Annals of DAAAM for 2011 & Proceedings of the 22nd International DAAAM Symposium, Volume 22, No. 1, ISSN 1726-9679
ISBN 978-3-901509-83-4, Editor B. Katalinic, Published by DAAAM International, Vienna, Austria, EU, 2011
Make Harmony between Technology and Nature, and Your Mind will Fly Free as a Bird
Annals & Proceedings of DAAAM International 2011

CONFIGURATION OF QUALITY OF SERVICE PARAMETERS IN COMMUNICATION NETWORKS

KOZACINSKI, H[rvoje] & KNEZEVIC, P[etar]

Abstract: *Quality of Service with its mechanisms has a deep impact in package transfer in communication networks. This impact is analyzed within this paper trough a simulation scenarios of package transfers. It is shown that combined mechanisms of Quality of Service not only improve package transfer, but also make it predictable and controllable.*
Key words: *QoS, quality of service, communication network, token bucket*

1. INTRODUCTION

Quality of Service (QoS) is a set of requirements that guarantee a certain performance in package transfer. A good implementation of QoS has minimal package delay and a minimal number of discarded packages. To ensure such performance, QoS has a set of mechanisms that help regulate the transfer, so that the performance requirements are met.

A communication network consists of communication nodes and routes between them. The information is sent trough the network within IP packages. One node sends the packages - a sender, the other receives them - a receiver. They are not necessarily connected together, in most cases there are many nodes and routes between them. There are two major problems that appear in communication networks:

1. The architecture of communication networks. Because the communication networks consist of many nodes and routes, it is possible that not all packages are sent trough the same route. Because of this, the following errors can occur:

a. Packages arrive in wrong order via different routes.

b. Jitter because packages arrive in the wrong order, but are delay sensitive. The effect can be heard and seen in video streaming or VoIP communication.

c. Errors in transfer. Packages can be lost in transfer, or they can become corrupt during the transfer. They can also get stuck on congested routes and get delayed. When they are delay sensitive and are delayed too much, they become unusable.

2. In real life situations, the receiver is usually slower than the sender. Senders are clients that connect to servers (receivers). Clearly, all clients cannot be handled in the exact same time and need to wait for the server response. This results in slower transfer and possible package loss.

2. QUALITY OF SERVICE MECHANISMS

Retransmission: A mechanism responsible for resending packages lost in the transmission, thus preventing loss of information.

A buffer: A container used to temporarily store packages that the receiver cannot process upon arrival.

Package priority: A mechanism that classifies packages. Important packages get higher priority than those less important. This classification helps reducing package loss and delays of packages with higher priority.

Policing mechanisms: Mechanisms used to discard packages from the buffer to prevent its congestion. The most common are RED (*Random Early Detection;* also *Discard* or *Drop*) and WRED (*Weighted* RED, uses package priority in discarding).

Shaping mechanisms: Mechanisms used to shape the flow of packages to prevent buffer congestion, package drop and retransmission. The most common are the *Token Bucket* and *Leaky Bucket* algorithms.

3. THE SIMULATOR OF QUALITY OF SERVICE

The goal of this paper is to give theoretical insight of Quality of Service and its mechanisms, but also to back it up with a simple simulation to further illustrate the use of mechanisms, and the effect they have on package transfer. The main focus is on the following mechanisms: retransmission, buffer, package priority and the Token Bucket algorithm, as they are supported by the simulator. They will be further explained, along with pros and cons, trough a scenario of package transfers supported by the simulator.

The simulation covers the basic case of one sender and one receiver connected together. This is because the implementation of all the mechanisms and parameters of a communication network is very complex. It is also simplified deliberately to better illustrate the use of mechanisms of Quality of Service and their effect on package transfer so they can be better understood and interpreted.

4. SIMULATION RESULTS

The simulation was executed five times, with the same basic parameters, but each time adding a new mechanism to improve performance. The basic parameters are: buffer size 60 (1 in the first simulation), 150 packages, sender speed 1, receiver speed 20. After every simulation three graphs are generated: two package delay graphs and a buffer capacity graph.

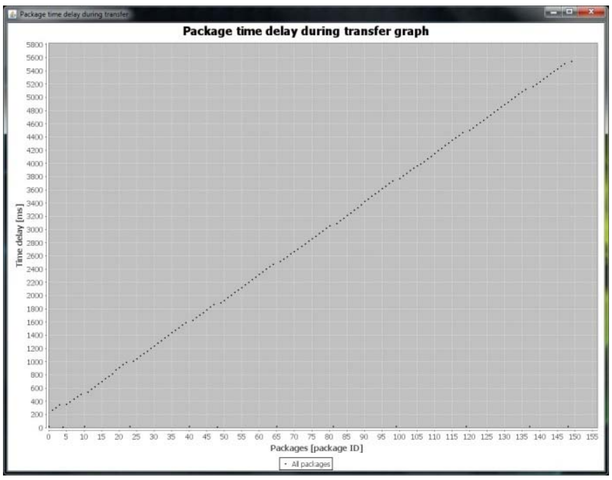

Fig. 1. Buffer time delay graph from the first simulation

Figure 1 shows the first graph of the basic case of package transfer between a sender and a receiver, and the only mechanism activated is retransmission. The delay has a linear growth, because packages keep getting dropped, and it takes time to resend packages in retransmission. That is the main flaw of the retransmission mechanism. After adding the buffer mechanism to the first case, the graph will have two linear growths of time delay, one is the result of packages waiting in line in the buffer, and the other is the result of retransmission, once the buffer gets congested. The use of a buffer did not stop the linear growth of package delay, because packages are collected from the buffer from one side to the other.

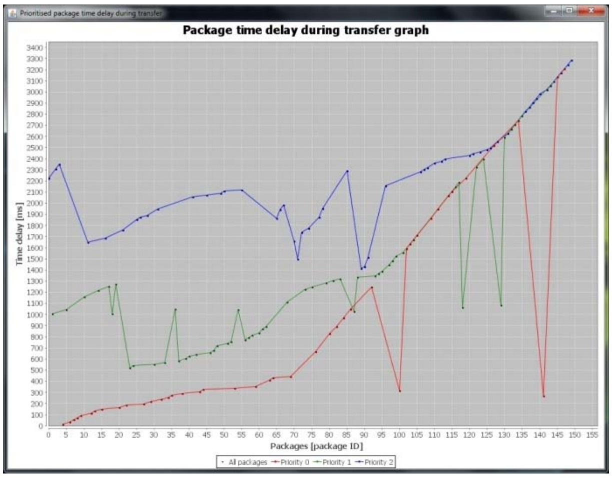

Fig. 2. Buffer time delay graph from the fourth simulation

An improvement to the way the packages are taken from the buffer is achieved with a round-robin algorithm. The packages are being taken from the buffer in a circular way and this can be seen on a simulation graph as an almost horizontal line of package delay, alongside linear growth.

Figure 2 illustrates adding package priority. There are three different priorities, 0, 1 and 2, with 0 being the highest and 2 the lowest. The result of package time delay is shown on the second graph the simulator generates. It shows that priorities

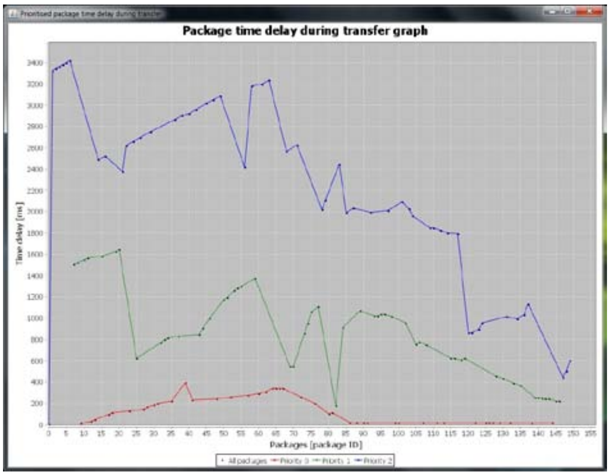

Fig. 3. buffer time delay graph from the fifth simulation

were taken in consideration during the transfer, and that the packages of the highest priority have the smallest time delay, while lower priorities have greater delays. However, this still has not resolved the need for retransmission, which can be seen as a linear time delay increase, starting about halfway on the graph.

Picture 3 shows the situation when all possible mechanisms in the simulator are activated. There is no linear time delay increase because of retransmission, instead there is a drop because of the Token Bucket algorithm, which now controls the transfer flow. The third graph generated after a simulation is complete shows how buffer capacity changes over time during the transfer, so that the user can see how the capacity differs and at which precise moment the buffer became congested.

5. NUMBER CRUNCHING

The basic case with retransmission as the only mechanism gave the following statistics: total transfer time: 6.001 s, number of dropped packages: 138. Other statistics are irrelevant because there are no priorities and there is no point talking about buffer delay times. With buffer added, total transfer time dropped to 4.01 s, number of dropped packages was 81 and average buffer time delay was 1477 ms. After adding the round-robin algorithm and package priority, the numbers were: total transfer time: 4.0 s, number of dropped packages: 81 (21 priority 0, 24 priority 1 and 36 priority 2), average buffer time delay: 1499 ms (885 ms priority 0, 1271 ms priority 1 and 2248 ms priority 2). Finally, when the Token Bucket algorithm was added, the numbers were: total transfer time: 3.5 s, number of dropped packages: 0, average buffer time delay: 1118 ms (133 ms priority 0, 862 ms priority 1 and 2179 ms priority 2).

This shows that combining the mechanisms in the simulator gave better package transfer performance, while combining all the mechanisms gave the best performance. After the simulation, five text documents are generated in the folder where the simulator is located. They display detailed information about all the packages present in the transfer, each with a different criteria.

6. CONCLUSION

As it was demonstrated in the simulation, the package transfer was improved with the use of the QoS mechanisms. The time delay was significantly lowered for some packages, and with the use of the Token Bucket algorithm, no packages were dropped during the transfer. The simulation also showed that combining more mechanisms gives better results. The transfer also becomes more predictable and controllable. This enables guaranteeing set performance parameters, exactly the thing necessary for establishing a good QoS.

7. REFERENCES

System i Networking Quality of Service (Version 6 Release 1), IBM Corp., 2009., *Available from:* http://publib.boulder.ibm.com/infocenter/iseries/v6r1m0/topic/rzak8/rzak8.pdf, *Accessed:* 2011.5.14

Ali Mantar, H. (2000). *The Token Bucket (Leaky Bucket) Model*, Syracuse University, *Available from:* http://qbone.internet2.edu/bb/Bucket.doc, *Accessed:* 2011.5.14.

Chimento, P. F., (2000). *Standard Token Bucket Terminology*

Queue Management, Rutgers University, *Available from:* http://www.research.rutgers.edu/~xili/cs352/queue-management.ppt, *Accessed:* 2011.5.14.

Domokos Botond, B. (2001). *Traffic Shaping*, *Available from:* http://ftp.utcluj.ro/pub/users/tarc/biro.ppt, *Accessed:* 2011.5.15.

Francis-Cobley, P. (2004). *Congestion Control*, *Available from:* http://ftp.utcluj.ro/pub/users/cemil/prc/CONGESTION%20CONTROL.ppt, *Accessed:* 2011.5.15.

Quality of Service Networking, Cisco Press, *Available from:* http://docwiki.cisco.com/wiki/Quality_of_Service_Networking, *Accessed:* 2011.5.28.

Anonymous, (2010). *Komunikacijske mreže, 7. Transportni sloj. Transportni protokoli u Internetu* (*notes from the lesson*)