univ. Posts Telecommun.*, 2014.

[21]   J. Zhang, D. Zhang, K. Huang, and Z. Qin, "Minimizing datacenter flow completion times with server-based flow scheduling," *Computer Networks*, vol. 94. pp. 360–374, 2016.

[22]   P. Anelli, R. Diana, and E. Lochin, "FavorQueue: A parameterless active queue management to improve TCP traffic performance," *Comput. Networks*, vol. 60, pp. 171–186, 2014.

[23]   P. Prakash, A. Dixit, Y. C. Hu, and R. Kompella, "The TCP Outcast Problem : Exposing Unfairness in Data Center Networks," *NSDI'12 Proc. 9th USENIX Conf. Networked Syst. Des. Implement.*, 2012.

[24]   S. Misra, B. J. Oommen, S. Yanamandra, and M. S. Obaidat, "Random Early Detection for congestion avoidance in wired networks: A discretized pursuit learning-automata-like solution," *IEEE Trans. Syst. Man, Cybern. Part B Cybern.*, 2010.

[25]   A. M. Alkharasani, M. Othman, A. Abdullah, and K. Y. Lun, "An Improved Quality-of-Service Performance Using RED's Active Queue Management Flow Control in Classifying Networks," *IEEE Access*, 2017.

[26]   K. Zhou, K. L. Yeung, and V. O. K. Li, "Nonlinear RED: A simple yet efficient active queue management scheme," *Comput. Networks*, vol. 50, no. 18, pp. 3784–3794, 2006.

[27]   D. Lin and R. Morris, "Dynamics of Random Early Detection," *ACM SIGCOMM Comput. Commun. Rev.*, 1997.

[28]   K. Xu, Y. Tian, and N. Ansari, "TCP-Jersey for wireless IP communications," *IEEE J. Sel. Areas Commun.*, 2004.

[29]   S. Joy and A. Nayak, "Improving flow completion time for short flows in datacenter networks," in *Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management, IM 2015*, 2015, pp. 700–705.

[30]   Q. K. Kadhim, R. Yusof, H. S. Mahdi, and S. R. Selamat, "The Effectiveness of Random Early Detection in Data Center Transmission Control Protocol - based Cloud Computing Networks," *Int. J. Commun. Antenna Propag.*, vol. 7, no. October, pp. 1–7, 2017.

[31]   S. Belamfedel Alaoui, E. H. Tissir, and N. Chaibi, "Active queue management based feedback control for TCP with successive delays in single and multiple bottleneck topology," *Comput. Commun.*, vol. 117, no. January, pp. 58–70, 2018.

[32]   Y. Zhang and N. Ansari, "On mitigating TCP Incast in Data Center Networks," in *Proceedings - IEEE INFOCOM*, 2011.

[33]   L. Xu, K. Xu, Y. Jiang, F. Ren, and H. Wang, "Throughput optimization of TCP incast congestion control in large-scale datacenter networks," *Comput. Networks*, vol. 124, pp. 46–60, 2017.

[34]   V. H. Tran, Q. De Coninck, B. Hesmans, R. Sadre, and O. Bonaventure, "Observing real Multipath TCP traffic," *Comput. Commun.*, 2016.

[35]   B. Wang, Z. Qi, R. Ma, H. Guan, and A. V. Vasilakos, "A survey on data center networking for cloud computing," *Computer Networks*. 2015.

[36]   S. T. Ahmed, C. Science, and B. Education, "Innovative Queue Management Mechanism for Congestion Control in Wireless Cloud Computing Environment," *Jour Adv Res. Dyn. Control Syst.*, vol. 10, no. May, pp. 1208–1214, 2018.

[37]   J. Zhang, F. Ren, and C. Lin, "Survey on transport control in data center networks," *Network, IEEE*, 2013.

[38]   M. Flores, A. Wenzel, and A. Kuzmanovic, "Enabling router-assisted congestion control on the Internet," in *Proceedings - International Conference on Network Protocols, ICNP*, 2016.

[39]   S. C. Satapathy, V. Bhateja, A. Joshi, S. Kumari, D. C. S. Lamba, and A. Kumar, *Performance Analysis of Adaptive Approach for Congestion Control In Wireless Sensor Networks*. 2017.

*Corresponding Author: Ban Jawad Khadhim, Email id: Fulastar12@gmail.com