Journal of Theoretical and Applied Information Technology
15th December 2011. Vol. 34 No.1
© 2005 - 2011 JATIT & LLS. All rights reserved.
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ARCHITECTURE AND FUNCTIONAL STRUCTURE OF TRANSMISSION CONTROL PROTOCOL OVER VARIOUS NETWORKS APPLICATIONS

GHASSAN A. ABED, MAHAMOD ISMAIL, KASMIRAN JUMARI
Faculty of Engineering and Built Environment
Universiti Kebangsaan Malaysia, 43600 UKM, Bangi, Selangor, Malaysia
E-mail: ghassan@eng.ukm.my, mahamod@eng.ukm.my, kbj@eng.ukm.my

ABSTRACT

The communication functions and applications divided into layers, and each device or terminal has a assigned 'stack' in these layers, where that permit each layer to speak and address the corresponding stack in other layers. That is one of the benefits of layering: the ability to change a layer without requiring other layers to change as well, as long as the service and semantics provided by the layer remain intact. The fourth layer in layered model is transport layer where the protocols can provide a tight connection between any two hosts or more than two. Also it provides explicit of data transferring between end terminals or users, and that will help the upper layers from any involvement to provide reliable and cost-effective in transferring data in links, even it will control the reliability of a given links. Transmission Control Protocol (TCP) is a basic communication language, and a connection oriented protocol tied with transport layer consists of collection of rules and procedures to control communication between links. TCP offers important features of flow control, reliability, congestion control, and connection management. This article provide full-overview to the network layering based on Open Systems Interconnection (OSI) model and explains the architecture and the functional structure of TCP over transport layer and analyzes the factors and parameters that effect on TCP behavior and performance.

Keywords: *TCP, OSI model, Network Layers, RTT.*

1. INTRODUCTION

Layering in communications depends on each layer deals with the other two layers, below and above it. From previous concept, the layering approach is a method to re-architect the communication system vertically. The services, or functions, provided by any given layer depend solely on the layer immediately below it. In addition, each layer of the stack has a peer interface with the same layer running on a different node in the network [1]. In the 1980's the International Standards Organization (ISO) began developing a model for a network system called Open Systems Interconnection (OSI) model. This model had seven layers and is mostly used as a reference in the networking world [2]. The model never appealed to a wide audience and instead TCP/IP became the widely accepted and deployed model. In illustrative purposes only, the standard layered model used is the OSI Model 1, where this model includes of seven layers as shown in figure 1, in decreasing from seventh layer (Application Layer) to the first layer (Physical Layer) [3]:

- Application Layer. In OSI model, the Application Layer defined with limited and small role where this layer provides end-user services, like e-mail and other Internet applications.
- Presentation Layer responsible on data managements, data delivery, and data formatting to the upper layer (Application Layer) for more proceedings such as data conversation or data compressing and de-compressing.
- Session Layer. In Session Layer, the managing between the application processes of the end-use done here and the sessions of communication which depend on request and response between applications occur in this layer too such as Authentication/Authorization.

Journal of Theoretical and Applied Information Technology
15th December 2011. Vol. 34 No.1
© 2005 - 2011 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

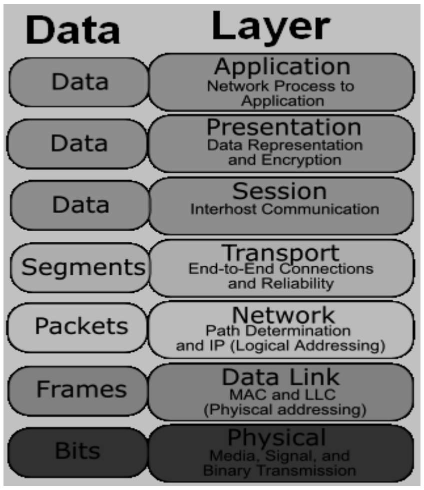

Figure 1 OSI model

- Transport Layer guarantee the data transfer over end-to-end with other convenient services like flow rate control, reliability, and connection oriented of the data stream.
- Network Layer provides different length data sequences transferring from source to a destination over one or many networks in same time with maintain the QoS task. Also it responsible on packets routing and the routing within intermediate routers.
- Data Link Layer responsible on transfer data between networks entities and provide transmit/receive packets and resolve hardware addresses and also to detect the error correction as possible that may be happen in physical layer.
- Physical Layer represents the most complex layer in the OSI model which it include the main material of the hardware technologies of data transmission in network. An example for the physical layer tools are physical cables, air, and the transmission medium.

The network layering depend on the concept of that no data transferred directly between layers at the same level but the data and the control are flow from one layer to the layer below it until it reaches physical layer, where all transmissions only at the physical layer. The physical payer is responsible for actually sending the data over the communication link connecting the node to the next one en route to the eventual destination, while the network layer provides unreliable packets transferring service between any two hosts. The OSI model, more commonly known as simply OSI, is another model that can help break the TCP/IP suite into modules as shown in figure 2. Technically speaking, it is exactly the same as the TCP/IP (Transmission Control Protocol/Internet Protocol) model, except that it has more layers.

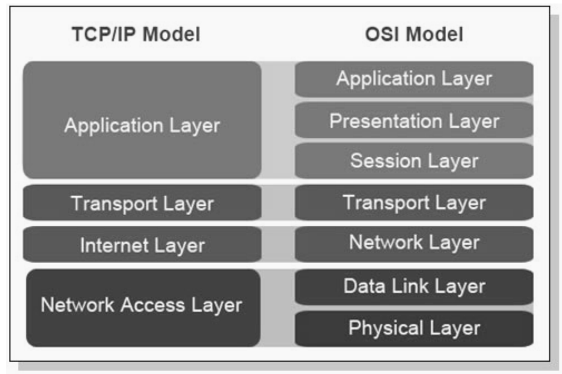

Figure 2 OSI model versus TCP/IP model

This is currently being pushed by Cisco since it aids in learning the TCP/IP stack in an easier manner. Instead of four layers, the OSI model has seven and we can see a direct comparison of the two models in figure 2. Where that only the Application Layer and Network Layer are divided into smaller layers and the Internet Layer is renamed to the Network Layer. The Transport Layer responsible to provide transmission services between any two applications in the network. These applications include email (SMTP: Simple Mail Transfer Protocol), remote login (TELNET: Terminal emulation program for TCP/IP networks), file transfer (FTP: File Transfer Protocol), web browsers (HTTP: Hypertext Transfer Protocol), remote file systems (NFS: Network File System), name-to-address translation (DNS: Domain Naming System), voice and video streaming, etc. [4]. Some protocols are connection oriented, this means that the transport layer can preserve the packets track and retransmit the other packets which that fail to reach. The interface between the application and the transport layer is provided via the "sockets" interface, which provides primitives to open, write, read, and perform other functions, making the network look like a local file with special semantics to the application. The best known example of a layer four protocol is TCP. In principle, the application can change from TCP to some other protocol rather easily, particularly if the other protocol provides semantics similar to TCP (reliable, in-order delivery).

Journal of Theoretical and Applied Information Technology
15th December 2011. Vol. 34 No.1
© 2005 - 2011 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2. TRANSMISSION CONTROL PROTOCOL (TCP)

There are many TCP variants that modified and developed with respectively with the communications needs. Most of TCP current versions are include set of algorithms which built to control the congestion in critical links of network with maintaining the network throughput [5]. In present years, TCP has been faced the fast growth in internet in parallel with the demand increasing to transfer the media on high speed links supported TCP. In order to improve its performance TCP cuts down the size of its congestion window resulted in further performance degradation. This is a more serious problem in bursty and highly mobile networks which have rapid topological changes [6].

TCP provides division for sequenced data stream into packets, confirms the packets delivery with the possibility of losing the IP layer loses, retransmit, reorders, or packets duplication, and monitoring the network band capacity to avoiding congestions. TCP protocol can provide over two end points connection, flow rate controlling with bidirectional link and data reliability [7].In addition, each TCP sender can regulates the size of the congestion window using the congestion control mechanism and the TCP can update and dynamically regulate the window size depending on the packets ACK or by indicates the packets losses when occur. If the congestion window has constant value, the ACK timing of the sent packets will depend on the ACK of the first set of packets (early packets). TCP sliding window depend on ACK clock which calculate the sender flow rate and when RTT changed with different values, the sliding window will determine the mean sending rate of complete window per average RTT.
The transmission window size controlled by dependence on the ACKs received each RTT and these parameters indicate the general differences between TCP versions. The main function of TCP window control is to obtain high packets rate with minimum losses by avoiding network overloading in the same time to provide optimum sharing to the network bandwidth among connections. The optimum bandwidth sharing can changed because the varying amounts of overcrowding between traffics over the network, also it because the varying in network itself like the updates in routing or the time-varying capacity over radio links [7].
TCP Tahoe and TCP Reno are mostly applied over many wireless applications because of the effective congestion control mechanisms.
These mechanisms provide varying in size of congestion window depending on ACK status, thus when packets acknowledged the window size is increased and decreased when detect lost in packets. In TCP Tahoe, Reno, and Vegas, the congestion avoidance phase algorithm permit to the window size to increase by one segment every RTT. This increment stop when the window size reaches the congestion point and that will stimulates the window size to decrease and slow-down to the next phase.
Basically, TCP seeks to provide reliability to data transmitted between two hosts. TCP is trying to provide reliable data transmission between two entities. TCP applies set of rules to handle lost in packets resulted from physical errors in transmission or because of the congestion in cross traffics [8]. In recent days, the need to provide reliable data transmission over Internet traffics or cellular mobile systems becomes very important. TCP represents the prevailing protocol that provide reliability to data transferring in all end-to-end data stream services on the Internet and many of new networks.

Usually, it's not easy to determine the available bandwidth for TCP packets flow. In fact, it's very complex problem due to the effects of the congestion control of TCP and the network dynamics. These two factors make the proceedings of exact allocation for the packets flow complicated. The approved mechanism to detect the optimum bandwidth to send packets from TCP sender is congestion control [9]. The understood of TCP behavior and the approaches to enhance the performance of TCP in wireless channels have been many difficulties and challenges. In parallel with this, considerable researches dealt with in detail many proposed development and mechanisms to raise the efficiency of the performance of TCP, some of these problems already solved, but many others are still open [10].
The object behind the differences of TCP is that each type has some distinct criteria such as the base TCP has become known as TCP Tahoe. TCP Reno adds one new mechanism called fast recovery to TCP Tahoe. Newreno uses the newest retransmission mechanism of TCP Reno. The use of Sacks permits the receiver to specify several additional data packets that have been received out of order within one dupack. TCP Vegas proposes its own unique retransmission and congestion control strategies. TCP Fack is Reno TCP with forward acknowledgment [11].

© 2005 - 2011 JATIT & LLS. All rights reserved
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3. TCP CONNECTION ORIENTED

TCP is a protocol with reliable connection oriented for end-to-end connections, thus it includes algorithms to confirm the links reliability by requesting the acknowledgment from the receiver. For that reason, TCP used as a transport layer protocol because it provide a connection oriented link and also introduce a reliable delivery for packets transferred over unreliable links. Taking in to account that TCP does not based on the other low network layer, due to TCP variants depend on the specifications of wired channels [25,26]. Generally, TCP is very complicated protocol but it provides reliability, warranty and connection oriented for data streaming over all its applications. TCP working to achieve reliable connection due to it exploits the ACK reply form receiver to determine the needs for packet retransmission. One of important technique supported by TCP protocol by using sliding window size mechanism and variation the size scheme to achieve optimal flow rate control over network with maximum exploitation to network bandwidth. In other side, TCP receiver can regulate the flow of data streaming form the sender by estimate the used window size in a way to avoiding buffer overflow [5], in same time, TCP can provide synchronization to a huge number of connections [27]. The facility of connection oriented within TCP permit the user to send packets only when the link between the two hosts is established, and at this point the TCP allows exchanging data between TCP peers. Rationally, the connections of TCP are including three main phases start with connection establishment, then data transmission, and connection termination [28]. In connection establishment, the client try to associate with the corresponding server, while the server itself must connected to other ports be ready to open it up for connections. This process termed passive open, and when the passive open is established, the client will start the active open. If the server sends ACK to the client in the network, actually, both of client and server have received ACK to proving the connection and firstly a virtual plinth connection established to maintain the data transfer cycle. The last end point between each TCP peers is termed sockets. The socket defined as the integration between the source and the destination of the hosts addresses and ports, where each packet arrived must identified with a corresponding socket and belonging to a particular connection. In the sense that all TCP peers directly connected with others by socket connection and all the packet writing and reading to any socket is how transport layer interfacing with IP layer which below it in layering structure.

4. TCP RELIABLE DELIVERY

The data stream divided to sequence of packets by TCP; also TCP provides guaranteed reliable delivery when packets lost or duplicated and prevents the network to reach congestion state or overloading [7]. TCP's reliable, in-order delivery service, designed for application convenience, comes at a fundamental cost of delaying data delivery to the application. When the network loses one data segment, the receiving TCP must buffer and delay all segments within at least the next RTT, until the sender reacts and successfully retransmits the lost segment [29]. The sequence of packets divides by TCP for data stream sending independently using IP layer. After packets deliver, the packets re-assembly to the ordinary form. This process is very complex because of the incidental lost and damaging in packets within transmission period, and that force TCP to determines the lost packets and retransmission the missing packets again. The collection of these procedures called reliable delivery, where TCP will provide a fully covering for the packets losses, packets duplication, delayed packets, data corruption, transmission synchronization, and data congestion.
Delivering of datagram's between sources and destinations is the responsibility of network layer in the same time the transport layer detects how to employ the network layer to equip point to point connection with error-free to ensure send data from source host to its peer without corruption and with correct sequence. The functional overlap between network and transport layers reduces the stress in networks traffic, thus network layer can also provide reliable delivery services, but the major guarantee become from TCP, where the network get full transmission rate adjustment in same time with guaranty to avoid closed up states and ensuring that all applications operating cooperatively with other connections and terminals.

Spontaneously, a lot of new problems generated when using TCP in heterogeneous networks and many research dealt with TCP on wireless channels proved that TCP is not always suffering from missing packets because of wireless links, due to the support introduce from the lower layers which provide other reliable delivery services for the protocols in upper layer. But the channel allocation state may cause expiring in the timeout period of TCP retransmission and that leads to extra delay in network data queue. Actually, this TCP

© 2005 - 2011 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

retransmission not because lost packets or timeout expiring, but due to physical reasons. However, this spurious timeout will push TCP sender to unnecessary slow start phase and resend segments. Despite this, TCP is possible to provide an oriented reliable delivery for data stream but it doesn't provide an explicit assign for QoS and this defect actually one of main reasons for TCP prosperity by avoiding the features, it has managed to be too efficient and too powerful simultaneously [30]. In new application domains, TCP is very limited and poor in new network technologies such as multimedia applications where its need adaptive and efficient congestion control mechanism because of TCP designed with specific considerations. As an example, if lost in some segments occurred; TCP assumes this lost happened due to link congestion, but that not always happened because in wireless links the segments lost because of the bad connectivity between the source and the user terminal [31].

5. TCP FLOW CONTROL

Functionally, TCP is responsible for duly matching the transmission rate between source and destination in the network because it is very important to send and exchange data between sources and destinations with the exploitation of the maximum capacity of the channel bandwidth to introduce good performance but not at the expense of losing in data stream [32]. The flow control mechanism in classical TCP is simple and conservative. It operates based on buffer occupancy, and does not track application read rate directly [33]. The need to flow and congestion control increased with modern communication systems due to the large rate of data transmission over new networks links where that need for efficient, reliable, and fair utilization with avoiding the collapse resulted by congestion.

Most of Internet applications are use TCP to provide reliable and steady data transmission. TCP assumes window-based to control packets flow and to detect congestion in network, most of TCP versions (except Vegas) based on control the window size (increasing or decreasing) by triggering the sender to change the window size depending on the approach used in congestion avoidance by TCP version [34]. However, each TCP need to base on manually tuning mechanism to tuning the buffers and provide influential scaling to the network path to be able to deals with large bandwidth networks [35].

Sliding window protocol used by TCP to control the flow of data where the TCP window detects the real amount of packets that ready to receive by TCP host in specific time. Or it can be said the sliding window determine the maximum flow rate of packets from TCP sender to receiver without acknowledgment. Each connection include buffer to receive packets but this buffer have limited size and it is very risk if buffers exceeded the allowable limits or if the packets flows faster than the standard read out time of buffers. This problem solved in many transmission protocols, including TCP by using flow control mechanism to avoid the overloading in buffer to occur. The basic concept of flow control based on the receiver directs to the sender about the available capacity in receiver buffer. This mechanism also force the sender to fix the actual quantity of packets that sender should prepare to the link pipeline where it must be equal or less the available space in receiver buffer.

Data transfer between sender and receiver requires high degree of synchronization and compatibility to arrangement the transmission process, thus TCP flow control allows to send packets from sender to receiver with fully matching between them and with taking into account the receiver read rate [36]. Actually, TCP flow control seeking to organize the size of sliding window but the efficiency of flow control diminishes because of the receiver buffer size. In other word, even flow control mechanism can provide large window, the limited size of buffers can degradation the total performance of data transmission [37]. In parallel with flow control, other technique used by TCP to limits the sender transmission rate called congestion control. Flow control not be able alone (or independently) to limits the congestion if happened, because TCP receive a large window of packets while the receive-window restrictive due to flow control not real and definitive factor to arrange sending rate. As a result, the sending rate related with congestion control mechanism and the transmitted data subjects to congestion control conditions [38]. One of the metrics to evaluating TCP performance are the sliding window, flow control, and congestion avoidance and in spite of the advanced steps to improve TCP performance, but many of problems and constraints still face the dream to get typical TCP.

Many research presented introduced a series of modifications to get high performance and an adaptive TCP to be reliable with modern telecommunication networks and the requirements of large bandwidth channels. The modifications become now as a requirements such as slow start

© 2005 - 2011 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

and congestion avoidance mechanisms. These two algorithms added to most TCP variants to solve some risks in bottleneck while congestion happened and the two phases of slow start and congestion avoidance are merged integrally to work together [39].

6. TCP SLIDING WINDOW

When TCP sender permits to send number of packets even not gets acknowledgment, this called sliding window protocol. The sliding window technique allows for multiple packet transmission with retransmission for lost packets automatically, which play the role to avoid unlimited buffer at the receiver. Generally, each transferred packet bearing the number of sequence represents the bytes organization in the packets that happen in sending state, but in receiving, the packets includes the ACK of the bytes which received successfully to the receiver. That mean, the transmission line consists of two-way packets transmission (bidirectional), one for number of sequences and the other for acknowledgments, where the acknowledgments does not send separately by independent packets. In fact, these two functions (sequence and acknowledgment) are very important to facilitate the communication links among TCP hosts to ensure that all sent packets are received successfully and in an orderly sequence. Beyond any doubt, the function of sliding window increase the comprehensive throughput of TCP peers and thus for the network en bloc. The size of the window depends on the following factors [40]:

- The amount of traffic allowed on the network.
- The amount of TCP buffer space the receiver has advertised.

The possibilities and the functions of sliding window allows the TCP sender to pumps the possible packets in channel pipeline without waiting any notification from the receiver and the size (width) of the sliding window represents the packet amount can injected by the sender into network, but without acknowledging the sender frequently. In spite of sliding window allows TCP sender to send packets without acknowledging, but when the sender has not received ACK from receiver for long period, the sliding window forces the sender to refrain sending more packets [41]. However, sliding window requires a lot of resources, and somewhat not considered an ideal choice for real time communication systems [42].

7. TCP CONGESTION CONTROL

The massive and rapid growth in Internet propagation and with the widespread use of TCP/IP, the congestion control mechanism becomes the decisive factor with the rapid growth of Internet population and the intensive usage of TCP/IP protocol suite, the TCP congestion control algorithm has become a key factor which effect on the level of performance and the demeanor of the amount of data stream within networks. Internet congestion is one of the main issues in computer network especially in growing network of internet which necessarily requires being controlled. Congestion happens when the number of received packages of a node is more than its output capacity [2].
TCP Tahoe is the first TCP variant includes the first congestion control algorithm developed by Jacobson and Karels in 1986, followed by series of algorithms but all based on the concepts which presented by Jacobson and Karels. After that, many enhancements and modification are conducted on Tahoe and that leads to design and develop new TCP variants with different congestion window algorithms [43]. The performance of TCP variants are directly affected by its own congestion control mechanisms where the packets amount which transferred over network connections based on the work and the behavior of the congestion control and its role to exploitation the capacity of the network path [44]. RFC 793 is standardized the first TCP version with basic configuration based on scheme of window-based flow control. TCP Tahoe represents the second generation of TCP versions which includes two new techniques, congestion avoidance and fast retransmission. Reno is the third version of first developed series and it's standardized in RFC 2011, extended the congestion control mechanism is by fast recovery algorithm [45].

The implementation of early TCP versions includes a simplified go-back-n modeling but not include any assumption to the congestion control. In this model, the flow rate of data is transmitted from sender to receiver where the sender is not waiting ACK to send new data, but when the receiver is received the segment successfully in right sequence and without error, the receiver send ACK to the sender. In other hand, when the ACK of segments are going to timeout, the sender will retransmit all the segments, beginning from the oldest segment which lost [46]. The other approach to send data by TCP protocol depends on using end-to-end congestion control without adopting the congestion

© 2005 - 2011 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

control of network-assisted connection due to in IP layer there is no explicit feedback to the systems end regarding to the congestion of network. Actually, when TCP connection put data into connection pipe, the data amount is controlled and limited by congestion control of the sender where the congestion window determines the send rate essentially [36]. The window-based congestion control technique employed by TCP tries to adjust the data flow rate speed by adjusting the size of window to avoid the network congestion and tries to provide fair sharing to the bandwidth of the network over all possible connections [47, 48].

All TCP variants are considered the properties and the characteristics of wired networks and not depend on the lower network layers, but certainly, the congestion control of TCP is not give high performance over heterogeneous networks [25]. Not in heterogeneous networks only, but in high bandwidth links too, TCP suffers from poor performance, that mainly because the slow response of congestion control over large bandwidth connection and the poor exploitation of the available bandwidth [13]. Another hurdle, the regular TCP variants because of unawareness of the network conditions, could not able to fully adjustment to the limited resources and not also not able to recognize the lost in packets if it happened by congestion or randomly. All these reasons stand beyond the poor performance of TCP when depend it as a transmission protocol in wireless networks.

In some of standard TCP variants, such as Reno, the congestion control increase exponentially the packets over the connection, where this slow start increasing period must be control to avoid decline the performance of TCP due to the expected overflow of the receiver buffer. One of the ways of TCP modification is based on estimate the available bandwidth to provide fair sharing to all flows and adjusting the window according to the available bandwidth and flows number [49]. Some other TCP variants are count an indicators to accurate estimation to the available bandwidth to adjust the flow rate amount over the paths which connect the sender with its peer. The accurate bandwidth estimation is depending on many complex parameters and factors like the traffic stability of the network the path length.

The mechanism of congestion control classifies into four phases, slow start, congestion avoidance, fast retransmit, and fast recovery. The nature of wireless link is the reason because it the link fails and many research dealt with the performance of TCP over wireless channels has been suggest new approaches and new techniques especially in last ten years. This attention comes from the expansion in modern high-speed wired-wireless networks where this environment is still support TCP was initially developed, where more and more of research and studies still in progress to expand the effective of the TCP domain [49].

8. CONCLUSION AND DISCUSSION

In this article, the design and architecture of TCP explained under transport layer of OSI model. Also, the phases, factors and parameters of TCP investigated in details where that will assist the researchers, students, and other interested centers to understanding the behavior of TCP and the possible approaches to improve the performance over different applications. Most of the data applications are built on top of TCP since TCP provides end-to-end reliability via retransmissions of missing IP packets. TCP was originally designed for wired networks where the packet losses are due to network congestion and hence the window size of TCP is adjusted upon detection of packet losses. However, the packet losses in wireless networks are due mostly to bad radio conditions and not to network congestion. Errors in the air-link are caused by several factors e.g., interference from other sources, fading due to mobility, and scattering due to a large number of reflecting surfaces. Typically, TCP is used in wired communication systems with very small errors probabilities. The error characteristics of wireless channels, however, differ significantly from that of wired channels. Therefore, TCP gives very poor performance if it is directly applied to a wireless communication system. Wired channel are characterized by miniscule packet loss probabilities and randomly spaced errors. In contrast, wireless channels are characterized by time varying packet loss probabilities that are generally much larger than for wired channels. Also, the errors are typically bursty on wireless channels. Most of the TCP variants based congestion control, decreases the cwnd and the transmission rate accordingly, thus a dramatic degradation in TCP throughput can occur. Mobile users explicitly, can significantly affect the TCP throughput due to mobility and handoff that may cause frequent disconnections. Recently, many congestion control protocols have been proposed, especially for streaming multimedia applications. The effective bit error rates in wireless networks are significantly higher than that in wired networks. Since TCP does not have any mechanism to differentiate between congestion losses and wireless random losses, the latter may cause severe throughput degradation.

© 2005 - 2011 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ACKNOWLEDGMENT

This study is sponsored by Universiti Kebangsaan Malaysia (UKM) through the university research grant UKM-OUP-ICT-36-185/2011.

REFRENCES:

[1] "Network Layering," MIT 6.02 DRAFT Lecture Notes, vol. LECTURE 21, 2010.

[2] H. Shirazi, "Smart Congestion Control in TCP/IP Networks," Journal of Information and Communication Technology, vol. 2, pp. 73-78, 2009.

[3] S. Alex, "Computer Network Layers," CIS748 Class Notes.

[4] J. Antila, "TCP Performance Simulations Using Ns2," e-mail: jmantti3@ cc. hut. fi.

[5] B. Qureshi, M. Othmanet, and N. Hamid, "Progress in various TCP variants (February 2009)," pp. 1-6.

[6] S. Henna, "A Throughput Analysis of TCP Variants in Mobile Wireless Networks," 2009, pp. 279-284.

[7] N. Möller, Automatic control in TCP over wireless: School of Electrical Engineering, Royal Institute of Technology, 2005.

[8] B. Moraru, F. Copaciu, G. Lazar and V. Dobrota, "Practical analysis of tcp implementations: Tahoe, reno, newreno," 2003, pp. 125–138.

[9] H. Abrahamsson, O. Hagsand and I. Marsh, "TCP over high speed variable capacity links: A simulation study for bandwidth allocation," 2002, pp. 117-129.

[10] X. Chen, H. Zhai, J. Wang and Y. Fang, "A survey on improving TCP performance over wireless networks," Resource management in wireless networking, pp. 657-695, 2005.

[11] G. A. Abed, M. Ismail, and K. Jumari, "Behavior of cwnd for TCP source variants over parameters of LTE networks," Information Technology Journal, vol. 10.

[12] K. S. Siyan and T. Parker, TCP/IP: Apogeo Editore, 2002.

[13] P. Sharma, "Performance Analysis of High-Speed Transport Control Protocols," Clemson University, 2006.

[14] T. Mahmoodi, "Transport Layer Performance Enhancements over Wireless Networks," University of London, 2009.

[15] H. Wu, Z. Feng, C. Guo and Y. Zhang, "ICTCP: Incast Congestion Control for TCP in data center networks," 2010, p. 13.

[16] P. Sarolahti, "TCP Performance in Heterogeneous Wireless Networks," 2007.

[17] M. Podlesny and C. Williamson, "Providing fairness between TCP NewReno and TCP Vegas with RD network services," 2010, pp. 1-9.

[18] P. Karn and C. Partridge, "Improving round-trip time estimates in reliable transport protocols," ACM SIGCOMM Computer Communication Review, vol. 17, pp. 2-7, 1987.

[19] V. Jacobson, "Congestion avoidance and control," 1988, pp. 314-329.

[20] M. Haeri and A. H. M. Rad, "TCP retransmission timer adjustment mechanism using model-based RTT predictor," 2004, pp. 686-693 Vol. 1.

[21] J. Sing and B. Soh, "TCP New Vegas: improving the performance of TCP Vegas over high latency links," 2005.

[22] G. A. Abed, M. Ismail, and K. Jumari, "Characterization and observation of (transmission control protocol) TCP-Vegas performance with different parameters over (Long term evolution) LTE networks," Scientific Research and Essays, vol. 6, pp. 2003-2010, 2011.

[23] Y. C. Chan, C.T. Chan, and Y.C. Chen, "Performance improvement of TCP Vegas over heterogeneous networks," 2004, pp. 36-41.

[24] Sarolahti, P., TCP Performance in Heterogeneous Wireless Networks. 2007.

[25] Subedi, L., M. Najiminaini, and L. Trajkovi Performance Evaluation of TCP Tahoe, Reno, Reno with SACK, and NewReno Using OPNET Modeler. 2008: Citeseer.

[26] M. Rahman, A. Kabir, K. Lutfullah, Z. Hassan and M. Amin, "Fair comparisons of different TCP variants for future deployment of Networks," 2008, pp. 260-263.

[27] Swales, A., Open Modbus/TCP Specification. Schneider Electric, 1999. 29.

[28] "The TCP/IP Protocol Suite," Fujitsu, vol. Release 1.0, 2006.

[29] J. Iyengar, S. O. Amin, M. F. Nowlan, and N. Tiwari, "Wire-Compatible Unordered Delivery in TCP and TLS," Arxiv preprint arXiv:1103.0463, 2011.

[30] TCP Revisited. 2006.

[31] Goutelle, M., A survey of transport protocols other than standard TCP. 2005.

© 2005 - 2011 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[32] M. Allman, V.P., and W. Stevens, TCP Congestion Control. RFC 2581, 1999.

[33] Sanadhya, S. and R. Sivakumar, Adaptive Flow Control for TCP on Mobile Phones.

[34] G. Bajkó, I. Moldován, O. Pop and J. Bíró, "TCP Flow Control Algorithms for Routers," Department of Telecommunications and Telematics, Technical University of Budapest.

[35] M. Fisk and W. Feng, "Dynamic right-sizing in TCP," 2001.

[36] "Transport Layer," STUDY COMPANION FOR COMPUTER NETWORKING,, vol. Chapter Three.

[37] M. Luglio, C. Roseti and M. Gerla., "The impact of efficient flow control and OS features on TCP performance over satellite links," ASSI Satellite Communication Letter (Sat-Comm Letter), 9th edition, special issue on Multimedia Satellite Communication, vol. 3, pp. 1-9, 2004.

[38] V. Carney, "Message Integrity, Reliable Delivery, Acknowledgements, Non-Repudiation and Encryption."

[39] G. Corral, A. Zaballos, T. Fulgueira and J. Abella, "Simulation-based study of TCP flow control mechanisms using OPNET Modeler," algorithms, vol. 2, p. 8.

[40] L. Chappell, "TCP Sequencing and Sliding Windows," NOVELL CERTIFIED PROFESSIONAL, 2001.

[41] H. Koga, "Dynamic TCP acknowledgment with sliding window," Theoretical Computer Science, vol. 410, pp. 914-925, 2009.

[42] Wojek, G. Dorkó, A. Schulz and B. Schiele, "Sliding-windows for rapid object class localization: A parallel technique," Pattern Recognition, pp. 71-81, 2008.

[43] J. Mo, R. J. La, V. Anantharam and J. Walrand, "Analysis and comparison of TCP Reno and Vegas," 1999, pp. 1556-1563 vol. 3.

[44] P. Sarolahti and A. Kuznetsov, "Congestion control in Linux TCP," 2002, pp. 49-62.

[45] Y. C. Lai and C. L. Yao, "TCP congestion control algorithms and a performance comparison," 2001, pp. 523-526.

[46] S. Fahmy and T. P. Karwa, "TCP Congestion Control: Overview and Survey of Ongoing Research," Citeseer2000.

[47] M. Kodama, G. Hasegawa and M. Murata, "Implementation experiments of TCP Symbiosis: bio-inspired mechanisms for Internet congestion control," Proceedings of CQR 2008, 2008.

[48] T. Iguchi, G. Hasegawa and M. Murata, "A new congestion control mechanism of TCP with inline network measurement," Information Networking. Convergence in Broadband and Mobile Networking, pp. 109-121, 2005.

[49] Q. Wang and D. Yuan, "An improved TCP congestion control mechanism with adaptive congestion window," pp. 231-235.