Middle-East Journal of Scientific Research 21 (9): 1618-1620, 2014
ISSN 1990-9233
© IDOSI Publications, 2014
DOI: 10.5829/idosi.mejsr.2014.21.09.21742

The Evaluation of the Behavior of Computer Networks by Ns Simulator and the Effect of Queuing Systems in the Performance of Especial Networks

[1]Saman Afrasiabi, [2]Mehdi Jafari Shahbaz Zadeh and [1]Ramin tavakoli

[1]Department of Computer Engineering, Islamic Azad University, Kerman Branch, Iran
[2]Department of Electronic Engineering, Islamic Azad University, Kerman Branch, Iran

Abstract: The current study aimed to evaluated the computer networks behavior by NS simulator version 2 (NS-2) and implementation of the network by this simulator and the investigation of the effect of queuing systems in the network performance. Thus, various queuing systems such as CBQ, SFQ, DRR, FQ, RED and Drop Tail are implemented by the purpose simulator. In an elementary scenario are compared with each other and throughput of the network is calculated for each of them. It can be said that the purpose of this paper is depicting the effect of queuing disciplines in the network and selecting a good system and as the selection of the type of optimized queue discipline depends upon the network topology, the results are dedicated for special topology of the network in this paper and is not generalized.

Key words: Computer network · NS-2 simulator network · CBQ · SFQ · DRR · FQ · RED · Drop-Tail

INTRODUCTION

NS computer network simulators are computer network software and an event-based simulator being designed and implemented in California (Berkley) University that can simulate a wide spectrum of the protocols and local and wide networks assembly [1]. The queues are the locations in which the data packages are kept or dropped. In queuing, we need the schedule of the packages, it means that there should be a process on which a decision is made to keep which packages and which one is dropped. This process is raised as "buffer management" and shows a special system that can be applied for order occupy trend in a special queue [1]. The important thing in queue disciplines is the volume of the kept and dropped packages and throughput calculation of the network. The issues related to the efficiency are of great importance. When hundreds and thousands computers are linked to each other, their complex interaction is with unexpected outcomes and most of the time, these complexities lead into the weak efficiency of the network and nobody knows what the reason is. The origin of some of the efficiency problems is excess use of the existing resources in the network. If suddenly excess traffic of the router is delivered, congestion is created and reduces the efficiency [1-5].

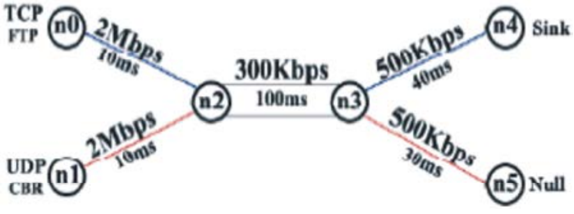
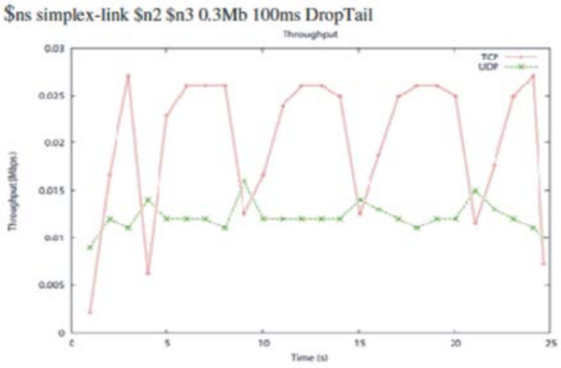


Fig. 1: Topology of this study

Queuing disciplines that are implemented in NS2netowrk simulator are Drop Tail, RED (Random Early Discard), FQ (Fair Queuing), DRR (Deficit Round Robin), SFQ (Stochastic Fair Queuing),CBQ (Class Based Queuing). In a simple scenario, in accordance with Figure 1, n2n3 ink plays as bottle neck of the network in transferring data package. As the sent packages go from node n0 to n4 and sent packages from n1 to n5 are done via this link. Thus, testing different kinds of queuing disciplines is implemented on this link and it is defined how various queuing disciplines by giving turn and scheduling of the received packages and dealing with extra packages, affect the efficiency of the network.

Corresponding Author: Saman Afrasiabi, Department of Computer Engineering, Islamic Azad University, Kerman branch, Iran.



```
$ns simplex-link $n2 $n3 0.3Mb 100ms DropTail
```

Fig. 2: The network performance in Drop-Tail queue

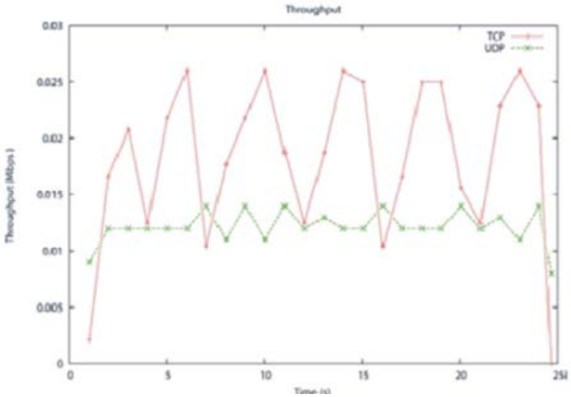


Fig. 3: The network performance in RED queue model

Table 1: The volume of the exchanging packets in megabyte

	TCP		UDP	
	SENT	RECV	SENT	RECV
Drop Tail	0.5138	0.50444	0.306	0.304
RED	0.467	0.45348	0.306	0.304
SFQ	0.58556	0.58556	0.306	0.306
FQ	0.58556	0.58556	0.306	0.306
DRR	0.58452	0.58452	0.306	0.306

The Investigation of the Queue and Queuing Disciplines: In NS2, one of the most important components of Simple Link object in class is Queue. This class models the buffering mechanism for network router. As it stores the sent packages in buffer and when the transference was done, it sends a package inside the buffer to the neighboring object (store and forward) [4]. Queue in NS2 tool has three changing parameters including limit (queue size based on the number of packages), blocked (by default it has false and if it is true, it means that the queue can not send the packages to the neighboring nodes) and unblock on resume (by default it is true and it shows that when the final sent pack is passed, the queue itself should be unblocked). Queuing disciplines as being derived of Queue parent class, have three mentioned parameters [1]. In this scenario, the queue size for link n2n3 is defined as 10 ($ns queue –limit $n2 $n3 10).

Drop Tail (FIFO): It is one of the simplest structures of queuing schedule. In this method, servicing of the packs is done based on their entrance and it is called first in first out (FIFO) or first in- first service (FIFS). One of the important drawbacks of this method is two points: First, an explosive flow can occupy the whole buffer space of such as queue and causes the lack of good servicing to other flows. Second, during the congestion of the packs, this queuing is for the benefit of UDP [2]. In NS2, no varied method, parameter or specific variable are not defined for Drop Tail method [1]. By the following statement and without changing any other parameter, link n2n3 is defined with system Drop-Tail:

Red (Random Early Discard): It is an active queue management technique, RED applies a weighted average of queue size for decision making to drop or mark the packets. During the entrance of the pack, If the average is less than the minimum queue threshold, the arriving packet is queued and If the average queue size is greater than the maximum threshold, the packet is automatically dropped [2, 3]. RED in NS2 has some various varied parameters except the other variables and the most important are mean-packet size (Mean packet size based on byte to update the size of the calculated queue after idle period), Drop tail (implementing Drop Tail policy when the queue is overcrowded or *maxthres –≤–weight*), bytes (determines the data state, if it is true, the data is based on byte and if it is false, the data are calculated based on packet)- queue-in-bytes (if it is false, q-weight- is calculated based on packet)- wait(if it is true, the packet that is dropped between two packets is kept), set bit-(if it is false, the packets are dropped and if it is true, the packets are marked)[1, 3]. In this scenario, all the values are by default, it means that q _weight_=0.002, mean _packet size_=500, maxthres_15, min thres_=5 and the values underlined are default Boolean values.

FQ (Fair Queuing): This technique aims to dedicate equal band width to al the flows in each time as the packets are classed by the systems to some flows and then join the queue that are predicted and in each round robin or quantum, a packet is serviced [2]. In NS2, the only changing parameter of this queue is the time dedicated for each byte (sec Per Byte)).

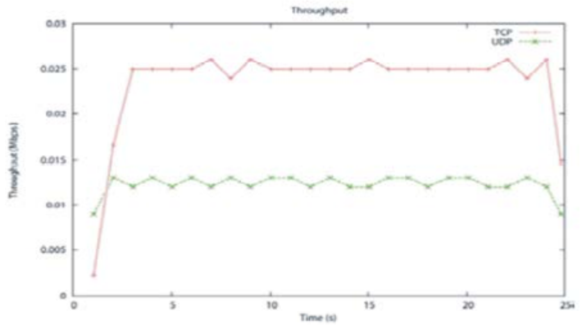


Fig. 4: The performance of the network in FQ queue model

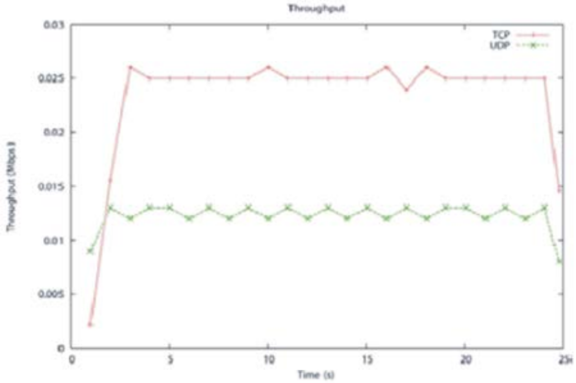


Fig. 5: The performance of network in DRR queue

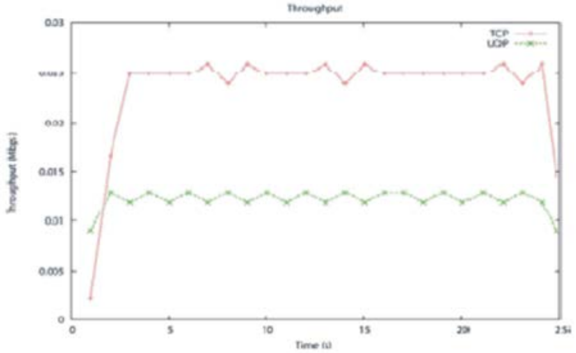


Fig. 6: The performance of the network in SFQ queue

DRR (Deficit Round Robin): The scheduling determines the number of queue bytes, if the packet size is greater than buckets_ (The total number of the bytes that the queue can send in each visiting scheduling) bucket_ is increased as quantum. Otherwise, buckets_ is reduced as the size of the number of packet bytes and the packet is sent. Other parameters of this queue in NS2 are including limit_(the common buffer size in byte default=25000) and quantum (default=250bytes)[1, 2].

SFQ (Stochastic Fair Queuing): It is modified type of FQ by the aim of removing its limitations as this method reduces the number of required queues. One of the most important drawbacks of this method is unfair behavior with the flows colliding with other flows. Thus, as the name reveals, fair is guaranteed as stochastically [6]. In NS2, there are two varied parameters for this queue, buckets_ (with default value 16), max queue_(with default value 40)

Class Based Queuing (CBQ): This method is called also Weighted Round Robin (WRR), the packets are divided into various service classes and are transferred into the queue that is considered for the service and each of the queues are serviced in a quantum [2]. The only varied parameter CBQ in NS2, packet size_ (The maximum packet size based on byte) and the default value is 1024 [7].

CONCLUSION

By investigating the send and receiving rate of the packs (for each of the flows) in each of queue models, some results are achieved and the results are plotted as charts in the required parts. In this part, the statistics of the volume of sent packets from the source and received in the destination are presented. As is shown in Figure 1, it is observed that SFQ,FQ queue disciplines had the best performance as regarding TCP and UDP, all the sent packets reached the destination with maximum volume. Regarding DRR, no packet was dropped. It seems that this queue model didn't have a good performance in facing with congestion and flows collision.

REFERENCES

1. Fall Kevin and Kannan Varadhan, 2007. The ns Manual(formerly ns Notes and Documentation).
2. Hossein Mohammad and Yaqmayi Moqadam, 2009.Computer networks and internet. Ferdosui University publications.
3. Eitan Altman and Tania Jimenez, 2003. NS simulator for beginners.
4. Teerawat Issariyakul and Ekram Hossain, 2009. Introduction to Network Simulator NS2, Springer.
5. Hossein Pedram, Ehsan Malekian and Alireza Zarepour, 2005. Computer networks. Nas publications. Fifth edition.
6. Golestani, 1994. A Self-Clocked fair queuing scheme for boradband applications, IEEE INFOCOM, 94: 636-646.
7. Reza Kargar, Hamid, Mohammad Baqeri Dastgerdi and Majid Akbari, 2010. Comprehensive guide of NS-2 simulatorsoftware.Naquspublications.Firstedition.