A SURVEY ON THE IMPORTANCE OF QOS METRICS ACROSS NETWORKING DOMAINS

Juan Flores, Oscar Sandford, and Ben Wunderlich

University of Victoria

ABSTRACT

Write this last.

***Index Terms*—** Quality of service, survey

1. INTRODUCTION

The design of computer networks is compelled by a drive towards a target quality of experience (QoE) and quality of service (QoS). These measures define the success of a network service in a competitive market. It is important to make a distinction between QoE and QoS. QoS is the set of characteristics in a computer networks system that affect its ability to satisfy the utilitarian needs of its users [1]. QoE focuses closer on the end-user, and includes user perception, expectation, and their specific experiences using the service [2]. To expound on that, [1] states that QoS is viewed from a system's perspective, and QoE from a user's perspective. This study will focus on quality of service, as it is easier to evaluate due to its quantitative nature. QoS requirements stem from domain and system requirements, rather than the requirements of an user-end, which are difficult to measure. In the interest of isolating the analysis of a network to its technical requirements and how well companies correctly prioritize the maintenance of their system in order to best fit a generalized view of the customers' needs, we will only consider quality of service (QoS) in this study.

Countless studies in computer networking have addressed quality of service when it comes to the development and analysis of new algorithms, measurements on the efficacy of existing solutions, as well as domain-specific requirements for specific systems. However, various domains are differing standards on quality of service for their applications. It is difficult to correctly understand what it means for an application to have a "good" quality of service, especially under the requirements of the application domain itself, without giving weights, or prioritization, to certain parameters.

This paper surveys a set of well-defined, domain-agnostic quality of service parameters and defines each parameter's importance in the context of different networking domains. Such domains include online multiplayer gaming, peer-to-peer file transfer services, video streaming, and financial applications.

Thank you to Dr. Jianping Pan at the University of Victoria for his teaching and guidance throughout this project and the course.

2. RELATED WORK

Software-defined networks (SDN) address the problem of static architectures by enabling programming and measurement in a dynamic setting. In order to help developers focus more on the design goals and less on tedious implementation tasks, SDN research has inspired the utilization of network emulators [3]. Network emulators are often used to test the quality of service of server architectures in a pseudo-live setting. Experiments with such emulators reveal key quality of service parameters that have been used in domain-agostic settings.

Network emulation using Mininet in order to test quality of service parameters for TCP and UDP has been used in [4, 5]. The metrics presented in [4] include throughput, delay, packet loss, and jitter. The metrics are further defined as follows:

- Throughput (total transmitted data in bits)/(total time taken in seconds)
- Delay (time required to transmit the data from sender to receiver)
- Packet loss (the number of packets not delivered to their destination)
- Jitter (the variance in latency)

The authors used these metrics in a generic setting (i.e. no specific domain) in order to compare TCP and UDP protocols. While this domain-agnostic approach makes for a good start, there are several flaws in their results. Firstly, no packet loss was experienced, and that measurement was thereby ignored. Secondly, previous studies [6, 7, 8] used round trip time (RTT) as a QoS metric, which [4] did not consider.

Another study that investigated load balancing algorithms looked at throughput, response time, and memory utilization [9]. Measuring memory utilization is effective for load balancing, but we are considering quality of service, with the focus on the end user. The weight on the server only matters if it affects the client. This paper reinforces throughput as a

necessary metric, and that response time (or RTT) must be considered.

A different paper [10] proposes a framework titled AweQoS which combines both quality of service tests with security and reliability tests. The framework includes bandwidth speed measurement, delay tests and jitter tests, as well as SYN flood test, UDP flood test, and Slow HTTP flood test.

In order to provide foundation for quality of service analysis in differing domains, [11] created a taxonomy for clustering specification into categories. We will reuse their specification in our approach, as well as extend it in relation to the domains we will be looking at.

3. APPROACH

As we conclude our investigation on common quality of service metrics used in simulated and domain-agnostic settings, we compiled a set of general quality of service metrics as well as policies, according to specifications outlined in [11]. The following hierarchy of parameters is based on [11], with additional notes pertaining to parameters discovered in related works discussed prior.

At this point, we make the distinction between metrics and parameters, where the former is a subset of the latter.

- Metrics
  - Security
  - Performance
    - Timeliness (e.g. delay)
    - Precision (consistency, e.g. jitter)
    - Accuracy (lack of errors, e.g. packet loss)
    - Combinations (e.g. throughput)
  - Relative Importance (cost of given service to user)
- Policies
  - Management
  - Levels of Service (lower QoS to maintain availability)

Note the inclusion of security as a general metric. While many applications simply prioritize speed and reliability, security is key to some application domains. Reliability issues (system faults not as a result of a malicious attacker) are covered by the performance category.

3.1. Project Website

Please see the associated website for progress reports and results at https://oscarsandford.github.io/qoemf/.

4. DOMAINS

4.1. Online Multiplayer Games

4.2. P2P File Transfer

4.3. Video Streaming

4.4. VoIP

5. DISCUSSION

Text here.

6. CONCLUSION

Text here.

7. REFERENCES

[1] Martín Varela, Lea Skorin-Kapov, and Touradj Ebrahimi, "Quality of service versus quality of experience," *Quality of Experience*, pp. 85–96, Mar 2014.

[2] Markus Fiedler, Tobias Hossfeld, and Phuoc Tran-Gia, "A generic quantitative relationship between quality of experience and quality of service," *IEEE Network*, vol. 24, no. 2, pp. 36–41, 2010.

[3] Rogério Leão Santos de Oliveira, Christiane Marie Schweitzer, Ailton Akira Shinoda, and Ligia Rodrigues Prete, "Using mininet for emulation and prototyping software-defined networks," in *2014 IEEE Colombian Conference on Communications and Computing (COLCOM)*, 2014, pp. 1–6.

[4] Akhilesh Sharma and Aakanksha Sharma, "Qos parameter analysis of tcp and udp traffic over open flow enabled software defined network," in *Proceedings of International Conference on Data Science and Applications*, Mukesh Saraswat, Sarbani Roy, Chandreyee Chowdhury, and Amir H. Gandomi, Eds., Singapore, 2022, pp. 13–25, Springer Singapore.

[5] Pinkey Chauhan and Mithilesh Atulkar, "Achieving enhanced network performance in udp and tcp traffic of software defined networking by selecting java based controllers decisively," *International Journal of Innovative Technology and Exploring Engineering*, vol. 9, no. 7, pp. 268–274, May 2020.

[6] SM Shamim, Shahadat Shisir, Ahosanul Hasan, Mehedi Hasan, and Arafat Hossain, "Performance analysis of different open flow based controller over software defined networking," *Global Journal of Computer Science and Technology*, 2018.

[7] Pinkey Chauhan and Mithilesh Atulkar, "Achieving enhanced network performance in udp and tcp traffic of software defined networking by selecting java based controllers decisively," *Int. J. Innov. Technol. Explor. Eng.(IJITEE)*, vol. 9, no. 7, 2020.

[8] Majed Althobyani and Xingang Wang, "Implementing an sdn based learning switch to measure and evaluate udp traffic," *Computers & Electrical Engineering*, vol. 66, pp. 342–352, 2018.

[9] Himanshi Babbar, S. Parthiban, G. Radhakrishnan, and Shalli Rani, "A genetic load balancing algorithm to improve the qos metrics for software defined networking for multimedia applications," *Multimedia Tools and Applications*, 2022.

[10] Zhibin Liu, Ziang Liu, Yuanyuan Huang, Xin Liu, Xiaokang Zhou, and Rui Zhou, "A research of distributed security and qos testing framework," in *2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, 2020, pp. 174–181.

[11] B. Sabata, S. Chatterjee, M. Davis, J.J. Sydir, and T.F. Lawrence, "Taxonomy for qos specifications," in *Proceedings Third International Workshop on Object-Oriented Real-Time Dependable Systems*, 1997, pp. 100–107.