

Thesis no: XXXX-20XX-XX

Relationships between Quality of experience and TCP flag ratios for web services

Bamshad Gholamzadeh Shirmohammadi

Faculty of Computing
Blekinge Institute of technology
SE-371 79 Karlskrona Sweden

This thesis is submitted to the Faculty of Computing at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering. The thesis is equivalent to 20 weeks of full time studies.

Contact Information:

Author(s):

Bamshad Gholamzadeh Shirmohammadi

E-mail: bagh11@student.bth.se

Faculty of Computing

Blekinge Institute of Technology

SE-371 79 Karlskrona, Sweden

Internet: www.bth.se

Phone: +46 455 38 50 00

Fax: +46 455 38 50 57

University advisor:

Professor Markus Fiedler

Faculty of Computing

Blekinge Institute of Technology, Sweden

ABSTRACT

Context. Nowadays one of the most beneficial business in IT area is web services with huge amount of users. The key of success in these type of services is flexibility in terms of providing same quality of services (QoS) and ability of fast troubleshooting when number of users increase rapidly. To achieve these targets, evaluation of the user satisfaction is highly essential. Moreover it is required to link user dissatisfaction to QoS parameters in terms of troubleshooting.

Objectives. The main aim the research is to find an intelligent method for evaluation of the user satisfaction. The method is proposed to estimate quality-of-experience (QoE) without asking users to send their feed back. Connecting to this aim, the second target is finding the definition of function in equations of $QoS = \text{function}(QoE)$. And finally, comparison of the impact of QoS parameters on mobile application users and web site users is the last objective.

Methods. For this research a web-server for video sharing propose is designed. The users can use it via web site or an Android mobile application. The three main QoS parameters (Packet-loss, delay and throughput) are changed gradually. The users are asked to score the mobile application and web site at the same time. In parallel the traffic of web-server is captured and analyzed. Then based on variations in mean opinion scores (MOS) and also changes in TCP flags, the proper patterns for each of the QoS parameters is provided. In this part the QoE is linked to transport layer. For the second objective, the QoE is directly linked to QoS. On the other words the graphs with QoE as horizontal axis and one of the QoS parameters as vertical axis are provided. And finally based on the gradient of these trends, the amount of impact of QoS parameters on mobile application users and web site users is compared.

Results. Based on the results of the research, decrement in SYN and FIN flags and increment in ACK is an alarm for down going user satisfaction. In this situation, the problem is belongs to packet-loss. Increasing in the percentage of SYN is also a signal for user dissatisfaction. In this case, the problem is result of delay. And finally if the web-server problem is about throughput then, SYN, FIN and ACK has up going trends. In all of the cases the rest of TCP flags has not clear up going or decreasing trend.

The correlation between QoS and QoE is formulated. The trends of MOS relative to QoS parameters for mobile phone and laptop are very similar in case of packet-loss. For throughput the mobile phone users are a little more sensitive. The most significant difference between the MOS values for mobile application and web site is belongs to delay. The increment in delay has really big negative effect on mobile application users.

Conclusion. The final method for user satisfaction evaluation is based on the way of variations in the TCP flags. Among all the flags, SYN, FIN and ACK passed the criteria to make the patterns. Moreover the method indicate the problem is belongs to which of the QoS parameters. The correlation between QoE and QoS is formulated. And finally according to these formulas, two separate web-servers for mobile application and web site is recommended.

Keywords: QoE, QoS, TCP flags

ACKNOWLEDGEMENT

First and foremost, I have to thank my research supervisors, Professor Markus Fiedler. Without his assistance and dedicated involvement in every step throughout the process, this paper would have never been accomplished. I would like to thank you very much for your support and understanding over the past year.

CONTENTS

List of figures.....	7
List of tables.....	9
List of abbreviations.....	10
1 Introduction.....	11
1.1 Problem statement.....	11
1.2 Research questions.....	11
1.3 Approach.....	11
2 Related works.....	11
3 Method.....	13
3.1 Web-Server.....	14
3.1.1 Hardware.....	14
3.1.2 Context.....	15
3.1.3 Access points.....	16
3.2 Traffic Shaping.....	16
3.2.1 Packet-Loss.....	16
3.2.2 Delay.....	17
3.2.3 Throughput.....	18
3.3 Traffic Shaping Effects on Videos.....	19
3.3.1 Packet-Loss.....	19
3.3.2 Delay.....	19
3.3.3 Throughput.....	20
3.4 Traffic Monitoring.....	20
3.4.1 Statistics test.....	21
3.4.2 Traffic filtering.....	22
4 Results.....	24
4.1 Variations in TCP flags when user actions is constant.....	24
4.1.1 Packet-Loss.....	25
4.1.2 Delay.....	26
4.1.3 Throughput.....	27
4.2 User action.....	28
4.2.1 Packet-Loss.....	28
4.2.2 Delay.....	29
4.2.3 Throughput.....	30
4.3 QoE.....	31
4.3.1 Packet-Loss.....	32
4.3.2 Delay.....	33
4.3.3 Throughput.....	34
4.4 Correlation between QoE and QoS.....	35
4.4.1 Packet-loss.....	36
4.4.2 Delay.....	36
4.4.3 Throughput.....	37
5 Analysis.....	37
5.1 Packet-Loss.....	37
5.1.1 Web browser.....	37
5.1.2 Mobile application.....	38
5.2 Delay.....	39
5.2.1 Web browser.....	39
5.2.2 Mobile application.....	40
5.3 Throughput.....	41
5.3.1 Web browser.....	41
5.3.2 Mobile application.....	42
5.4 Discussion.....	42
6 Conclusion and future work.....	44
Appendix A.....	46

List of figures

Fig. 1. A photo of raspberry pi board.....	15
Fig. 2. The access points for web-server.....	16
Fig. 3. Variations in PCL Percentage for traffic shaping.....	17
Fig. 4. Variations of delay for traffic shaping.....	18
Fig. 5. Variations of throughput for traffic shaping.....	18
Fig. 6. A snapshot of Wireshark.....	21
Fig. 7. Results of traffic filtering.....	23
Fig. 8. SYN% relative to packet-loss (lab).....	25
Fig. 9. FIN% relative to packet-loss (lab).....	25
Fig. 10. PSH% relative to packet-loss (lab).....	26
Fig. 11. ACK% relative to packet-loss (lab).....	26
Fig. 12. SYN% relative to delay (lab).....	26
Fig. 13. FIN% relative to delay (lab).....	26
Fig. 14. PSH% relative to delay (lab).....	27
Fig. 15. ACK% relative to delay (lab).....	27
Fig. 16. SYN% relative to throughput (lab).....	27
Fig. 17. FIN% relative to throughput (lab).....	27
Fig. 18. PSH% relative to throughput (lab).....	28
Fig. 19. ACK% relative to throughput (lab).....	28
Fig. 20. SYN% relative to packet-loss (real).....	29
Fig. 21. FIN% relative to packet-loss (real).....	29
Fig. 22. PSH% relative to packet-loss (real).....	29
Fig. 23. ACK% relative to packet-loss (real).....	29
Fig. 24. SYN% relative to delay (real).....	30
Fig. 25. FIN% relative to delay (real).....	30
Fig. 26. PSH% relative to delay (real).....	30
Fig. 27. ACK% relative to delay (real).....	30
Fig. 28. SYN% relative to throughput (real).....	31
Fig. 29. FIN% relative to throughput (real).....	31
Fig. 30. PSH% relative to throughput (real).....	31
Fig. 31. ACK% relative to throughput (real).....	31
Fig. 32. MOS relative to packet loss (web).....	32
Fig. 33. MOS relative to packet loss (Cell phone).....	33
Fig. 34. MOS relative to delay (web).....	33
Fig. 35. MOS relative to delay (Cell phone).....	34
Fig. 36. MOS relative to throughput (web).....	34
Fig. 37. MOS relative to throughput (Cell phone).....	35
Fig. 38. Packet-loss=QoS=function(QoE) For laptop.....	36
Fig. 39. Packet-loss=QoS=function(QoE) For Cell phone.....	36
Fig. 40. Delay=QoS=function(QoE) For laptop.....	36
Fig. 41. Delay=QoS=function(QoE) For Cell phone.....	36
Fig.42. Throughput=QoS=function(QoE) For laptop.....	37
Fig.43. Throughput=QoS=function(QoE) For Cell phone.....	37
Fig. 44. SYN relative to MOS (Laptop).....	37
Fig. 45. FIN relative to MOS (Laptop).....	37
Fig. 46. PSH relative to MOS (Laptop).....	38
Fig. 47. ACK relative to MOS (Laptop).....	38
Fig. 48. SYN relative to MOS (Cell Phone).....	38
Fig. 49. FIN relative to MOS (Cell Phone).....	38
Fig. 50. PSH relative to MOS (Cell Phone).....	38
Fig. 51. ACK relative to MOS (Cell Phone).....	38
Fig. 52. SYN relative to MOS (Laptop).....	39
Fig. 53. FIN relative to MOS (Laptop).....	39

Fig. 54. PSH relative to MOS (Laptop).....	39
Fig. 55. ACK relative to MOS (Laptop).....	39
Fig. 56. SYN relative to MOS (Cell phone).....	40
Fig. 57. FIN relative to MOS (Cell phone).....	40
Fig. 58. PSH relative to MOS (Cell phone).....	40
Fig. 59. ACK relative to MOS (Cell phone).....	40
Fig. 60. SYN relative to MOS (Laptop).....	41
Fig. 61. FIN relative to MOS (Laptop).....	41
Fig. 62. PSH relative to MOS (Laptop).....	41
Fig. 63. ACK relative to MOS (Laptop).....	41
Fig. 64. SYN relative to MOS (Cell Phone).....	42
Fig. 65. FIN relative to MOS (Cell Phone).....	42
Fig. 66. PSH relative to MOS (Cell Phone).....	42
Fig. 67. ACK relative to MOS (Cell Phone).....	42

List of tables

Table. 1. Quality specification for videos	15
Table. 2. Traffic shaping for packet-loss.....	16
Table. 3. Traffic shaping for delay.....	17
Table. 4. Traffic shaping for throughput.....	18
Table. 5. Effect of packet-loss on video (laptop).....	19
Table. 6. Effect of packet-loss on video (Cellphone).....	19
Table. 7. Effect of delay on video (laptop).....	20
Table. 8. Effect of delay on video (Cellphone).....	20
Table. 9. Effect of throughput on video (laptop).....	20
Table. 10. Effect of throughput on video (Cellphone).....	20
Table. 11. Statistical test for Wireshark.....	22
Table. 12. Traffic filtering tests results.....	23
Table. 13. Variations in TCP flags for packet-loss.....	25
Table. 14. Variations in TCP flags for delay.....	26
Table. 15. Variations in TCP flags for throughput.....	27
Table. 16. Variations in TCP flags for packet-loss (Real scenario).....	28
Table. 17. Variations in TCP flags for delay (Real scenario).....	29
Table. 18. Variations in TCP flags for throughput (Real scenario).....	30
Table. 19. Variations in MOS relative to packet-loss (laptop).....	32
Table. 20. Variations in MOS relative to packet-loss (Cellphone).....	33
Table. 21. Variations in MOS relative to delay (laptop).....	33
Table. 22. Variations in MOS relative to delay (Cellphone).....	34
Table. 23. Variations in MOS relative to throughput (laptop).....	34
Table. 24. Variations in MOS relative to throughput (Cellphone).....	35
Table. 25. QoS=function(QoE).....	44

List of abbreviations

App	Application
AVG	Average
B	Short stop at the beginning
CN	Command Number
DLY	Delay
LB	Long delay for start of stream
ms	milliseconds
PCL	Packet-loss
QoD	Quality of Delivery
QoDN	Quality of Delivery by Network
QoE	Quality of Experience
QoP	Quality of Presentation
QoS	Quality of Service
RTT	Round Trip Time
SONBA	Subtraction of Number of packets before filter and after filter
V	Long stop with loading
VL	Very long stop with loading

1 INTRODUCTION

1.1 Problem statement

“Web services are client and server applications that communicate over the World Wide Web’s (WWW) HyperText Transfer Protocol (HTTP)”[23]. Today the number of web sites and web based mobile applications with huge number of users is increasing. Finding user satisfaction level is always one of the most essential keys to be successful in these web services. However it is not always easy to ask users to evaluate the service directly. Moreover it may have some bad effects on user to ask lots of questions. So it is really important to have a clear strategy in terms of being aware of user satisfaction level. Regarding to this issue there are lots of researches to propose different ways of evaluation. One of the best methodologies used in this case is finding the correlation between user satisfaction and their actions. In the article called “ Classification of TCP Connection Termination Behaviors for Mobile Web” [11], some of the user actions are related to TCP packets. To make it more clear it is tested what will happen to TCP packets when user does special act like click refresh button, stop and so on. So it is possible to find correlation between user action and variations in TCP flags.

In this research the final aim is, providing a evaluation method for QoE without asking user directly. Therefore it is focused on finding the correlation between user satisfaction and variations in TCP packet flags. Moreover it is also important to find the reason behind user dissatisfaction. Of-course there are lots of reasons that user may not be happy with the service. However in this research the relation between user satisfaction and QoS parameters is going to be formulated. From mathematical aspect it means finding the definition of function in $QoS = \text{function}(QoE)$. This equation can help in terms of troubleshooting.

1.2 Research Questions

1. How to estimate user satisfaction without asking them for evaluation?
2. How to link QoE to QoS? On the other words: what is the definition of function in $QoS = \text{function}(QoE)$?
3. How variations in QoS parameters effect mobile application users and web site users?

1.3 Approach

To answer the research questions, combination of survey and traffic analysis is used. In the survey part, the user is asked to score the the web site and mobile application in different QoS levels. So the correlation between QoE and QoS is resulted from this side. From mathematical point of view it means the definition of function in $QoS = \text{function}(QoE)$. The slope in these formulas is an indicator of the amount of impact of QoS on QoE. A comparison between the slope for mobile application equations and web site is done. The outputs of the comparison is used to find the way of effect of QoS on mobile application and web site users.

In parallel to the survey, the traffic of web server is also monitored. So it has been observed when the user is not satisfied with the service which changes is happened to the TCP flags. Therefore from this dimension the QoE is linked to transport layer. By the usage of this way it is possible to estimate the user satisfaction according to variations in TCP flags changes. Moreover it shows the reason of dissatisfaction is belongs to which of the QoS parameters.

2 RELATED WORK

There are several researches to mapping the correlation between QoS and QoE in IT services. Those which are directly related to this research will be discussed briefly here. A way of capturing user's

perception of quality in streaming of multimedia contents is explained in [2]. In this research a new QoE management methodology which shows how QoE data may be used for the benefit of network operators and service providers is proposed. They employed a statistical modeling technique that correlates QoS parameters with estimate of QoE perception and identifies the degree of impact of each QoS parameter on the user satisfaction.

Certain studies [3] have also suggested that evaluations of QoE from the users should be an indication to the service providers to consider fine tuning and adjusting QoS in order to improve the QoE for the users. This research is based on ConEx; which is a protocol defined by the IETF. The protocol allows the sender of a flow to convey the received Explicit Congestion Notification (ECN) information back into the network. In IPv6, implementation of ConEx happens in an option header with 28 unused bits. These option header bits are used to send ECN feedback into the network. So a real time objective QoE is sent into the network by the end-users.

In [4] EvalVid and NS2 were used to provide estimations for QoS and QoE metrics in WiMax networks. The results indicate that the evaluations of metrics of QoE parameter's variations are very much dependent on the variations in the QoS parameters. A software called mBenchLab which is designed to measure the QoE on tablet and smart phones accessing could hosted web services is presented in [5] that could be useful in studies on the area of QoE for web services. More than these studies some of web service providers try to evaluate user's satisfaction by the usage of survey. For example when we use Skype some time after a session it asks about our opinion. When we chose one score (except maximum score which is 5) it will ask some more questions to figure out what is the reason of bad score. The questions are sometime boring for users and maybe some of them do not answer the questions correctly. Therefore the main aim of this research is to find an alternative method for the evaluation of user satisfaction.

The authors of [11] classified TCP connection termination behaviors for mobile web. A different sequence of web termination flags based on user actions is observed in this research. For instance it is monitored if a user stops/reloads a video web page in Windows, then we will have one or more RSTs from the client. The method used in [11] could be considered as an idea of providing an alternative mechanism for evaluation of user satisfaction. On the other hand, it should be possible to find a correlation between user actions and their satisfaction. The mechanism will be discussed in the research methodology section. Modeling QoE based on QoS and other measurable parameters could be considered as one of the main tasks to correlate QoE to QoS. A conceptual model of QoE in parallel with the classical Internet architecture hourglass model is proposed in [6]. This model included 4 layers: QoE, Quality of Presentation(QoP), Quality of Delivery(QoD) and QoS. By usage of the model the impact of each parameter on QoE could be shown more clearly.

The World Wide Web Consortium (W3C) [7], categorized QoS requirements for a web service as follow: performance, reliability, scalability, capacity, robustness, exception handling, accuracy, integrity, accessibility, availability, interoperability, security, and network-related QoS requirements. Based on W3C the definition of performance is: "The performance of a web service represents how fast a service request can be completed. It can be measured in terms of throughput, response time, latency, execution time, transaction time, and so on. Throughput is the number of web service requests served in a given time interval. Response time is the time required to complete a web service request. Latency is the round-trip delay (RTD) between sending a request and receiving the response. Execution time is the time taken by a web service to process its sequence of activities. Finally, transaction time represents the time that passes while the web service is completing one complete transaction. This transaction time may depend on the definition of web service transaction. In general, high quality web services should provide higher throughput, faster response time, lower latency, lower execution time, and faster transaction time" [7]. So performance as one of the QoS requirements for a web service could be measured from different points of view. Therefore finding the most important performance parameters is one of the most important research questions. There are a lot of researches in the area of QoE for web services. To exemplify the case in [8] they evaluated the effect of page load duration, total session duration and task completion in web-QoE. Another mechanism by the name of Mean Option Score (MOS) is suggested in [9], the research tries to evaluate the user satisfaction level in response to increasing and decreasing response times in login process. However there is no research that covers all considerations in this area like impact of QoP and QoD on final QoE. So in this research the role of all these observations based on the hourglass model that mentioned above will be considered to evaluate the effect of QoS parameters on QoE in web services.

The concept of finding user satisfaction regarding to network performance required lots of researches. In this case there are lots of works that provided different methods. For instance in [13], they “presented a generic QoS/QoE framework for enabling quality control in packet-switched networks”. In their work it has been shown how information gathered from network can be used in terms of finding quality perceived by the user. the result of their work is network management in a real-time situation. One the most important usages of the Internet is video streaming. While the bandwidth required for video streaming is bigger than other services, still there is demand of working on better quality-of-experience in lower network performance.

To exemplify the case the authors of article called “Preserving Video Quality in IPTV Networks ” [14], provided a set of techniques to inject granular QoE control mechanisms into IPTV networks. In [15] the research is focused on Next Generation Networks. They worked on challenges and a possible solution for optimizing end-to-end QoE . In this article, they “ proposed an E2E QoE assurance system that contains two major components: a QoE/QoS performance reporting component installed at TE, and the QoE management component installed at networks and sources.”

In this article the effect of users on TCP flags is going to be categorized. Today the amount usage of mobile phones for watching videos is rapidly increasing. Regarding to to this aspect in [16], a QoE-based carrier scheduling scheme is proposed for multi-service long-term evolution -Advanced networks . Moreover in this dimension according to [17], “In mobile and pervasive computing environments, understanding and measuring users’ quality of experience (QoE) is an important and challenging task .” Based on their research there are lots of investors that are interested to know about user satisfaction about their services. However it is really essential to know the QoE indicator should work based on which parameters. This parameters can change for different services. In this paper we are going to find a pattern for video sharing server which is considered as a multimedia service. For this question based on [18], “multimedia services are typically much more sensitive to throughput, delay, and packet loss than traditional services”.

3 METHOD

Survey and objective study on QoS parameters is considered in research methodology. Mapping user dissatisfaction to the QoS parameters is the main objective of this study. According to the QoE hourglass model which is described in [6], the effect of QoS parameters to final QoE is going to be formulated. However the parameters that have effects on QoE in two other layers (QoD and QoP) should be considered. Therefore all the information belongs to these two layers like operation system and hardware will be mentioned in final outputs. Moreover these parameters will be kept constant or will have purposeful variations. Impact of network performance which is belongs to Quality of Delivery by Network (QoDN), on user satisfaction could be considered as main observation in this research.

This research is applied in Sweden so most of times the network performance is high. By the usage of traffic shapers the low level network performance perception from user point of view is produced. A comparison of shapers is done in [10]; based on their comparison NetEm is the most proper traffic shaper.

In the QoP level; the effect of hardware and software interface should be observed. For this case the open source operations systems like Linux and Android will be used. The hardware information will be mentioned in results to see how it can effect user satisfaction. Finding definition of function in equation of $QoS = \text{function}(QoE)$ is another result of this research. To reach this target in the research methodology all the experiments are based on $QoE = g(QoS)$, on the other words in each survey one of the QoS parameters will be changed and others are constant ($QoE = g(\text{one of the QoS parameters} | \text{other QoS parameters})$).

QoE is often observed over time and QoS can be monitored over time. So by matching the time scales of both measurements links between QoS and QoE can be established. To make it clear a time plan for varying influential parameters on user satisfaction is designed. Then by the usage of traffic shaper the time plan is applied. Finally based on MOS given by users and time plan, the patterns will be considered as output of research.

Asking the user for rating too often may lead to bad user feeling. Therefore a mechanism for evaluating QoE without asking users directly will be proposed. This mechanism is based on methodology used in [11]. In this paper the effect of different type of the user actions on termination flags is classified. For example how killing browser

before page is completely loaded will effect on TCP packet. TCP termination behavior highly depends on the application used by client. Therefore a set of criteria is considered to identify the termination made by the user.

To make the method for evaluation of the user satisfaction. The QoS parameters are varied gradually. Users are asked to score the web site and mobile application. In parallel the web-server traffic is captured. Therefore the way of changes in TCP flags in different user satisfaction level is observed. And finally the correlation between user satisfaction and variations in TCP flags is formulated.

To make it more clear:

- A. Changes in QoS parameters => Changes in MOS
- B. Changes in QoS parameters => Changes in TCP flags

From A and B it is resulted how TCP packets are changed in different level of satisfaction. As it mentioned above there are lots of considerations in the research. To cover all these effects the following structure is considered:

- I A web-server for video sharing with two access points is configured.
- II By the usage of NetEM, The traffic is shaped.
- III The effect of traffic shaping on the videos is tested.
- IV To have the desired traffic the traffic filtering strategy is considered.
- V The way of variations in TCP flags when user actions is constant is analyzed.
- VI The way of variations in TCP flags when the real users are involved is analyzed.
- VII The users scores in different level of QoS parameters are categorized.
- VIII According to the results of the last two steps the final algorithm for user satisfaction evaluation is concluded.

3.1 Web Server

Based on the final aim of the research, providing the proper web server should be considered as a first step. The web server must have the functionality of making variations in QoS parameters. Also all the input and out put traffic must be monitored.

It should be flexible and fast in the case of programing. The web server is video sharing center so it must have enough space.

Also it is important to be connected to the Internet through proper bandwidth. Apart from these features it is going to be used in terms of traffic monitoring and traffic shaping. Therefore accessibility is really essential.

The accessibility must be for users as well. To sum it up, the web server must have enough space for all videos with the possibility of traffic shaping and traffic monitoring and be accessible for the users.

3.1.1 Hardware

Considering all the above features an embalmed board called Raspberry Pi, has been used as a web-server. “The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It’s capable of doing everything you’d expect a

desktop computer to do, from browsing the Internet and playing high-definition video, to making spreadsheets, word-processing, and playing games.” [19].

Mainly there are two version for the Raspberry Pi. The first version is released on February 2012. It has 700 MHz single-core CPU and 256/512 MB RAM. Linux (e.g. Raspbian), RISC_OS, FreeBSD, NetBSD, Plan 9 and Inferno can be installed as operation system. The second version is released on February 2015. It has 900MHz quad-core ARM cortex-A7 CPU and 1GB RAM. The operation systems are same as first version plus it includes Windows 10 an more variations of Linux.

The B Model of first version of the embedded board is used in this project. This model is the higher-spec variant of the Raspberry Pi. This model has 512 MB of RAM, two USB ports and a 100mb Ethernet port. While it is a web-server for video sharing propose, 32 GB memory is used to have enough space.



Fig. 1. A photo of raspberry pi board

3.1.2 Context

As it mentioned above it is a video sharing web page. There are several categorizes like documentary, sports and so on. Users can send a request to upload their videos. Also they can score the whole web server.

One of the purposes of this research is finding the correlation between variations in packet loss, delay and throughput with user satisfaction level. To achieve this purpose the rest of QoS parameters should keep constant. One of the most important QoS parameters that should be constant is quality of videos. In this case all the videos are webm format and have same quality. In the table shown below the different parameters of all videos are mentioned. All of them has the same quality as shown in the table.

Video	
Dimensions	640*360
Codec	On2 VP8
Framerate	25 frames per second
Bitrate	N/A
Audio	
Codec	Vorbis
Channels	Stereo
Sample Rate	44100 Hz
Bitrate	N/A

Table. 1. Quality specification for videos

3.1.3 Access Points

The server is accessible from ordinary browsers and a web based mobile application. Raspbian Debian Wheezy is used as a operation system. To make it a web-server the package called is installed.

It is a open source package consisting mainly of the Apache, MySQL, and interpreters for scripts written in the PHP and Perl. By the usage of PHP, HTML, Java script and MySQL as database the web page and mobile application is built.

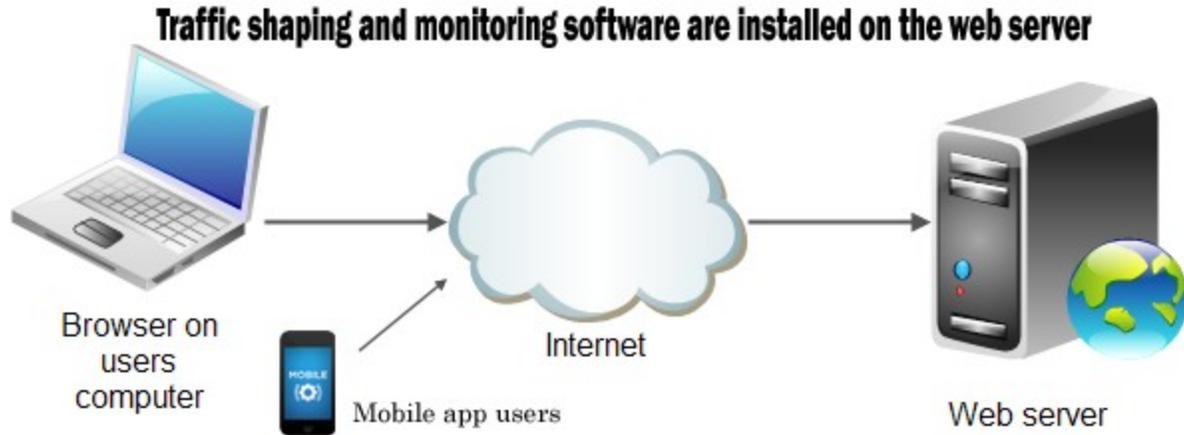


Fig. 2. The access points for web-server

3.2 Traffic Shaping

There are three QoS parameters in the research, Packet loss, Delay and throughput. For each of them there is a different method for traffic shaping. Also for each of them, a proper way for testing the traffic shaping is considered. Each of the QoS parameters are varied in 8 scales. So totally there are 24 sages. In each stage only one of the QoS parameters is changed. For example packet-loss is varied from 7% to 14%.

3.2.1 Packet-Loss

A comparison of shapers is done in [10]; based on their work, NetEm is the most proper traffic shaper. In this research, NetEm is used as a traffic shaper in terms of generating the traffic with specified amount of packet loss.

A testing methodology is considered to get sure that it is possible to shape the traffic purposefully. In the case of packet loss, for each percentage, by the usage of ping service ten batches of packets are sent to the server. Each batch include 10000 packets. In each test the number of received packets is divided to total number of packets. So the percentage of packet loss is tested ten times for each command. By the usage of this method first it is possible to prove the possibility of traffic shaping. And secondly it shows the level of accuracy of the traffic shaper. In the next table the results of the tests for each command is mentioned.

Test Number	1	2	3	4	5	6	7	8	9
AVG of PCL Percentage	0	7.04	7.93	8.88	9.91	10.9	11.89	12.99	14.03

Table. 2. Traffic shaping for packet-loss

According to the above table, by the usage of NetEm the smallest unit for packet loss that can be varied is 1 percent. So in next step for finding the proper ranges for packet loss, the ranges should be divided based on percentages as a unit of QoS parameter.

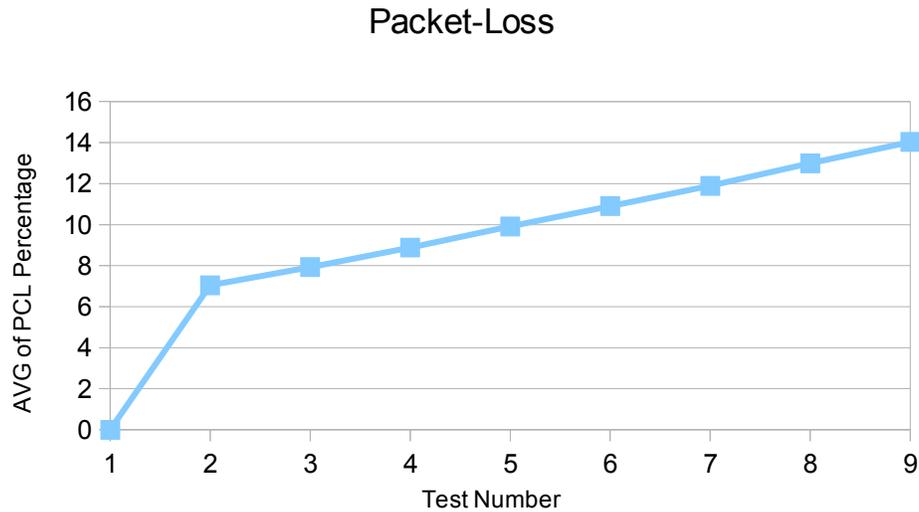


Fig. 3. Variations in PCL Percentage for traffic shaping

Above graph illustrates that by the usage of NetEm, the stages for variations in packet-loss are successfully provided.

3.2.2 Delay

Similar to packet-loss, NetEm is used in terms of making variations in delay. By the usage of ping service, the accuracy of traffic shaping is tested. As it can be seen in the next table, without any traffic shaping the web-server has around 0.8 ms average RTT. Then by adding 1 ms delay in each stage is is increased accordingly.

Delay (ms)	Round Trip Time		
	Minimum RTT (ms)	Average RTT (ms)	Maximum RTT (ms)
1) 0	0.671	0.823	1.237
2) 1	1.691	1.849	2.108
3) 2	2.712	2.861	5.519
4) 3	3.712	3.860	5.490
5) 4	4.699	4.859	5.203
6) 5	5.726	5.857	6.213
7) 6	6.707	6.862	7.495
8) 7	7.723	7.861	8.097
9) 8	8.723	8.865	9.128

Table. 3. Traffic shaping for delay

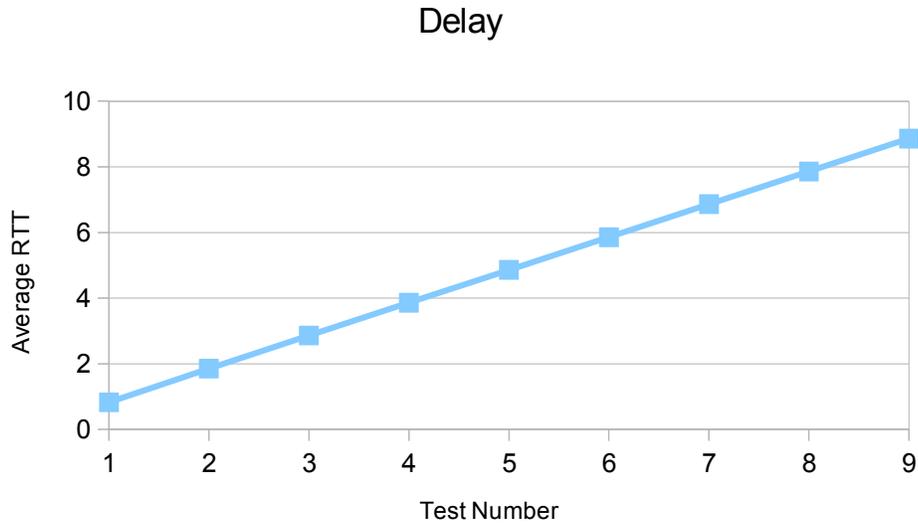


Fig. 4. Variations of delay for traffic shaping

This graph shows that successfully the amount of delay is increased by the usage of NetEm.

3.2.3 Throughput

Traffic control (tc) is also used in terms of changes in throughput. For testing the traffic shaping in this case “wget” program has been used. In each test a similar file is downloaded. Wget, shows the speed of download. So by the usage of this method, throughput of web-server in each stage is calculated.

Test Number	1	2	3	4	5	6	7	8	9
AVG of Throughput (b/s)	1127006	1121278	1113050	1105740	1106592	1102879	1097639	1090553	1079117

Table. 4. Traffic shaping for throughput

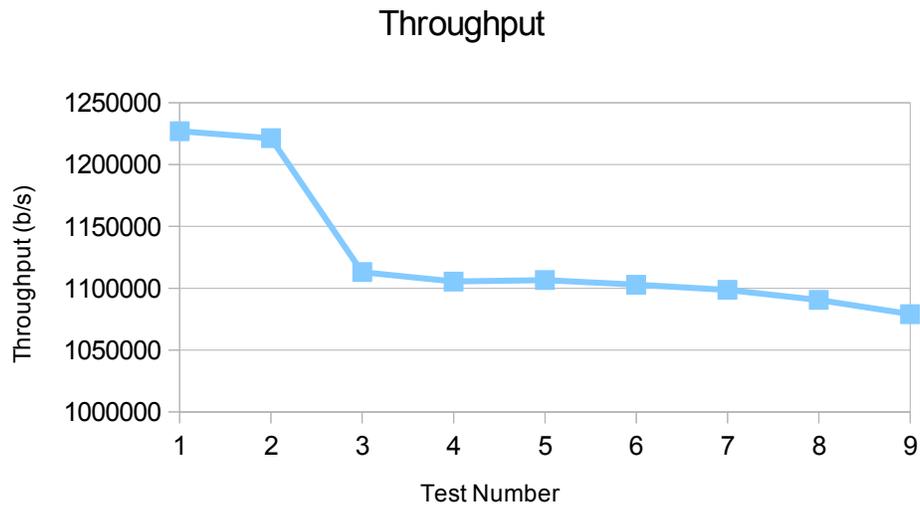


Fig. 5. Variations of throughput for traffic shaping

This graph shows the amount throughput is decreased gradually by the usage of traffic control command.

3.3 Traffic shaping effect on videos

In this part the effect of variations in QoS parameters on videos is discussed. These experiences are done completely manually. The main reason of the test, is to get sure the variations in QoS parameters has effect on videos. For both mobile phone and laptop, the wireless network called Eduroam is used. The effects on videos proves that the variations in QoS parameters overcomes the changes in the wireless network.

To find this effect the smallest unit for variations is figured out in previous session. For each test, number of times a special video in specified period of time is stopped is counted. Each test is ten times repeated. For example, when packet loss is 10% it has been observed that mostly there are four stops in the video during the time. There are two different access methods to the web server that are mobile app and web browsers. The tests are done for both of them.

3.3.1 Packet-loss

Packet-loss	1) 0%	2) 7%	3) 8%	4) 9%	5) 10%	6) 11%	7) 12%	8) 13%	9) 14%
Number of stops in the video (laptop)	0	0	2	1B2L1	1B5L2	3L2VL	1L4VL	5VL	6VL

Table. 5. Effect of packet-loss on video (laptop)

Packet-loss	1) 0%	2) 7%	3) 8%	4) 9%	5) 10%	6) 11%	7) 12%	8) 13%	9) 14%
Number of stops in the video (Cell phone)	0	0	1B	1LB	1LB1L	1LB1L	1LB2L	1LB3L	1LB4L

Table. 6. Effect of packet-loss on video (Cellphone)

Based on the above tables variations in packet-loss is resulted to 0 to 6 very long stops for laptop and 0 to 5 stops for mobile phone. So it is expected that mobile phone users get more effected by packet-loss relative to web browser users.

3.3.2 Delay

The effect of delay on videos is different with packet-loss. In packet-loss we have couple of pauses when the QoS is low. In delay in most of cases there is only one stop and in is some of them maximum two or three. But in delay the duration of stop will increase by increment in delays in packets. On the other words the number of stops is mostly one (at the beginning of video) but it will be bigger when delay is increased. So in this case instead of counting the number of pauses, the total amount of delay is calculated. In the table shown below, D means the amount time that the video took more than its real duration. For example if the video without any delay take one minute and thirty seconds but after delay it take one minute and thirty five seconds, then the D is five.

Delay	0 ms	100 ms	200 ms	300 ms	400 ms	500 ms	600 ms	700 ms	800 ms
Delay in the video (laptop)	0	0	D<1	1<D<2	2<D<3	3<D<4	5<D<6	6<D<7	D>8

Table. 7. Effect of delay on video (laptop)

Delay	0 ms	100 ms	200 ms	300 ms	400 ms	500 ms	600 ms	700 ms	800 ms
Delay in the video (Cell phoen)	0	D<1	2<D<3	8<D<9	13<D<14	17<D<18	20<D<21	23<D<24	D>24

Table. 8. Effect of delay on video (Cellphone)

So delay also has more negative effect on videos in mobile application relative to web browser.

3.3.3 Throughput

Throughput (b/s)	1127006	1121278	1113050	1105740	1106592	1102879	1097639	1090553	1079117
Number of stops in the video(laptop)	0	0	2	1B2L2	1B3L4	1L1VL	1L3VL	4VL	3L2VL

Table. 9. Effect of throughput on video (laptop)

Throughput(b/s)	1127006	1121278	1113050	1105740	1106592	1102879	1097639	1090553	1079117
Number of stops in the video (Cellphone)	0	0	1B	1LB	1LB1L	1LB2L	1LB3L	1LB4L	1LB5L

Table. 10. Effect of throughput video (Cellphone)

In case of throughput, it looks generally the variations have similar impact on both laptop and mobile phone.

It should be mentioned these ranges are not used for the survey. They are only used to get sure that variations in QoS parameters has clear effect on quality of videos. In the final survey. Packet-loss is changed from 7% to 14%, delay from 500ms to 1200ms and finally the throughput from 1127006 (b/s) to 1105740 (b/s).

3.4 Traffic Monitoring

In this research the changes in TCP header in different user satisfaction level is going to be formulated. So here one the most essential issues is finding the proper packet monitoring methodology. For this aspect Wireshark is chosen as a traffic monitoring tool. Wireshark works really similar to tcpdump, but has graphical user interface as well. “Wireshark is a network protocol analyzer. It lets you capture and interactively browse the traffic running on a computer network. It has a rich and powerful feature set and is world's most popular tool of its kind. It runs on most computing platforms including Windows, OS X, Linux, and UNIX. Network professionals, security experts, developers, and educators around the world use it regularly. It is freely available as open source, and is released under the GNU General Public License version 2.” [20]

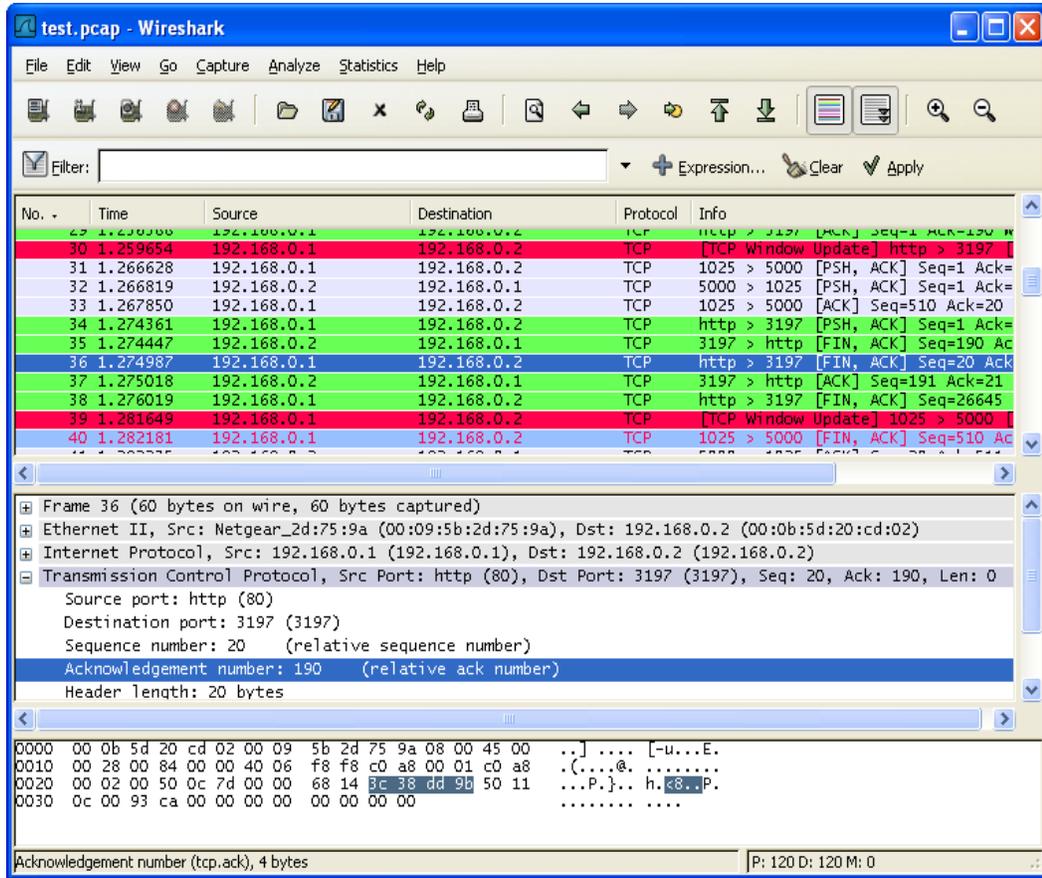


Fig. 6. A snapshot of Wireshark

To make sure that Wireshark works accurately the statistics of Wireshark in terms of TCP flags percentages are tested. In this test it is proved that all the Wireshark statistics are valid and accurate. In next two sessions the tests are discussed.

3.4.1 Statistics test

Accuracy of statistics is another important concern in terms of traffic monitoring. Regarding to this issue a testing system is provided. The system is a combination of one shell script and four functions that are created by the usage of C programming. It is possible to export all the Wireshark packets as a C array. However the .C file which is output of Wireshark, only contains the packets as char variables. To make it more clear it does not follow a C code syntax. In the first step the packets are exported as a C array. This C array is used as input of system. By the usage of shell script the C array is inserted to the first C code. The output of first C code is a text file with one long string. This string contains all the information of Wireshark packets in hexadecimal format. This long string is used as input of second file. Then after this step, all the hexadecimal values are transformed to binary format. Also instead of one long string the values are in different lines. By the usage of third C file the information about TCP flags are filtered. On the other words only 9 binary digits for TCP flags are left for each packet. In the fourth C file the total number of packets and also number of packets with each TCP flag up is counted. In this step it will find the pattern of TCP flags. For example if one line contains "100000001" one NS and one FIN flag up is counted. So by dividing number of packets with each flag up to the total number of packets, the percentage of each flag is resulted. In the next table the comparison of Wireshark statistics with the system is provided.

Test Number	Method	TCP Flags percentage								
		NS	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN
1	Wireshark	0	0	0	0	95.74	13.91	0	8.51	0.33
	Test System	0	0	0	0	95.74	13.91	0	8.51	0.33
2	Wireshark	0	0	0	0	97.22	10.74	0	3.98	2.77
	Test System	0	0	0	0	97.23	10.75	0.69	3.99	2.77
3	Wireshark	0	0	0	0	96.23	12.05	0	5.08	2.44
	Test System	0	0	0	0	96.23	12.05	0.94	5.08	2.45
4	Wireshark	0	0	0	0	93.60	14.14	0	12.45	1.01
	Test System	0	0	0	0	93.60	14.14	0	12.46	1.01
5	Wireshark	0	0	0	0	93.32	12.99	0	10.46	2.52
	Test System	0	0	0	0	93.32	13	0.44	10.47	2.53
6	Wireshark	0	0	0	0	97.38	11.21	0	4.45	1.53
	Test System	0	0	0	0	97.39	11.21	0.31	4.45	1.54
7	Wireshark	0	0	0	0	95.17	11.83	0	5.05	1.14
	Test System	0	0	0	0	95.17	11.84	0.30	5.06	1.15
8	Wireshark	0	0	0	0	95.05	12.74	0	9.23	4.17
	Test System	0	0	0	0	95.05	12.75	0.33	9.23	4.18
9	Wireshark	0	0	0	0	92.96	18.62	0	13.93	4.41
	Test System	0	0	0.28	0.28	92.69	18.62	0.28	13.93	4.55
10	Wireshark	0	0	0	0	96.77	6.56	0	6	0
	Test System	0.33	0	0.67	0.67	96.44	6.56	0.56	6.34	0.78

Table. 11. Statistical test for Wireshark

The above table shows that all the statistics are almost same in both Wireshark and the testing system. However there are some small differences that are not more than one percent. So according this observation it is concluded that one percent confidence interval should be considered for Wiresharks statics.

3.4.2 Traffic filtering

As it mentioned before, one of the main target of the research is to find the correlation between user satisfaction and changes in TCP packets. Regarding to this purpose, it is really essential to have a clear filtering method. The web server is connected to a router so always there are some traffic which are not related to the web page. Therefore a filtering string is considered so Wireshark filter the wanted traffic from the rest. To get sure that the filter works correctly a testing method is considered.

In terms of testing, all the traffic of web server with and without any connection to the web page is monitored for ten times. Each time has special number of connections to the web server. So by finding the differences in number of packets after filtering and before filtering in all the tests, it can be proved we keep only the desired traffic. In the table shown below all the out put of tests are mentioned.

Number of Connections	Number of Packets before Filter	Number of Packets after filter is applied.	SONBA
1. One Connection	2769	1074	1722
2. Two Connections	4795	2847	1948
3. Three Connections	7093	4421	2672
4. Four Connections	8745	5525	3220
5. Five Connections	9996	7871	2125
6. No Connection	2815	0	2815
7. No Connection	2227	0	2227
8. No Connection	2764	0	2764
9. No Connection	2670	0	2670
10. No Connection	1744	0	1744

Table. 12. Traffic filtering tests results

According to the above table, in tests that there is no connection to the web page, the number of packets after filter is zero. But in the tests that the web page is connected by users, always there is some traffic after filter. The average of Subtraction of the number of packets before filter and after filtering in first five tests (the tests that web page is browsed) is 2337. And the average of total number of packets in the test number 6 up to 10 (the tests that web page is not browsed) is 2444. The two averages are almost same.

So based this observation it is concluded that after filter we have the traffic which is exactly related to the research (Not all the traffic). However all the monitored traffic without filtering are always saved.

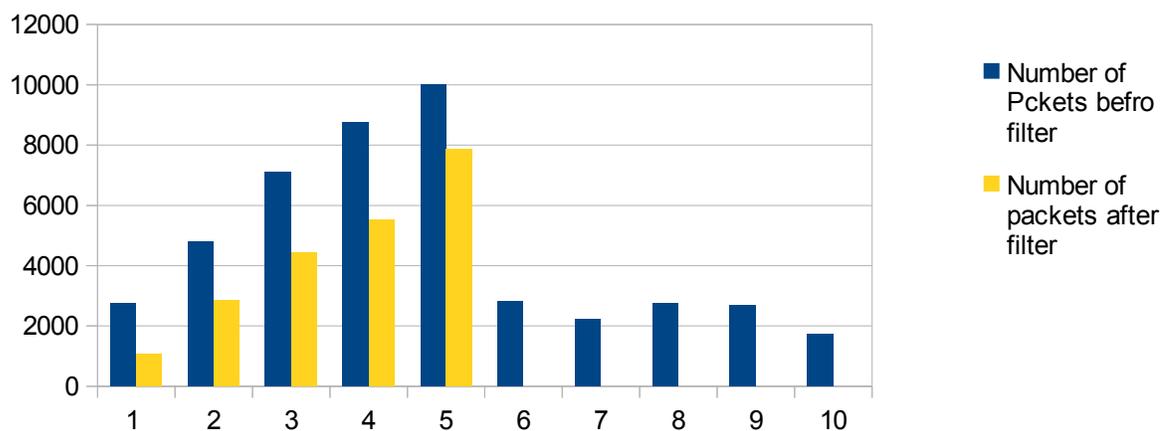


Fig. 7. Results of traffic filtering

4 RESULTS

When users are not satisfied with the web server, it is expected that they do some actions like pressing the refresh button, closing the window and so on. According to [11] these actions has special effects on TCP packets. In this research the final aim is to map the user satisfaction level to the changes in TCP packets.

A TCP packet is combination of a segment header and data section. There are ten mandatory and one optional fields in the TCP header. The first 32 bits are for source port and destination port. Then there are 32 bits that are considered as sequence number. This field has dual role based on SYN flag. If the SYN flag is up, then it is the initial sequence number. Otherwise it is the aggregate sequence number of the first data byte of the segment for the current session. The next 32 bits are belongs to acknowledgment number. When the ACK flag is set to one, this field shows the next sequence number that the receiver is expecting. The fifth field called data offset. This field determine the size of TCP header. After data offset three bits are reserved to use in future. The TCP flags occupy the seventh field in the header. This field is 9 bits and include 9 flags. So the flags can be set to 0 or 1. These TCP flags are used to make the final patterns in the research. The explanation of TCP flags and more details about the header is described in the appendix A. After the flags 16 bits are allocated for window size. It specifies the window size that the sender is willing to receive. Finally the last two mandatory fields are checksum and urgent pointer. The length of these two fields is 16 bits. The checksum is proposed for error checking of the header and data. If URG flag is up, urgent pointer indicate the last urgent data byte.

To find the correlation between TCP flags and QoE, the QoS parameters are gradually varied during specified period of time. In each stage the users are asked to score the web server. Also in the same time all the traffic of the web server is captured. So by the end of the survey it has been observed how TCP packets has changed in different user satisfaction level. In each stage the percentage of TCP flags is calculated. So the final patterns are based on MOS and the percentages of TCP flags.

As it mentioned above in each stage the variations in percentages of TCP flags should be calculated. Filtering the desired traffic is the first step of this process. The server (Raspberry pi) receive all the traffic from router. But only those that are related to web-server are desired for this research. While the percentage of TCP flags are used to make the patterns, the first criteria for filtering is TCP. Web browsers and the web based mobile application are the two access points of the web-server. So the port number 80 is considered as second filtering criteria. Considering both these two limitations, "tcp.port == 80" is used as wireshark string to filter desired traffic. The proof of this filtering method is described before.

Then the number of all TCP flags is calculated. For this case again Wireshark filtering strings are used. To exemplify the case "tcp.flags.ack == 1" is used for finding the number of packets with acknowledgement flag up. Then this number is divided to total number of packets after "tcp.port == 80" filtering. So by the usage of this method the percentage of each of the flags is resulted. In the rest of this session the way of variations in the flags in relation with changes in QoS parameters is discussed.

4.1 Variations in TCP flags when user actions is constant

Before making the patterns according to changes in TCP flags, it should be cleared if these changes are belongs to the user actions. For example maybe when the ratio of packet loss is increased, automatically the packets are resent. So the user is happy but we have increment in number of acknowledgements. So before start of the final step, it is necessarily to find, how other parameters together effect TCP flags. therefore the same as final survey the QoS parameters are varied and the web server traffic is monitored. As it mentioned here the effect of other parameters on TCP flags is going to be analyzed when the user actions is constant. So the user behavior is always similar. For example when the QoS is perfect he do the same thing like when it is too low. For each stage the actions of eight users is simulated. Each user open the browser, watch three videos and then close it. So all the changes in TCP packets in this scenario happen when the user actions are constant. In the next part the real users with different actions relative to QoS level are involved. So by the comparison between these two experiments, the effect of user actions on TCP flags is concluded.

The most important impact in this test is belongs to TCP behavior. CUBIC TCP that is one of the versions with optimized congestion control algorithm is used in the server. It is the improved version of BIC. The algorithm is basically designed for networks with high bandwidth. “CUBIC is one of the most popular version of TCP being used by many flavours of Linux now a days” [22]. The popularity is the reason that CUBIC TCP is used for the server. Therefore the outputs of the research can be used for bigger number of servers.

It should be emphasized, this research is not aimed to formulate the CUBIC TCP behavior. Because determination of the version reactions need more consideration which is out of the aims of this research.

In the following tables and figures the impact of all the parameters (Including TCP CUBIC behavior) when user actions are constant is illustrated.

4.1.1 Packet-loss

packet-loss	TCP Flags ratio in percentage								
	NS	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN
1) 0%	0.0000	0.0000	0.0000	0.0000	99.4067	13.0437	0.0000	0.1949	0.1413
2) 7%	0.0000	0.0000	0.0000	0.0000	99.8369	6.9908	0.0000	0.1046	0.0615
3) 8%	0.0000	0.0000	0.0000	0.0000	99.7657	6.3668	0.0000	0.1246	0.0783
4) 9%	0.0000	0.0000	0.0000	0.0000	99.7698	6.1244	0.0000	0.1236	0.0775
5) 10%	0.0000	0.0000	0.0000	0.0000	99.8350	6.4752	0.0000	0.1204	0.0726
6) 11%	0.0000	0.0000	0.0000	0.0000	99.8340	6.6900	0.0000	0.1239	0.0747
7) 12%	0.0000	0.0000	0.0000	0.0000	99.8574	6.4697	0.0000	0.0992	0.0584
8) 13%	0.0000	0.0000	0.0000	0.0000	99.8796	6.1952	0.0000	0.1022	0.0599
9) 14%	0.0000	0.0000	0.0000	0.0000	99.8768	6.3646	0.0000	0.1098	0.0624

Table. 13. Variations in TCP flags for packet-loss

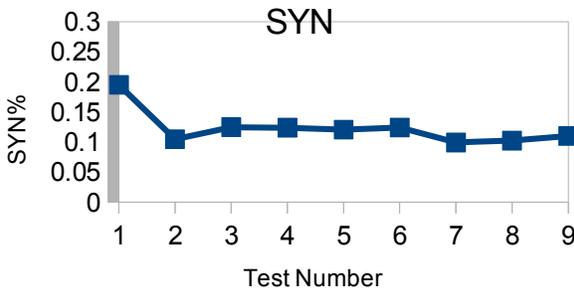


Fig. 8. SYN% relative to packet-loss (lab)

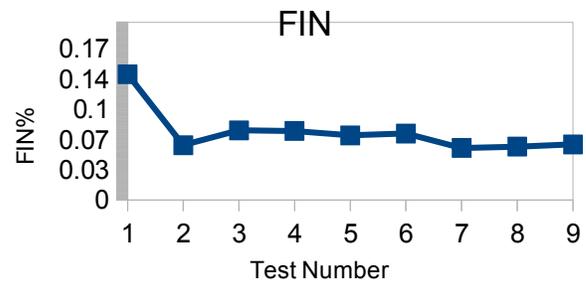


Fig. 9. FIN% relative to packet-loss (lab)

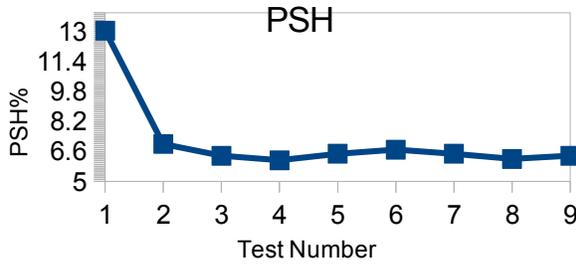


Fig. 10. PSH% relative to packet-loss (lab)

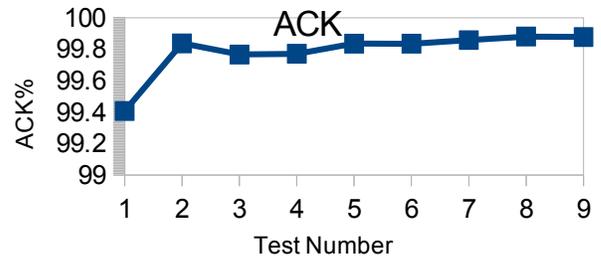


Fig. 11. ACK% relative to packet-loss (lab)

In the above diagrams there is no traffic shaping in the first test. As it can be seen in all of the graphs there is a big difference before and after traffic shaping. In SYN, FIN and PSH there is rapid decreasing from second test. However for packets with acknowledgement flag up it is exactly opposite of others. In all of them when the traffic shaping is applied there is no increasing or decreasing trend. On the other words the percentages of TCP flags always fluctuate around constant value.

4.1.2 Delay

Delay (ms)	TCP Flags ratio in percentage								
	NS	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN
1) 0	0.0000	0.0000	0.0000	0.0000	99.4067	13.0437	0.0000	0.1949	0.1413
2) 100	0.0000	0.0000	0.0000	0.0000	99.1440	2.1512	0.0000	0.4582	0.3184
3) 200	0.0000	0.0000	0.0000	0.0000	99.2486	1.7070	0.0000	0.4031	0.2714
4) 300	0.0000	0.0000	0.0000	0.0000	99.2107	2.1811	0.0000	0.5351	0.3351
5) 400	0.0000	0.0000	0.0000	0.0000	99.3273	2.1803	0.0000	0.4947	0.3121
6) 500	0.0000	0.0000	0.0000	0.0000	99.3137	2.2119	0.0000	0.5037	0.3139
7) 600	0.0000	0.0000	0.0000	0.0000	99.1750	2.3101	0.0000	0.5215	0.3135
8) 700	0.0000	0.0000	0.0000	0.0000	99.2576	2.4656	0.0000	0.5731	0.3461
9) 800	0.0000	0.0000	0.0000	0.0000	99.2536	2.2845	0.0000	0.5714	0.3143

Table. 14. Variations in TCP flags for delay

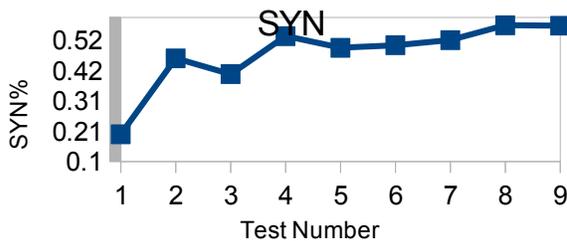


Fig. 12. SYN% relative to delay (lab)

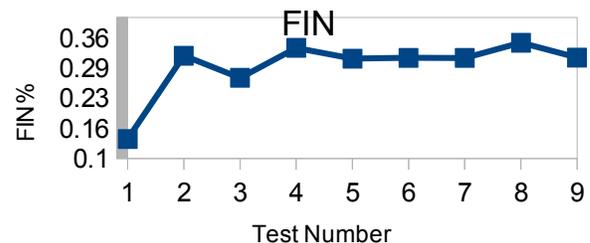


Fig. 13. FIN% relative to delay (lab)

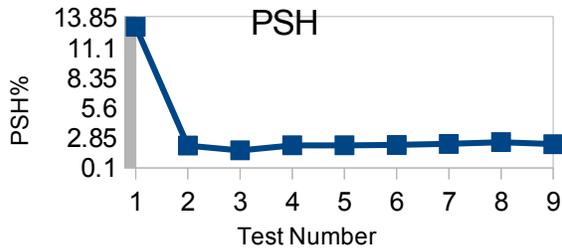


Fig. 14. PSH% relative to delay (lab)

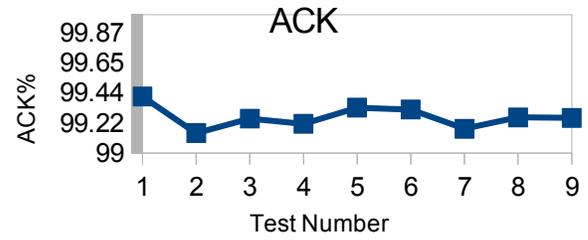


Fig. 15. ACK% relative to delay (lab)

The reaction of TCP to delay is similar to packet-loss for PSH flag. The amount of ACK packets is almost similar before and after traffic shaping. Also there is no up going or down going trend for this flag. There is a big increment after traffic shaping for both SYN and FIN. For SYN flag there is slight growth even after variations in QoS parameters. But in FIN flag there is just fluctuation after implementation of traffic shaping.

4.1.3 Throughput

Throughput (b/s)	TCP Flags ratio in percentage								
	NS	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN
1) 1127006	0.0000	0.0000	0.0000	0.0000	99.4067	13.0437	0.0000	0.1949	0.1413
2) 1121278	0.0000	0.0000	0.0000	0.0000	99.2499	1.8392	0.0000	0.2489	0.1467
3) 1113050	0.0000	0.0000	0.0000	0.0000	99.2625	1.7976	0.0000	0.2773	0.1569
4) 1105740	0.0000	0.0000	0.0000	0.0000	99.2730	1.8621	0.0000	0.3267	0.1428
5) 1106592	0.0000	0.0000	0.0000	0.0000	99.3351	1.7282	0.0000	0.3241	0.1376
6) 1102879	0.0000	0.0000	0.0000	0.0000	99.2874	1.7200	0.0000	0.3305	0.1379
7) 1097639	0.0000	0.0000	0.0000	0.0000	99.3399	1.9523	0.0000	0.3084	0.1301
8) 1090553	0.0000	0.0000	0.0000	0.0000	99.3488	1.9533	0.0000	0.3457	0.1425
9) 1079117	0.0000	0.0000	0.0000	0.0000	99.2876	2.0618	0.0000	0.4485	0.1512

Table. 15. Variations in TCP flags for throughput

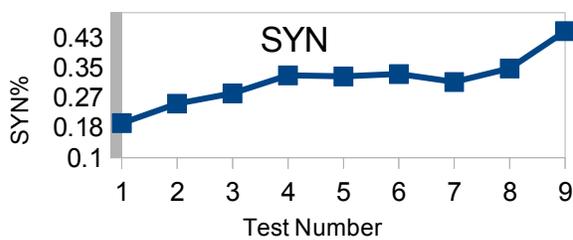


Fig. 16. SYN% relative to throughput (lab)

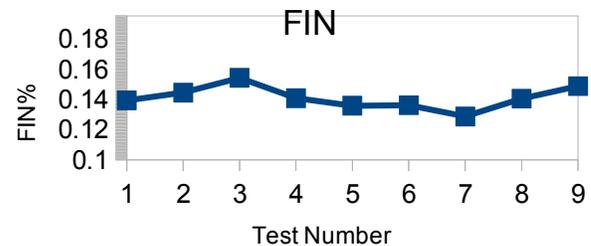


Fig. 17. FIN% relative to throughput (lab)

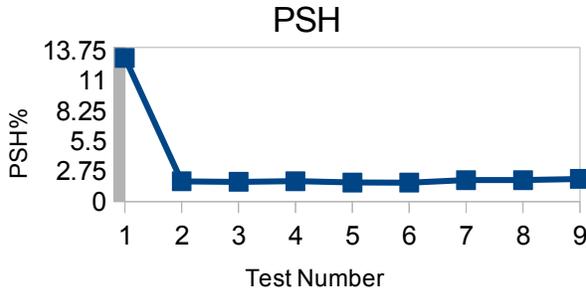


Fig. 18. PSH% relative to throughput (lab)

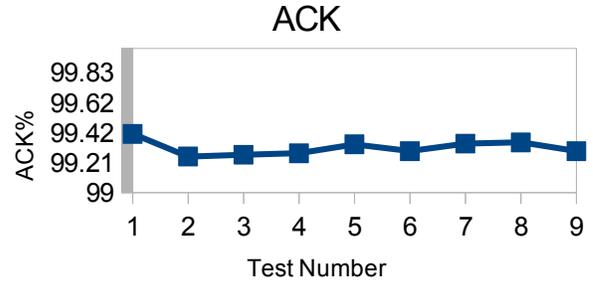


Fig. 19. ACK% relative to throughput (lab)

When there is variations in throughput, similar to packet-loss and delay rapid decrement is resulted for PSH percentage after traffic shaping. For both FIN and ACK, it just wave around constant value and there is no increasing or decreasing trend. And finally the amount packets with SYN flag up growth in parallel to MOS. So according to this observation generally when the traffic shaping is applied the percentage of TCP flags is changed. For packet-loss there is no increasing or decreasing trend after traffic shaping in any of flags. But for both delay and throughput, the graph growth slightly.

4.2 User action

In the last session it has been cleared how TCP behave in different QoS level in terms of percentages of flags. As it mentioned before, in previous part the user actions was always constant in all the experiments. In this session the variations in TCP flags when the real user is involved is analyzed.

4.2.1 Packet-Loss

Packet-loss	TCP Flags ratio in percentage								
	NS	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN
1)0%	0.0000	0.0000	0.0000	0.0000	99.0753	2.6368	0.0000	0.4327	0.3198
2)7%	0.0000	0.0000	0.0000	0.0000	99.5698	5.8635	0.0000	0.8492	0.5924
3)8%	0.0000	0.0000	0.0000	0.0000	99.6414	5.1804	0.0000	0.6168	0.4287
4)9%	0.0000	0.0000	0.0000	0.0000	99.4719	5.6739	0.0000	0.5875	0.3721
5)10%	0.0000	0.0000	0.0000	0.0000	99.6267	6.2944	0.0000	0.5097	0.3530
6)11%	0.0000	0.0000	0.0000	0.0000	99.4795	6.1377	0.0000	0.6654	0.4268
7)12%	0.0000	0.0000	0.0000	0.0000	99.7597	6.6041	0.0000	0.3022	0.2480
8)13%	0.0000	0.0002	0.0002	0.0000	99.8750	6.9331	0.0000	0.1893	0.1626
9)14%	0.0000	0.0000	0.0000	0.0000	99.8801	6.5131	0.0000	0.1877	0.1542

Table. 16. Variations in TCP flags for packet-loss (Real scenario)

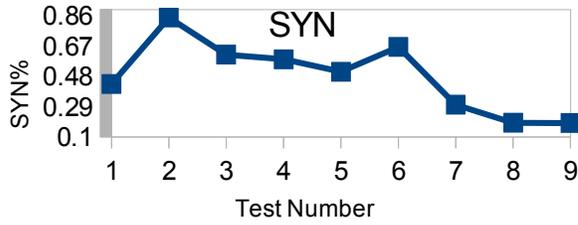


Fig. 20. SYN% relative to packet-loss (real)

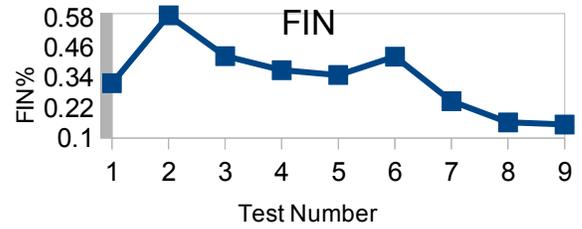


Fig. 21. FIN% relative to packet-loss (real)

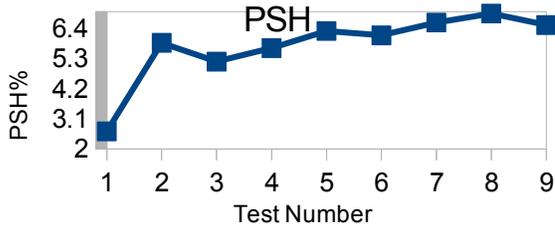


Fig. 22. PSH% relative to packet-loss (real)

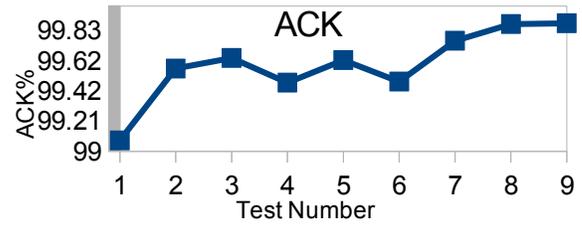


Fig. 23. ACK% relative to packet-loss (real)

When the users are involved, there is a big increment in the percentage of all TCP flags. The trends for SYN and FIN flags decrease rapidly from second experiments which is start of traffic shaping. But in PSH and ACK it increase gradually.

4.2.2 Delay

Delay (ms)	TCP Flags ratio in percentage								
	NS	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN
1) 0	0.0000	0.0000	0.0000	0.0000	99.0753	2.6368	0.0000	0.4327	0.3198
2) 500	0.0000	0.0000	0.0000	0.0000	97.3218	2.4311	0.0000	0.5142	0.4516
3) 600	0.0000	0.0000	0.0000	0.0000	96.1821	2.8450	0.0000	0.9877	0.6313
4) 700	0.0000	0.0000	0.0000	0.0000	97.9922	2.5584	0.0000	0.6647	0.4542
5) 800	0.0000	0.0000	0.0000	0.0000	97.3657	2.5126	0.0000	0.6654	0.4406
6) 900	0.0000	0.0000	0.0000	0.0000	97.9433	2.3970	0.0000	0.7287	0.4541
7) 1000	0.0000	0.0000	0.0000	0.0000	97.5892	3.0727	0.0000	1.3200	0.5309
8) 1100	0.0000	0.0000	0.0000	0.0000	98.0330	2.3534	0.0000	1.0188	0.3911
9) 1200	0.0000	0.0000	0.0000	0.0000	96.7829	2.0560	0.0000	1.0601	0.4001

Table. 17. Variations in TCP flags for delay (Real scenario)

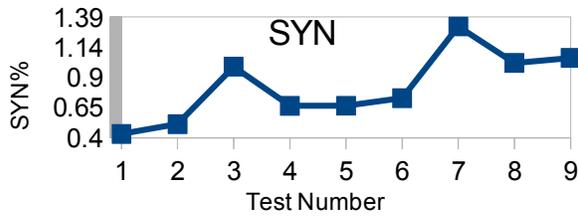


Fig. 24. SYN% relative to delay (real)

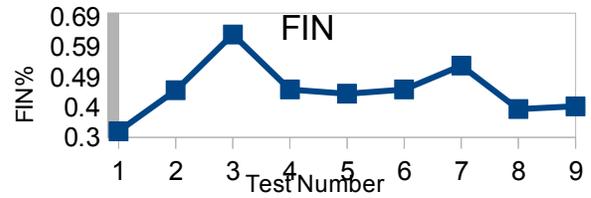


Fig. 25. FIN% relative to delay (real)

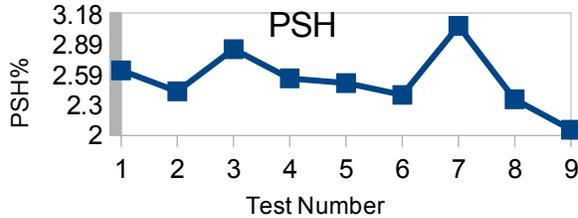


Fig. 26. PSH% relative to delay (real)

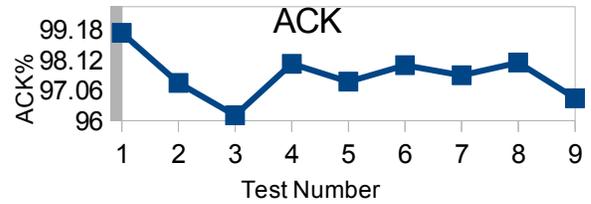


Fig. 27. ACK% relative to delay (real)

The above diagrams show the variations in SYN, FIN and PSH flags has really similar patterns. SYN has a very slight up going trend. For FIN flag almost the beginning and end of trend are same. And PSH decrease slowly when delay increase. All of these diagrams has two pick in third and seventh experiments. Also in all of them the first pick is smaller than second one. But the way of variations for the percentages of packets with acknowledgement flag up is totally different. The amount of this flag after second experiment which is start of traffic shaping is mostly constant. There is big drop in the beginning of the trend from first up to the end of third test. Then there is fluctuation around 97% in the middle of diagram. Finally again it move down at the end of the graph.

4.2.3 Throughput

Throughput(b/s)	TCP Flags ratio in percentage								
	NS	CWR	ECE	URG	ACK	PSH	RST	SYN	FIN
1) 1127006	0.0000	0.0000	0.0000	0.0000	99.0753	2.6368	0.0000	0.4327	0.3198
2) 1126314	0.0000	0.0000	0.0000	0.0000	98.5724	2.7605	0.0000	0.4209	0.2610
3) 1124133	0.0000	0.0000	0.0000	0.0000	98.7109	2.8980	0.0000	0.5209	0.3240
4) 1123211	0.0000	0.0000	0.0000	0.0000	98.8060	2.3246	0.0000	0.5745	0.2979
5) 1122901	0.0000	0.0000	0.0000	0.0000	99.0448	2.4367	0.0000	0.6235	0.3395
6) 1122500	0.0000	0.0000	0.0000	0.0000	99.1422	2.2684	0.0000	0.6238	0.3115
7) 1121278	0.0000	0.0000	0.0000	0.0000	98.9617	2.4218	0.0000	0.7257	0.3454
8) 1113050	0.0000	0.0000	0.0000	0.0000	99.3663	2.3444	0.0000	0.5531	0.1917
9) 1105740	0.0000	0.0000	0.0000	0.0000	99.1918	2.4005	0.0000	1.0007	0.4160

Table. 18. Variations in TCP flags for throughput (Real scenario)

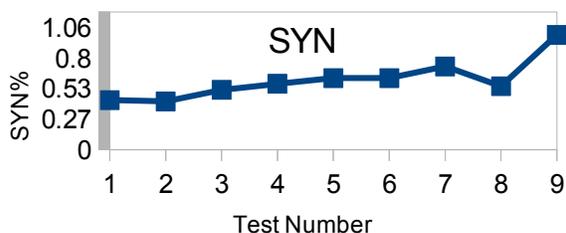


Fig. 28. SYN% relative to throughput (real)

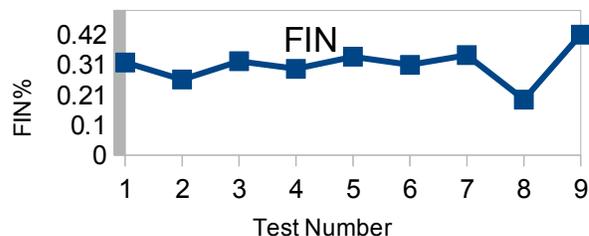


Fig. 29. FIN% relative to throughput (real)

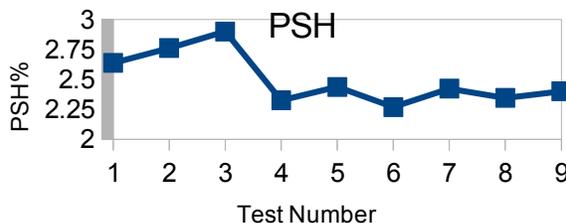


Fig. 30. PSH% relative to throughput (real)

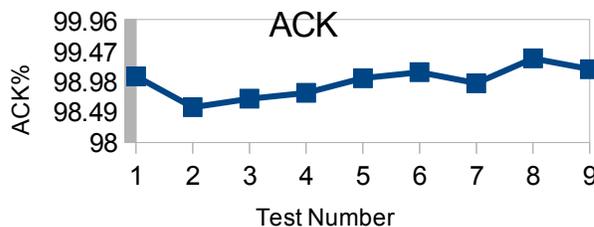


Fig. 31. ACK% relative to throughput (real)

The variations in throughput is resulted to really slight increment in SYN and FIN flags. However in both of them there is big drop in 8th experiment. In ACK it decrease after traffic shaping but but very gradually it increase up to 8th experiment. The percentage of packets with PSH flag up growth at the beginning, then there is a big down going trend, and finally it vary around constant value up to the end of experience. So it is concluded that in this case only for FIN and SYN flags there is clear pattern.

4.3 QoE

In the last session the traffic of research is analyzed. At same time while the traffic was monitoring, users are asked to score the web service. In this part the trends for variations in user satisfaction will be discussed. Each of the QoS parameters has eight different stages. And also there is one experiment without traffic shaping. So totally there are 25 stages that users are asked to score. 4.84 is the average of users who participated in each level. The smallest users group for one stage is four and biggest one is 6. So totally 121 times the web service is evaluated by users. However there are some users who participated in the survey more than one time. If each user who participated in the survey is counted one time, then it will be reduced to 108. On the other words 108 users participated in the survey.

The web site is designed for video sharing. There are lots of published videos, but only a few of them are used for survey. The videos that are used has the same QoS and also same size. Both web site and mobile app are available for everyone. To be sure that the scores are valid, the remote user evaluations are discarded. To make it more clear, all the survey is done manually and users are asked to score in same situation.

The main parameter that can effect the results of the survey is network performance. To limit it's effect for both mobile app and web browser, Eduroam is used for all the grades. So for all the experiments this parameter is constant. However also there is possibilities that the network performance of Eduroam vary in different days or different times of day. For each of the QoS parameters eight stages is considered. Each stage is scored by 5 users. So even if one of or two of the users are effected by network performance more than QoS parameters of web-server, still the MOS is valid. And in worst case if all the users in one stage are more effected by network performance, it has not big effect in whole trend, because it is result of 8 stages. It is almost impossible to avoid completely the effect of network performance on MOS. The research is in the real situation so it is not expected that MOS exactly change according to variations in web server QoS. But it is expected MOS trend generally vary

based on changes in QoS. For example when there is 10% packet-loss, MOS is higher than 9%. So maybe it is because of network performance or any other parameter which can not be controlled by web server. But generally when the packet-loss increased from 7% to 14%, the MOS is decreased from around 9 to around 3. So it can be concluded that variations in web-server QoS parameters could over come on the effect of other parameters like network performance.

Another consideration in the survey is different user expectations. For this case all the users are chosen from BTH students and they asked to score in same places.

The last main consideration of this survey is effect of hardware and software on user satisfaction. For this case all the experiments are done with similar hardware and software. Also while it is a web server, each time the user is asked to score, all the history of web browser is cleared. The web based mobile application is used in the survey. While it is not possible to clear the history for mobile application, the link of mobile app is used in web browser of the mobile phone. So similar to web browser of laptop, the web browser of mobile phone history is cleared each time. Sony Vaio laptop and Samsung galaxy s4 are used for the survey. Firefox web browser for laptop and google chrome is used for mobile phone. It should be mentioned there is nothing special about web browsers or hardware specification. The only reason behind these choices is to keep all parameters same for all the users. So we are sure the effect of these parameters is similar to all of them.

4.3.1 Packet-Loss

The main aim of this session is to shows the effect of variations in QoS on QoE. For these propose the graphs with QoS parameters and MOS as the two axes are provided. However in QoS axis instead of the value of each QoS parameter, the test number is used. The reason behind this choice is behind to the way of variations in QoS parameters. In all the tests, the first experiment is without any traffic shaping. So there is big difference between first test and second one. For example in case of packet-loss, it jumps from 0% to 7%. Moreover for throughput there is no absolute unit for variations. Considering the limitations for traffic shaping it was not possible to shape it with high accuracy. So it just generally decrease. To exemplify the case 1126314 (b/s), 1124133 (b/s) and 1123211(b/s) are respectively the throughput for second, thirds and fourth experiments. As it can be seen the difference between second and third experiments is not equal to third and fourth one. On the other words there is no constant unit in terms of variations in throughput. Considering these two reasons, for this session instead of value of QoS parameters, test numbers are used. So the way of variations in QoE by increasing the test number are resulted. In the next session (7.4 QoS=function(QoE)), the real values are used to find the correlation between QoS and QoE.

Packet-Loss	1) 0%	2) 7%	3) 8%	4) 9%	5)10%	6)11%	7)12%	8)13%	9)14%
MOS (Laptop)	9.8	9	8	6.8	7.2	4.8	4.2	3.4	2.8

Table. 19. Variations in MOS relative to packet-loss (laptop)

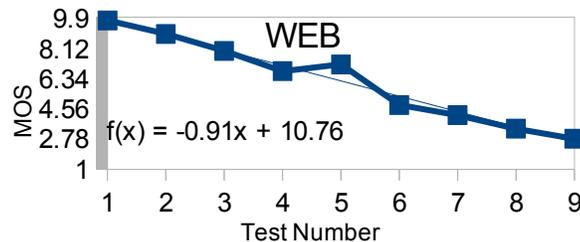


Fig. 32. MOS relative to packet loss (web)

The above diagram illustrates by increasing in the percentage of packet-loss, the user satisfaction level is decreased. However in the fifth experiment there is a small increment.

Packet-Loss	1) 0%	2) 7%	3) 8%	4) 9%	5)10%	6)11%	7)12%	8)13%	9)14%
MOS (Cell phone)	9.8	8.6	7.2	5.4	7	4.6	3.2	3.8	3

Table. 20. Variations in MOS relative to packet-loss (Cellphone)

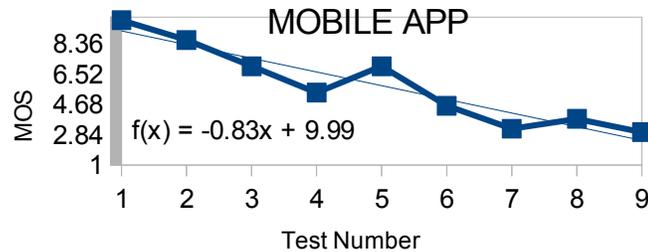


Fig. 33. MOS relative to packet loss (Cell phone)

The trend for cell phone is almost similar to laptop. There is inverse ratio between the amount of packet-loss and MOS. Also in cell phone there is increment in fifth experiment. The pick in fifth experiment for cell phone is bigger relative to web browser.

So in both mobile phone and laptop, the percentage of packet-loss has inverse ratio with user satisfaction. The scores for both of them are almost similar. Therefore it is concluded that packet-loss has same impact on user satisfaction for both of the access points.

4.3.2 Delay

Delay (ms)	1) 0	2)500	3)600	4)700	5)800	6)900	7)1000	8)1100	9)1200
MOS (laptop)	9.8	8.6	7.25	7.6	6.8	7	5.5	4.75	3

Table. 21. Variations in MOS relative to delay (laptop)

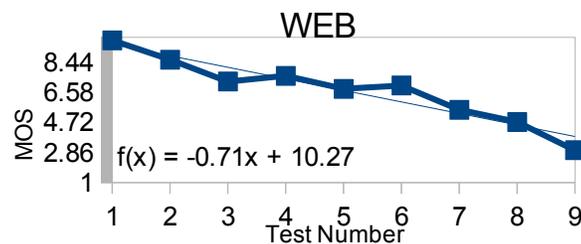


Fig. 34. MOS relative to delay (web)

This graph shows delay and MOS has clear reverse ratio. On the other words increment in delay is resulted to decrement in user satisfaction. For the first three test that delay is increased from 0 to 600 ms, the MOS is decreased from 9.8 to 7.25. Then in the next three experiments it always fluctuate around 7. And again in the last three tests, the trend is completely decreasing.

Delay (ms)	1) 0	2)500	3)600	4)700	5)800	6)900	7)1000	8)1100	9)1200
MOS (Cell phone)	9.8	9.6	7.5	6.6	4.6	6.4	1.5	1.25	2.5

Table. 22. Variations in MOS relative to delay (Cellphone)

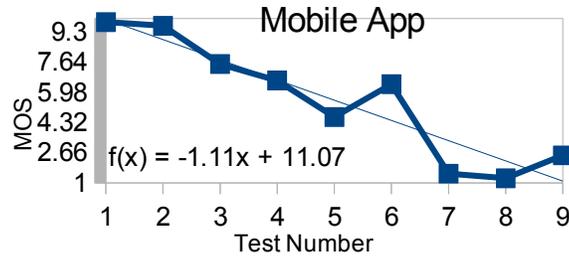


Fig. 35. MOS relative to delay (Cell phone)

Similar to laptop, in cell phone also the user satisfaction decrease in parallel with increment in delay. The variations in MOS shows that users are much more sensitive about big delays in cell phone rather than laptop. As it can be seen in the first two tests, there is no big difference in MOS. Then from second observation to the fifth one it decrease slightly. There is a big pick for six experiment, suddenly it growth from 4.6 to 6.4. Again the trend come back to its decreasing pattern for 1000 ms and 1100 ms delays. And finally there is small increment at the end of the diagram.

So from a comparison between cell phone and laptop it is concluded that up to 900 ms the MOS for both of the is almost similar. But after 900 ms, the delay has much more negative impact on user satisfaction for cell phone users relative to laptop users.

4.3.3 Throughput

Throughput (b/s)	1) 1127006	2) 1126314	3) 1124133	4) 1123211	5) 1122901	6) 1122500	7) 1121278	8) 1113050	9) 1105740
MOS (Laptop)	9.8	8.6	7.8	6.6	6	6.8	5.83	3	2.8

Table. 23. Variations in MOS relative to throughput (laptop)

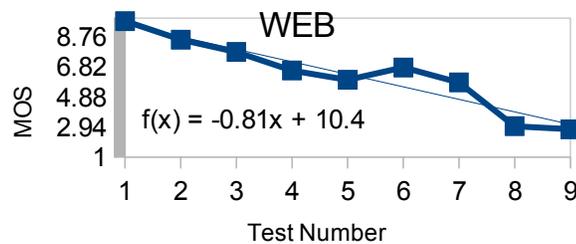


Fig. 36. MOS relative to throughput (web)

The graph for variations in MOS for laptop shows the amount of user satisfaction decrease in parallel to decrement in throughput. From first test up to test number five there is significant drop. Then from fifth experiment up to seven a slight growth can be see. And Finally again the end of graph follow the general pattern which is decreasing in MOS.

Throughput (b/s)	1) 1127006	2) 1126314	3) 1124133	4) 1123211	5) 1122901	6) 1122500	7) 1121278	8) 1113050	9) 1105740
MOS (Cell phone)	9.8	8.6	6	4	4.6	5	3.5	3.2	2.2

Table. 24. Variations in MOS relative to throughput (Cellphone)

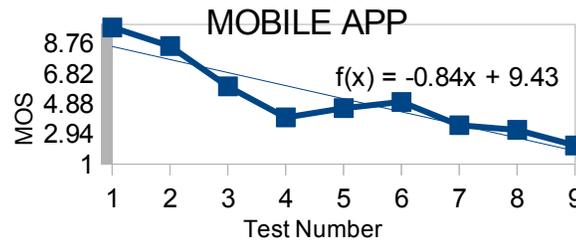


Fig. 37. MOS relative to throughput (Cell phone)

The same as laptop, the trend of cell phone is also decreasing. From the first test up to the end of fourth experiment there is a big drop. Then it growth very slightly at the middle of trend. And finally a slight increment up to the end of tests.

According to the above diagrams throughput has more effect on cell phone users. As it can be seen in the first four test the MOS in laptop is decreased from 9.8 to 6. But in the same part for cell phone it changed from 9.8 to 4.6. So the amount of decrement is bigger for cell phone. Also the increasing trend in the middle of cell phone diagram is more slight rather than laptop. So generally it is concluded that throughput is more essential for cell phone relative to laptop.

4.4 Correlation between QoE and QoS

The diagrams in the last session (for all the three QoS parameters) are proposed only to show the effect of variations in QoS on QoE. However as it mentioned in design and implementation, finding the definition of function in $QoS = \text{function}(QoE)$, is one of the main aims of the research. To achieve this target the horizontal axis is changed to QoE instead of test number. And vertical axis is changed to value of QoS parameters. All the values in the graphs are sorted according to MOS values. So the MOS axis is always increasing. By the usage of these method the new graphs for the equation of $QoS = \text{function}(QoE)$ are provided.

4.4.1 Packet-loss

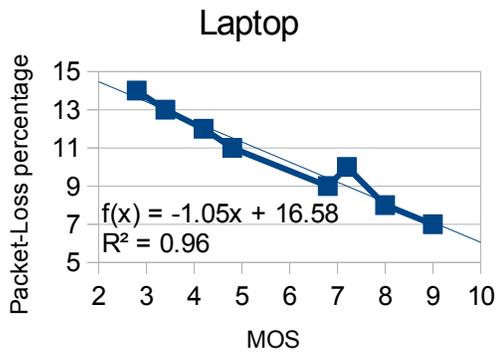


Fig38. Packet-loss=QoS=function(QoE)
For laptop

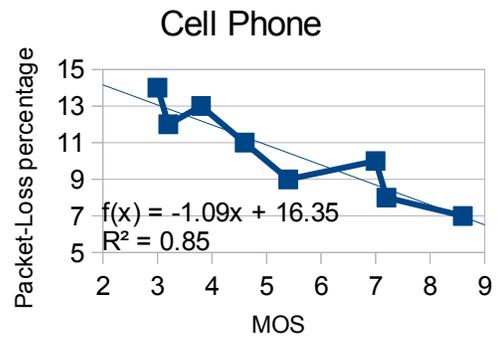


Fig39. Packet-loss=QoS=function(QoE)
For Cell phone

For packet-loss, $f(x) = -1.05x + 16.58$ and $f(x) = -1.09x + 16.35$ are the two trend line equations for laptop and mobile phone respectively. Clearly in these two formulas, $f(x)$ is packet-loss percentage and x is MOS.

4.4.2 Delay

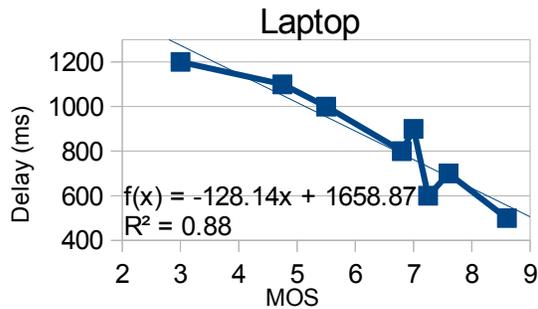


Fig40. Delay=QoS=function(QoE)
For laptop

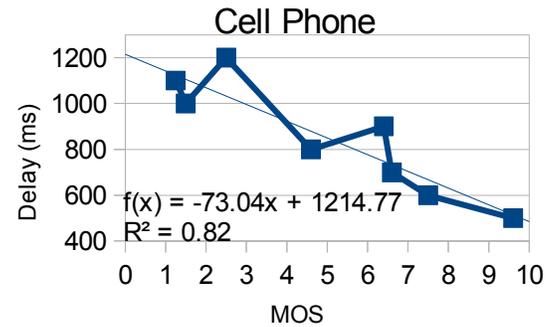


Fig41. Delay=QoS=function(QoE)
For cell phone

The trend lines for laptop and mobile phone has bigger difference in delay. The slope for laptop is almost twice bigger that cell phone. Similar to packet-loss, increasing in delay is resulted to decrement in the user satisfaction.

4.4.3 Throughput

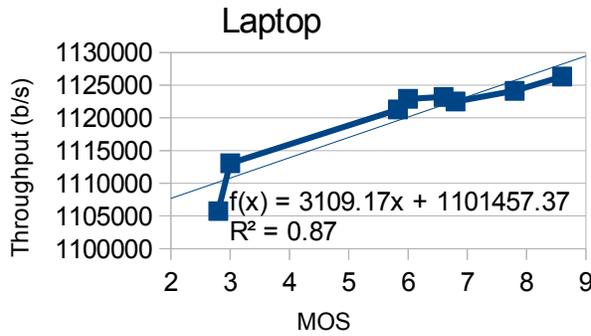


Fig.42. Throughput=QoS=function(QoE)
For laptop

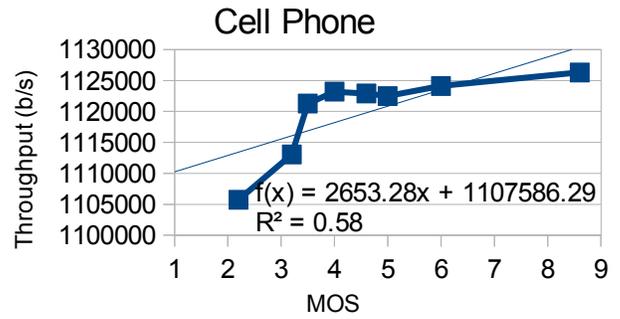


Fig.43. Throughput=QoS=function(QoE)
For cell phone

As it was expected, throughput has direct ratio with MOS. So the slopes for both laptop and mobile phone are positive.

The coefficient of determinations both for laptop and mobile phone in throughput are smaller than delay and packet-loss. Between delay and packet-loss, packet-loss has bigger coefficient of determination for both of the access points. Therefore it is concluded that the trend for packet-loss is the most reliable. Delay is in middle in terms of reliability of pattern and finally throughput is the last one.

5 ANALYSIS

In session 8 (Data analysis) it has been shown how TCP flags vary in different QoS. And in the ninth part (QoE) the user satisfaction level in different QoS is researched. So it is observed how user satisfaction level and TCP packets change by variations in QoS. In both of them there is a common dimension called QoS. For conclusion the common dimension (QoS) will be factored. So it will be shown how TCP packets are changed according to user satisfaction. All the graphs in this part has two axes. The horizontal axis is always MOS an vertical axis is one of the TCP flags. The MOS is sorted, so it is easy to see how tcp flags percentages vary in parallel with increment in MOS.

5.1 Packet-loss

5.1.1 Web Browser

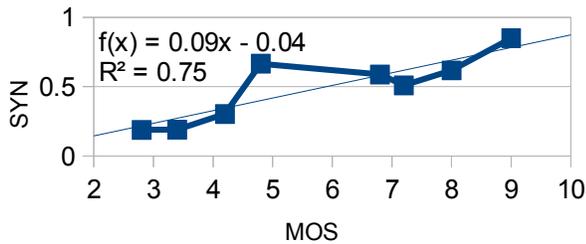


Fig. 44. SYN relative to MOS (Laptop)

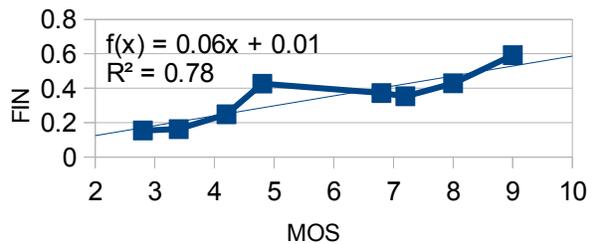


Fig. 45. FIN relative to MOS (Laptop)

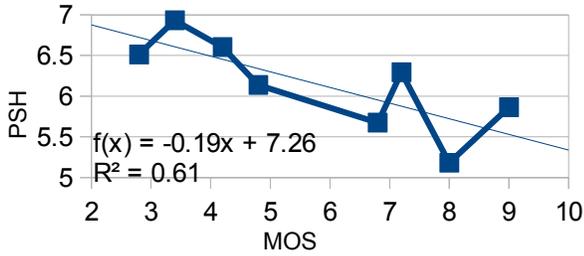


Fig. 46. PSH relative to MOS (Laptop)

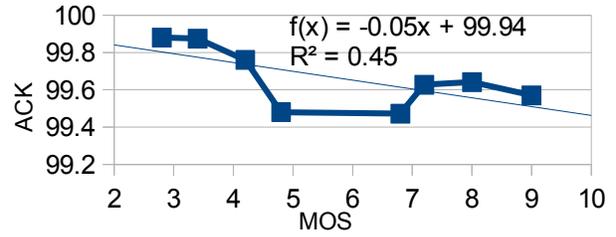


Fig. 47. ACK relative to MOS (Laptop)

Based on the above observation it is concluded that the patterns for SYN and FIN are almost similar. In both of them when the average of users score increase from 2.5 to 5, the trend is sharply increasing. In the middle of these two diagrams the parentages are almost constant. And finally at the end of graph it increase slightly. The variations of PSH has not a really clear pattern. However the first point has higher percentage than last last one. There are three picks in the graph which prove this trend can not be considered as real pattern for user satisfaction indicator. The acknowledgement diagram has generally decreasing trend. There is a big decrement at the beginning of the diagram. Then in the middle it keep constant around 99.4. At the end of graph firs there is small growth then, again it follow down going trend of the diagram.

5.1.2 Mobile Application

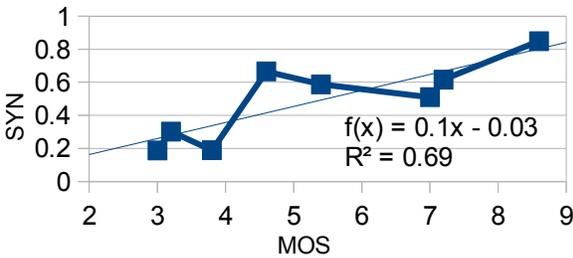


Fig. 48. SYN relative to MOS (Cell Phone)

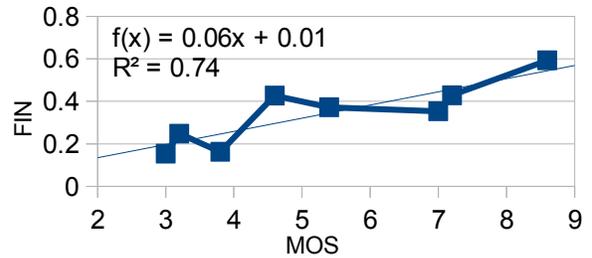


Fig. 49. FIN relative to MOS (Cell Phone)

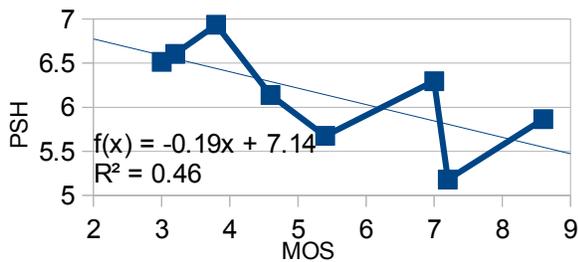


Fig. 50. PSH relative to MOS (Cell Phone)

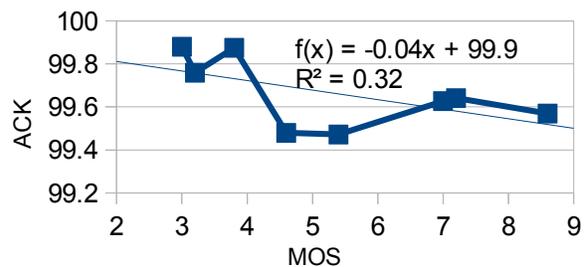


Fig. 51. ACK relative to MOS (Cell Phone)

By replacing the MOS for cell phone users instead of laptop, the graphs for acknowledgement flags has changed more than others. The same as laptop in the graphs for mobile phone, SYN and FIN has more clear up going trend. For PSH and ACK generally the diagrams has decreasing trends. For both of the SYN and FIN flags the first three points are fluctuating around 0.2 percents. There is a sharp increasing from third to fourth stage. Both of the diagrams has really slightly decrement in the middle and soft growth at the end.

The variations in PSH shows that the first point has higher percentage than last one. But similar to laptop diagram, there are three picks in the graph. First it increase during the first three experiments. Then there is rapid decrement up to middle of graph. Again there is one big pick which has around one percent different with the last one. And finally the last part of graph is combination of two points. First on is in the decreasing direction and the second on is big growth.

The ACK flag is the one which has different trend with laptop version. At the beginning it fluctuate around 99.8 percent. Then there is rapid decreasing trend from third point to fourth one. And the last part it has a really slight increasing trend.

To sum it up for both of the laptop and mobile phone, all the trends has similar up going or down going trends. However the way of variations in ACK has more differences in case of mobile phone and laptop relative to other flags.

5.2 Delay

5.2.1 Web Browser

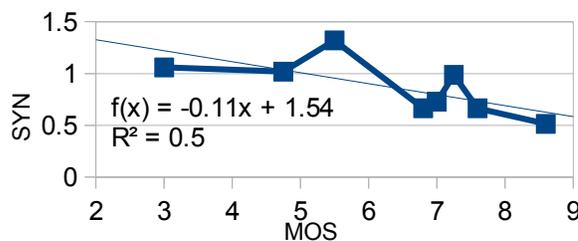


Fig. 52. SYN relative to MOS (Laptop)

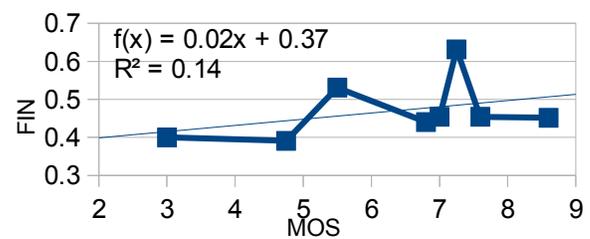


Fig. 53. FIN relative to MOS (Laptop)

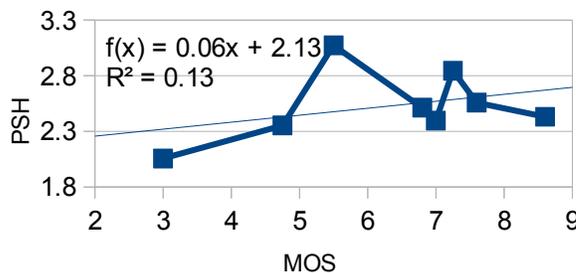


Fig. 54. PSH relative to MOS (Laptop)

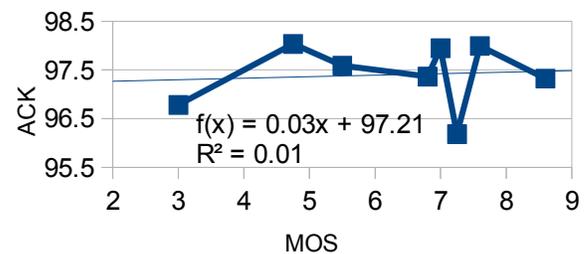


Fig. 55. ACK relative to MOS (Laptop)

In the four stages of delay, the MOS is always near seven. So it is resulted that there is congestion of values around

it. In the rest of experiments the distribution of scores is mostly fair.

The patterns for SYN, FIN and PSH are similar to each other in case of having two sharp picks. But FIN and PSH increase very slowly and there is slight decreasing in SYN. The way of variations for ACK is totally different with others.

At the beginning of SYN the percentage of the flag is constant one. Then there is a big pick when MOS is around 5.5. The graph follow decreasing pattern in the middle. And finally there is one big growth at the end of the trend.

The diagrams for FIN and PSH has generally up going pattern. In both of them there is sharp pick at third and sixth points. The patterns are very similar, however the main difference is in the ranges. In FIN it vary between 0.4 to around 0.6 percents. But the PSH variations is between 2 to 3 percents.

As it mentioned before ACK has not up going or down going trend. It always waves around 97 percents. The most significant change in the pattern is in the end of trend. at sixth observation there is big decrement.

5.2.2 Mobile APP

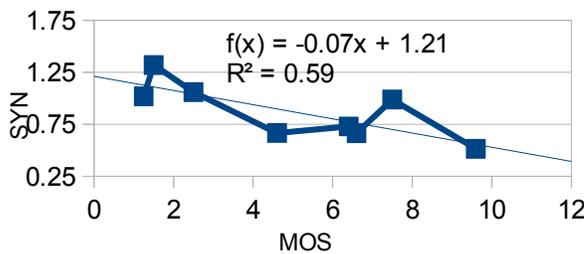


Fig. 56. SYN relative to MOS (Cell phone)

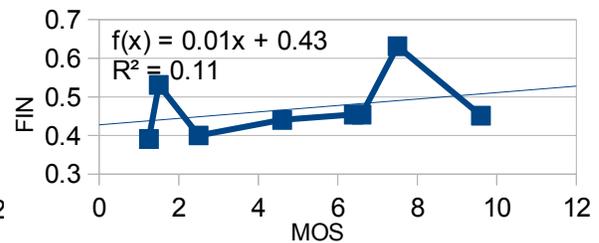


Fig. 57. FIN relative to MOS (Cell phone)

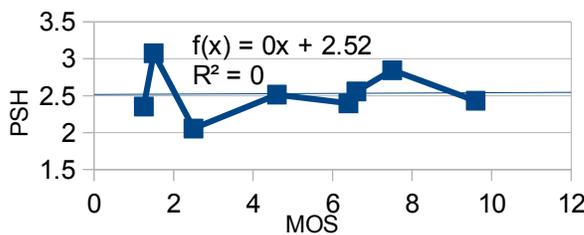


Fig. 58. PSH relative to MOS (Cell phone)

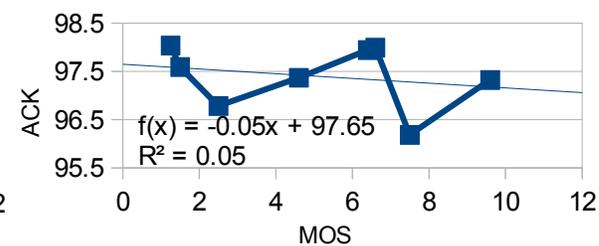


Fig. 59. ACK relative to MOS (Cell phone)

The distribution of scores is more fair for mobile phone users relative to laptop users. However there are two scores 6.4 and 6.6 that are almost equal.

The SYN diagram has generally decreasing trend. At the beginning there is a small pick. Then it goes down gradually from second point to fourth one. It just wave around constant value in the middle of trend. Then there is small growth and finally again it comes back to the down going trend.

The trends of SYN and PSH are almost similar. In both of them the first point and last point has not significant differences. So they can not be considered as increasing or decreasing graphs. FIN diagram has two clear picks while PSH has only one. However in both of them very slowly growth from beginning to the end can be seen.

The ACK pattern is the most unpredictable one. First it start with very sharp regressive trend. Then it growth significantly. Again a rally big drop can be seen in the seventh point of the graph. And Finally it growth sharply at the end of graph.

A comparison between mobile phone graphs and laptop diagrams, shows that generally they have same trends. The SYN has more significant down going trend in mobile phone. The amount of increment in FIN and PSH are same in mobile phone and laptop. And finally ACK in both of them has really unpredictable trend.

5.3 Throughput

5.3.1 Web Browser

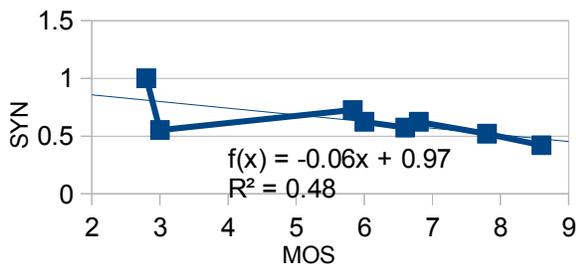


Fig. 60. SYN relative to MOS (Laptop)

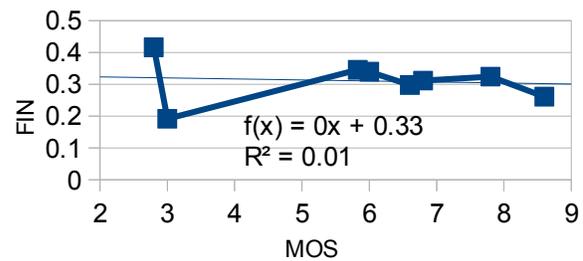


Fig. 61. FIN relative to MOS (Laptop)

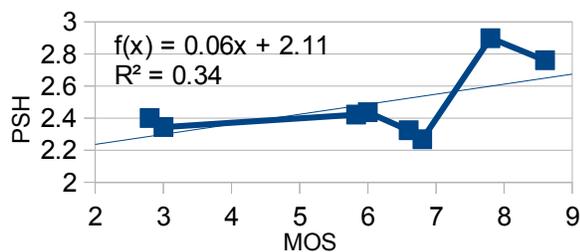


Fig. 62. PSH relative to MOS (Laptop)

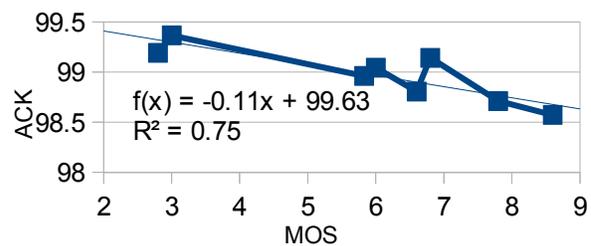


Fig. 63. ACK relative to MOS (Laptop)

The decreasing in throughput is not resulted to liner down going trend for MOS in laptop users. There fore suddenly it increased from three to six. This rapid increment is resulted that the percentage of TCP flags when MOS is three looks more effective than others. However in the conclusion this space should not effect the final patterns.

Comparison of the four graphs shows that the relation of variations in SYN and FIN flags with user satisfaction is almost similar. In both of them first there is really sharp decrement. Then it increase very slightly. And finally both of the trends has gradual decreasing trend. The two flags has down going trend.

PSH flag start point has smaller value relative to its end. When the MOS vary from 2.5 to 6, it is always around 2.4 percent. Then there is a small drop in the middle of graph. Finally it increase rapidly from around 2.2 percent to 2.8 percent. At the end of diagram again there is really small down going trend.

The ACK flag has really clear slight down going trend. There are three really small picks in the graph. But while

they are too small it really looks like a liner decreasing diagram. When the MOS is minimum 99.19 percents of packets has acknowledgement flag up. And in the end of graph when the MOS reach 8.6, the percentage of ACK flags decrease to 98.57.

5.3.2 Mobile APP

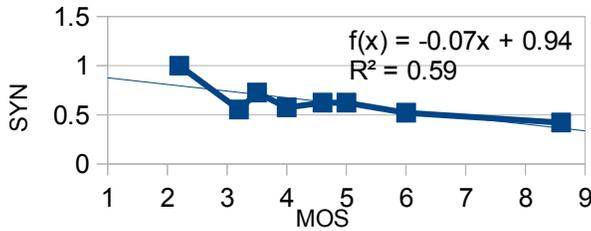


Fig. 64. SYN relative to MOS (Cell Phone)

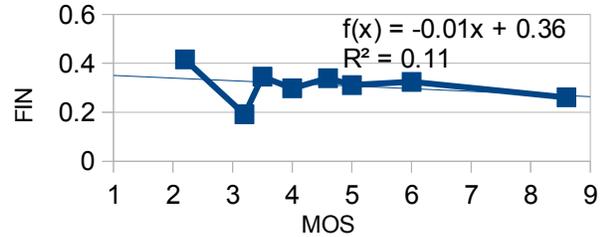


Fig. 65. FIN relative to MOS (Cell Phone)

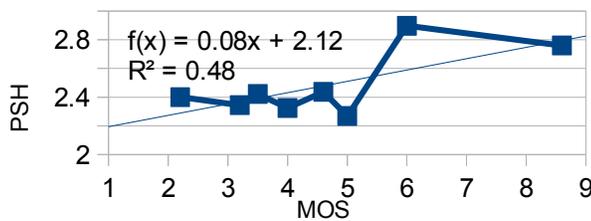


Fig. 66. PSH relative to MOS (Cell Phone)

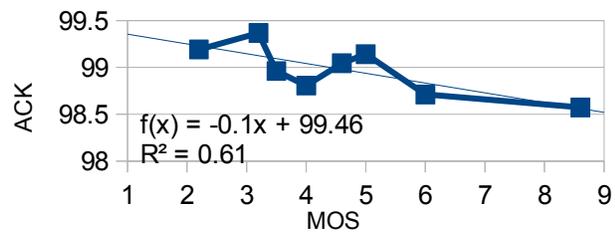


Fig. 67. ACK relative to MOS (Cell Phone)

By changing the MOS values for mobile phone users to laptop users, logically the trends are not changed. However the way of distribution of averages of scores for mobile phone has one clear difference. In laptop the value of MOS has jumped from tree to sex, but in mobile phone it is from six to nine. So this time there is long gap at the end of all trends instead of middle.

Similar to laptop in mobile phone also SYN, FIN and ACK has decreasing trend and PSH increasing. Again SYN and FIN has very similar pattern. In both of them first there is big drop and then they go down gradually.

For PSH in the first fifth experiments it always fluctuate around 2.4 percent. Then there is really rapid increment when MOS goes from five to six. And finally it drop very slightly.

And finally the ACK flag has a slight decreasing patterns with two picks. The first pick happened when MOS is three. In this point the ACK percentage increase from 98.71 to 99.14 percents. The second pick which has almost similar difference is in the fifth experiment.

As conclusion it has been observed that SYN, FIN and ACK flags has decreasing trend and PSH increase in parallel with MOS.

5.4 Discussion

Final discussion of the research is divided in to two sides. In the first part the distribution of users scores for mobile phone and laptop is analyzed. The second side of the discussion is focused on the correlation of variations in the percentages of TCP flags with user satisfaction.

According to the outputs of QoE session, the changes in the QoS parameters has really clear effect on user satisfaction. Growth in amount of packet-loss and delay is resulted to decrement in average of scores. And mean

opinion scores are getting smaller in parallel with decreasing in throughput. The trend of MOS for mobile phone and laptop are very similar in case of packet-loss. For throughput the mobile phone users are a little more sensitive. Specially after third experiment the mobile phone users scored smaller than laptop users. The most significant difference between the two graphs is belongs to delay. At the beginning for 500ms and 600ms, the mobile phone users scored higher than laptop users. Then in the middle of two trends MOS for mobile phone is a little smaller. At the end of survey of delay (1000ms, 1100ms and 1200ms) the MOS for mobile phone is much more smaller than laptop. The mobile phone users mostly scored one or two from ten, when the delay is bigger than one second. So it is concluded that delay has really essential role in case of user satisfaction for video streaming web services for mobile phone users.

As it mentioned above the other propose of the research is correlating user satisfaction to variations in TCP flags. For this aspect at the same time that users are asked to score the web-server, the traffic is also captured and the percentage of different TCP flags is calculated. So it has been observed how the amount of TCP flags has been changed by the variations in MOS. All the graphs with MOS as horizontal axis and one of the TCP flags as vertical axis are demonstrated in the last session. NS, CWR, ECE, URG and RST are always constant zero percent. So the analyzes are based on SYN, FIN, PSH and ACK. At this point the main consideration is to get sure if the variations are belongs to user actions. For this issue the same process is done, while user actions are kept constant. So those of patterns that after real case study (when users are involved) are similar to constant actions are discarded.

The correlation of the TCP flags and MOS for laptop and mobile phone in case of packet-loss are almost similar. In both of them SYN and FIN has increasing patterns. And PSH and ACK decrease slightly. On the other words the SYN and FIN has positive slope but there is a negative gradient for PSH and ACK in their trend line formulas. However PSH has lots of big up and down going points in the pattern. So it can not be considered as one the QoE indicators. In the experiment that user actions are constant all the trends for packet-loss are fluctuating. So the rest of patterns (SYN, FIN and ACK) are valid from this point of view.

For delay the changes for ACK in laptop and mobile phone are different. Also the patterns of this flag has two big drops in the mobile phone which is resulted to very small coefficient of determination R^2 . So ACK is not considered as an final pattern for delay. SYN and FIN has almost similar patterns for both of devices. Generally SYN is decreasing (negative gradient) and FIN is increasing (positive gradient). However FIN gradient is too small. Specially for mobile phone it is zero. So FIN also should be excluded from patterns for delay. Finally for PSH there is small increment from first pint to last one in both cases. So with same reason like FIN, PSH is not one of the signals for delay problem. In no user patterns for delay, SYN has slight growth and FIN and PSH are always constant. So there is no common pattern in this case. Therefore the variations in SYN is considered as final patterns for delay.

By looking on the patterns of throughput it can be seen that all the graphs are almost same for mobile phone and laptop. SYN, FIN and ACK has down going trends and PSH is increasing. The value of R^2 for PHS for both laptop and mobile phone is around 0.4. But PSH pattern should not be considered as one of the final indicators. Because it has completely different pattern for the first fifth points relative to last three ones. Non of the patterns is similar to no user patterns. There for the rest of flags (SYN, FIN and ACK) should be used in terms of creating final pattern for throughput.

Also from mathematical perspective, in all the graphs the formulas for each of the QoS parameters are mentioned accordingly. While there are two type of formulas for each of the flags (Laptop and mobile phone), a proper confidence interval is considered in final formulas. The confidence intervals are observed so each equation covers both laptop and mobile phone trends.

In terms of packet loss, FIN formula is identical for both laptop and mobile phone. So simply, $f(x) = 0.06x + 0.01$ is assigned as FIN flag mathematical pattern. For SYN the gradient is 0.09 for web browser and 0.1 for cell phone. And y-intercepts are -0.04 and -0.03 respectively. So $f(x) = 0.09x - 0.03$ with 0.01 confidence interval for both slope and y-intercept is the final equation for SYN. By the usage of same method, $f(x) = -0.04x + 99.92$ with 0.01 confidence interval for slope and 0.02 for y-intercept is the equation of a line in case of ACK.

For delay only one flag passed the criteria to be considered as signal for user dissatisfaction. The difference

between the SYN trend for laptop and mobile phone is bigger than packet-loss.

So logically there is bigger confidence interval. $f(x) = -0.09x + 1.37$ is the final formula for SYN. In this case, 0.02 confidence interval is observed for gradient and 0.17 for y-intercepts.

Similar to packet-loss, in case of throughput SYN, FIN and ACK made the final patterns. In throughput, $f(x) = -0.06x + 0.95$ is the equation of line for SYN trend. This equation has 0.02 confidence interval for both slope and y-intercepts. $f(x) = -0.01x + 0.34$ and $f(x) = -0.11x + 99.54$ are the formulas for FIN and ACK respectively. In both of them 0.01 confidence interval is for slope. But in case of y-intercepts 0.02 for FIN and 0.09 for ACK are the confidence intervals.

6 CONCLUSION AND FUTURE WORK

The first research question was about the evaluation of the user satisfaction, without asking them to score. For this propose the QoE is linked to transport layer. To make it more clear the correlation between the user satisfaction and TCP flags is formulated. In the previous session the proper flags for finding the patterns for each of the QoS parameters are specified. Now for making the final conclusion, look on the all graphs in the previous session. But this time from right to left. On the other words the the way of variations in TCP flag when user satisfaction is decreasing.

From this point of view it is conducted, decrement in SYN and FIN flags and increment in ACK is an alarm for down going user satisfaction. In this situation the problem is belongs to packet-loss. Increasing in the percentage in SYN is also a signal for user dissatisfaction. In this case, the problem is result of delay. And finally if the web-server problem is about throughput then, SYN, FIN and ACK has up going trends. The mathematical analysis for the patterns are widely discussed in the previous chapter.

The second main aim of the research is to find the definition of function in $QoS = \text{function}(QoE)$. This propose is considered as second research question. Regarding to this target the QoE is directly linked to QoS. For this propose, the graphs with MOS as x axis and one of QoS parameters as y axis are provided in session 7.4 (correlation between QoE and QoS). The final equations for both web site users and mobile phone are mentioned in the following table.

QoS parameter	Laptop	Cell phone
$f(x) = \text{Packet-loss percentage (QoX)}$ $x = \text{MOS (QoE)}$	$f(x) = -1.05x + 16.58$ $R^2 = 0.96$	$f(x) = -1.09x + 16.35$ $R^2 = 0.85$
$f(x) = \text{Delay in millisecond (QoX)}$ $x = \text{MOS (QoE)}$	$f(x) = -128.14x + 1658.78$ $R^2 = 0.88$	$f(x) = -73.04x + 1214.77$ $R^2 = 0.82$
$f(x) = \text{Throughput in bit/second (QoX)}$ $x = \text{MOS (QoE)}$	$f(x) = 3109.17x + 1101457.37$ $R^2 = 0.87$	$f(x) = 2653.28x + 1107586.29$ $R^2 = 0.58$

Table. 25. $QoS = \text{function}(QoE)$

Finally the last research question is about priority of the QoS parameters for the web-server. There are three QoS parameters with different units. Percentage for packet-loss, millisecond for delay and bit/second for throughput. It must be emphasized this research is not aimed to find which of the QoS parameters is more important. Because changing the ranges can effect the outputs. For example in this study, packet-loss is varied from 7% to 14% and delay is changed from 500 ms to 1200 ms. And according to these ranges, delay has bigger effect on mobile application users than packet-loss. But what if we change the range for packet-loss from 14% to 20%? Then most likely the effect of packet-loss will be more significant. Therefore in this thesis, the effect of variations in QoS parameters on mobile application users and web browser users are compared. The result of this comparison is in the equation of lines for correlation between QoE and QoS. The amount of packet-loss and delay are gradually increased during the survey. But the throughput has decreasing trend. So all of them are resulted to decrement in the user satisfaction. Therefore the equations for packet-loss and delay has negative slope but it is positive for

throughput. Absolute value of the slopes shows the amount of effect of the QoS parameters on MOS. The ratio of gradient for web browser users relative to mobile application users in case of packet-loss is 0.96. So the impact of packet-loss on mobile phone and laptop is very similar. This ratio for delay and throughput is 1.75 and 1.17 respectively. Therefore it is concluded that throughput has more effect on mobile application users. And in case of delay, it has really bigger impact on cell phone rather than laptop. To sum it up, generally mobile phone users are more sensitive to variations in QoS parameters. It can be really clearly seen in terms of delay.

Considering such big differences between the amount of tolerance for mobile phone users and laptop. It is highly recommended to have two different web-servers for mobile application and web site.

As it mentioned before providing a new method for user satisfaction evaluation, is the main aim of the research. However based on this method it is only possible to test if user is happy. Moreover the reason behind user dissatisfaction is also formulated. For example if the problem belongs to packet-loss or delay.

However still there is no solution provided for web server administration. On the other words the method can only find the problem but has no solution. For instance if the problem is belongs to throughput how web server administrations should react? The answer of these type of questions are considered as future work for this research.

The solution can be based on changes in queuing discipline. It is possible to vary QoS parameters by the usage of variations in queuing discipline. Therefore most likely there should be some possibilities to find the solution based on that as well. To make it more clear, queuing discipline should be used in terms of prioritizing packets according to the QoS parameters. To exemplify the case if the problem is belongs to throughput, all the packets should be delayed by the usage of queuing discipline.

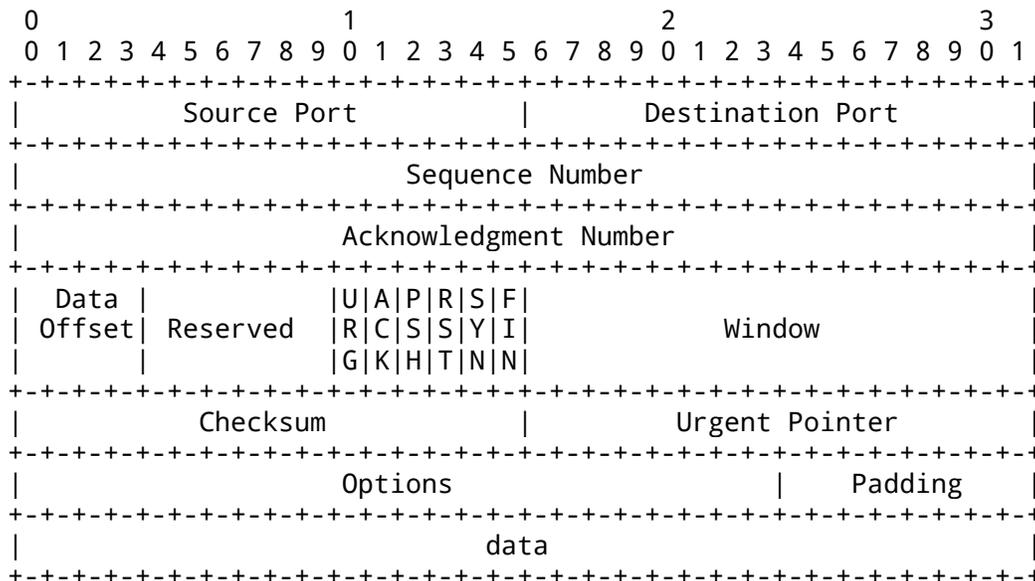
Another research that can be considered as a future work for this research is finding the technical reason behind the final patterns. In this research it is proved that how TCP flags vary relative to user satisfaction. But the technical analysis is missed. For example PSH might be initiated by the application to force delivery, or some other reasons behind variations in TCP flags. This analysis can be used as more filters for correlating user satisfaction to TCP flags. On the other words it can improve the accuracy of the final patterns.

APPENDIX A

The appendix is the TCP header format which is completely copied from the RFC793

“TCP header format

TCP segments are sent as internet datagrams. The Internet Protocol header carries several information fields, including the source and destination host addresses [2]. A TCP header follows the internet header, supplying information specific to the TCP protocol. This division allows for the existence of host level protocols other than TCP.



TCP Header Format

Note that one tick mark represents one bit position.

Source Port: 16 bits

The source port number.

Destination Port: 16 bits

The destination port number.

Sequence Number: 32 bits

The sequence number of the first data octet in this segment (except when SYN is present). If SYN is present the sequence number is the initial sequence number (ISN) and the first data octet is ISN+1.

Acknowledgment Number: 32 bits

If the ACK control bit is set this field contains the value of the next sequence number the sender of the segment is expecting to receive. Once a connection is established this is always sent.

Data Offset: 4 bits

The number of 32 bit words in the TCP Header. This indicates where the data begins. The TCP header (even one including options) is an integral number of 32 bits long.

Reserved: 6 bits

Reserved for future use. Must be zero.

Control Bits: 6 bits (from left to right):

URG: Urgent Pointer field significant
ACK: Acknowledgment field significant
PSH: Push Function
RST: Reset the connection
SYN: Synchronize sequence numbers
FIN: No more data from sender

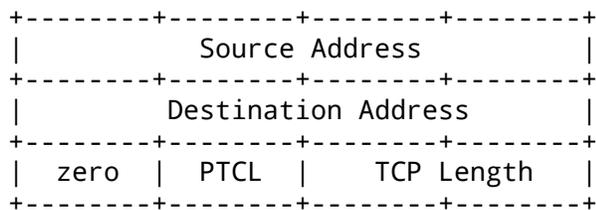
Window: 16 bits

The number of data octets beginning with the one indicated in the acknowledgment field which the sender of this segment is willing to accept.

Checksum: 16 bits

The checksum field is the 16 bit one's complement of the one's complement sum of all 16 bit words in the header and text. If a segment contains an odd number of header and text octets to be checksummed, the last octet is padded on the right with zeros to form a 16 bit word for checksum purposes. The pad is not transmitted as part of the segment. While computing the checksum, the checksum field itself is replaced with zeros.

The checksum also covers a 96 bit pseudo header conceptually prefixed to the TCP header. This pseudo header contains the Source Address, the Destination Address, the Protocol, and TCP length. This gives the TCP protection against misrouted segments. This information is carried in the Internet Protocol and is transferred across the TCP/Network interface in the arguments or results of calls by the TCP on the IP.



The TCP Length is the TCP header length plus the data length in octets (this is not an explicitly transmitted quantity, but is computed), and it does not count the 12 octets of the pseudo header.

Urgent Pointer: 16 bits

This field communicates the current value of the urgent pointer as a positive offset from the sequence number in this segment. The urgent pointer points to the sequence number of the octet following the urgent data. This field is only be interpreted in segments with the URG control bit set.”[21]

REFERENCES

- [1] Webopedia, "QoE." [Online]. Available: <http://www.webopedia.com/TERM/Q/QoE.html>. (The link is tested up to 14 June 2015)
- [2] Florence, A; Antonio, L "QoE-aware QoS management" MoMM '08 Proceedings of the 6th International Conference on Advances in Mobile Computing and Multimedia, 2008, ISBN 978-1-60558-269-6, pp. 111 – 116
- [3] Shirazipour, M; Charlot, G; Lefebvre, G; Krishnan, S; Pierre, S "ConEx Based QoE Feedback to Enhance QoS" ; Proceedings of the 2012 ACM workshop on capacity sharing, 12/2012, CSWS '12, ISBN 1450317804, pp. 27 – 32
- [4] Machado, V. A; Silva, C. N; Oliveira, R. S; Melo, A. M; Silva, M; Frances, C. R. L; Costa, J. C. W. A; Vijaykumar, N. L; Hirata, C. M "A New Proposal to Provide Estimation of QoS and QoE over WiMAX Networks"; 2011 IEEE Third Latin-American Conference on Communications, 2011, ISBN 1467302775
- [5] Cecchet, E; Sims, R; Xin, H; Shenoy, P "mBenchLab: Measuring QoE of Web applications using mobile devices" Quality of Service (IWQoS), 2013 IEEE/ACM 21st International Symposium, ISBN 978-1-4799-0589-8, pp. 1 – 10
- [6] Minhas, T;Fiedler, M"Quality of experience hourglass model" Computing, Management and Telecommunications (ComManTel), 2013 International Conference, 2013, ISBN 978-1- 4673-2087-0, pp. 87 – 92
- [7] World Wide Web Consortium (W3C) "QoS" [Online]. Available: <http://www.w3c.or.kr/kr-office/TR/2003/ws-qos/>. (The link is tested up to 14 June 2015)
- [8] Strohmeier, D; Mikkola, M; Raake, A"The importance of task completion times for modeling web-QoE of consecutive web page requests" Quality of Multimedia Experience (QoMEX), 2013 Fifth International Workshop,INSPEC Accession umber:13779622, pp. 38 – 39
- [9] Ramesh, D; Deepak, S "A Quantitative Study of Quality of Experience (QoE) for Decreasing and Increasing Network Performance"Student Thesis, Blekinge Institute of Technology, COM/School of Computing, 2011, pp. 25
- [10] Minhas, T "Network impact on quality of experience of mobile video" School of Computing, Blekinge Institute of Technology, 2012, ISBN 9789172952270
- [11] Shaikh, J;Fiedler, M;Arlos, P;Minhas, T "Classification of TCP Connection Termination Behaviors for Mobile Web",GLOBECOM Workshops (GC Wkshps), 2011 IEEE, ISBN 978-1-4673-0039-1, pp. 1111 – 1115
- [12] ITU-T Recommendation G.114 (2003), One-way transmission time.
- [13] Janne Seppänen ; Martín Varela; Aggeliki Sgora “An autonomous QoE-driven network managementframework ” VTT Technical Research Centre of Finland, PL 1100, Oulu 90571, Finland
- [14] Javed Asghar; Ian Hood and Francois Le Faucheur “Preserving Video Quality in IPTV Networks ” IEEE TRANSACTIONS ON BROADCASTING, VOL. 55, NO. 2, JUNE 2009
- [15] Jingjing Zhang and Nirwan Ansari “On Assuring End-to-End QoE in Next Generation Networks: Challenges and a Possible Solution ” IEEE Communications Magazine July 2011
- [16] Fei Liu; Wei Xiang; Yueying Zhang; Kan Zheng; Hui Zhao “A Novel QoE-Based Carrier Scheduling Scheme in LTE-Advanced Networks with Multi-Service ”
- [17] Karan Mitra; Arkady Zaslavsky; and Christer Åhlund “Context-Aware QoE Modelling, Measurement and

Prediction in Mobile Computing Systems ” IEEE TRANSACTIONS ON MOBILE COMPUTING

[18] Wu-Hsiao Hsu; Sheng-Cheng Yeh; Yuh-Pyng Shieh; Chaur-Heh Hsieh “Web-based QoE Measurement Framework ” 2013 International Conference on Signal-Image Technology & Internet-Based Systems

[19] <http://www.raspberrypi.org/help/what-is-a-raspberry-pi/> (The link is tested up to 14 June 2015)

[20] <https://www.wireshark.org/faq.html#q1.2> (The link is tested up to 14 June 2015)

[21] <https://www.rfc-editor.org/rfc/rfc793.txt> (The link is tested up to 14 June 2015)

[22] Mudassar Ahmad ; Dr. Asri Bin Ngadi ; Ambreen Nawaz ; Usman Ahmad ; Tasleem Mustafa; Ahsan Raza
“A Survey On TCP CUBIC Variant Regarding Performance ” 2012 15th International Multitopic Conferenc

[23] <https://docs.oracle.com/javaee/6/tutorial/doc/gijvh.html> (The link is tested up to 14 June 2015)

