Abstrakt

Bakalářská práce se zabývá problematiku IP telefonie, emulace sítě a kvalitou službou (QoS). Kapitol 2 pojednává o model TCP/IP a OSI, síťových protokolech (IP, TCP, UDP) a o protokolu SIP. Kapitole 3 se zabývá problematikou QoS, definicí QoS, QoS ve službách s reálným časem, parametry a způsoby změření kvality hlasu – subjektivními a objektivními testy. Na základě získaných poznatků byla navržena laboratoř kvality služby. Laboratoř je zaměřena na testování kodeků z hlediska kvality přenášeného zvuku, s ohledem na šířku pásma a ztrátovost paketů. Laboratoř kvality služby byla navržena tak, aby byly splněny požadavky na subjektivní testy uvedené v kapitole 3.4.

Návrh laboratoře

Schéma navržené laboratoř je zachyceno na obr. 4.1. Návrh je složen ze čtyř hlavních částí – Dvou softwarových IP telefonů (X–Lite), síťového emulátoru (WANem – *Wide Area Network emulator*), a SIP ústředny (3CX *Phone System pro Windows* od 3CX Ltd).

Sestavování spojení prbíhá podle stejného schématu viz obr. 4.1. Oba koncové uzly jsou připojeny přes emulátor WANem. Počítač, na kterém je emulátor instalován má dvě síťová rozhraní. PC1, resp. PC2 s WANem tvoří lokální síť 1 (LAN1), resp. lokální síť 2 (LAN2). Pakety prochází z jednoho uzlu do druhého přes WANem.

WANem emuluje chování skutečné WAN sítě, umožňuje provádět v laboratorním prostředí testy na síti s přesně definovanými parametry. Návrh celé laboratoře je virtualizován. Je využita platform VirtualBox od Oracle.

Podrobně je návrh popsán v kapitolách 4.1 a 4.2.

QoS měření

Během měření byly dva parametry sítě proměnné, resp. přesně určené nastavením WANemu – šířka pásma a ztrátovost. Protože testování probíhalo v místnosti, ve které se nachází elektronická zařízení a jiné zdroje ruchů, byla metoda ACR daná doporučením ITU-T P.800 upravena viz kapitola 4.3.1

Testovanému subjektu byly testovací nahrávky přehrávány do sluchátek. Dle ITU-T Rec. P.800 doporučuje několik metod hodnocení kvality služby – MOS_{LQ}, MOS_{LE}, MOS_{LP}, ACR a CCR. K praktickému ověření funkčnosti navržené laboratoře byla

vybrána modifikována verze ACR (M-ACR). Použité náhravky obsahují tři věty, mezi něž jsou vloženy mezery (volný interval – bez zvuku).

Výsledky subjektivních testů jsou shrnuty v tab. 4-1 až tab. 4-10. MOS byl vypočítán pomocí rovnice (3. 1), jehož výsledky shrnuty v tab. 4. 11.

Závěr

Navrhovaná laboratoř byla úspěšně otestována, návrh splnil očekávání. S využitím navržené laboratoře proběhly subjektivní testy kodeku G.711 na 10 osobách.

Abstract

The Bachelor's work, you are reading, deals with the issue of VoIP and quality measurement. In the following paragraphs the discussion will be about the basics of networking protocols such as TCP, UPD, IP and SIP, after that a brief introduction to QoS issues – ways of voice quality measurement; subjective and objective tests. Based on these fundamentals points we will design a QoS laboratory. The laboratory shall test voice codecs with regard to bandwidth, jitter, delay and packet loss. The solution lies in the fact that, which codec is actually suitable for the service.

Klíčová slova:

TCP/IP, UDP, VoIP, QoS, MOS, jitter, ztrátovost paketu, zpožděni.

Keywords:

TCP / IP, UDP, VoIP, QoS, MOS, jitter, packet loss, delay.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma Laboratoř kvality služby jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a~jsem si plně vědom následků porušení ustanovení § 11 a následujících zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne ....

(podpis autora)

PODĚKOVÁNÍ

Děkuji vedoucímu bakalářské práce Ing. Michael Polivka. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

V Brně dne ....

(podpis autora)

Bibliographic citations

ELTAHIR AHMADON, A. Laboratory of service quality. Brno: Brno University of Technology, Faculty of Electrical Engineering and Communication, 2011. 33 P. Thesis Supervisor Ing. Michael Polivka.

CONTENT

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

This Bachelor thesis deals with the issue of *Voice over Internet Protocol* (VoIP) technology, primarily building a laboratory suitable for quality measurements. This laboratory must enable users to conduct subjective tests.

The following section 2 discusses the basics of the network; *Transmission Control Protocol* (TCP)/ *Internet Protocol* (IP) and *Open Systems Interconnection* (OSI) models, network protocols such as TCP, IP, *and Session Initiation Protocol* (SIP) etc. Next following section 3 will discuss *Quality of Service* (QoS) – methods of measuring of voice quality such as *Mean Opinion Score* (MOS), subjective and objective tests. Next we will give a brief discussion about QoS parameters 3.3 *Bandwidth*, *packet los*, *jitters*, etc. Based on user's requirement (in this case our requirement is to fulfill the given assignment of this Bachelor's thesis – feedback "Zadání Bakalářské práce") we will design a laboratory for quality of service 4. The laboratory will focus mainly on testing voice codecs with regard to sound quality with respect to bandwidth, jitter, delay, and packet loss – i. e. *QoS parameters*.

Construction of the QoS laboratory is shown in figures 4.1 and 4.3. The subject (volunteer) will examine a human voice under different network conditions. According to his own perspective he will evaluate the voice quality according to a specific method of quality measurement – *Mean Opinion Score*. Ten students from our colleagues took part in the subjective test. The experiment was conducted in the laboratory on the fourth floor of *Purkynova* building.

2 IP TELEPHONY

2.1 TCP/IP and OSI model

In networking, *Transmission Control Protocol* (TCP) and *Internet Protocol* (IP) play a significant role in: Transmission, guaranteeing delivery to the target station, reporting the existence of transmission errors and many other services. At present, there are about three thousand network protocols[1]. Many of them, however, become obsolete over time, and so now only few remain. Two main sets of specification are taking apart in modern network communication technology; TCP/IP and *Open System Interconnection* (OSI) model.

TCP/IP was created in 1970s from the *Defense Advanced Research Projects Agency* (DARPA) U.S. Department of Defense, often referred to as the *Internet Model*. "From 1973 to 1974, Cerf's networking research group at Stanford worked out details of the idea, resulting in the first TCP specification[9]". It is commonly known as TCP/IP, because of its most important protocols: TCP and IP, which were the first networking protocols defined in this standard [9]. TCP/IP provides end-to-end communication services – specification on how data should be addressed, transmitted, directed and received at the destination. The model has four layers, each one of them with its own protocols. These layers are listed in order from lowest to highest as following: The *link layer* (most commonly *Ethernet*) contains communication technologies for a *Local Area Network* (LAN), the *Internet Layer* (IP), the *Transport Layer* (TCP), and The *Application Layer* (HTTP).

On the other hand model OSI deals with the issue from another perspective. It provides the general principles of network architecture by describing the layers, their functions and services. There are no protocols included that would require too much details. Each of the seven layers tightly explains the functions required for communication. The upper layer is served by the layer below it. The disadvantage of the ISO / OSI model is its complexity.

Table 2.1 illustrates the architecture of these two models (TCP/IP an OSI); their layers name and protocols that is being used. Application Layer provides communication at the highest level, i.e. communication between processes and applications that run on user's computer. This layer also deals with data representation (suitable encoding and decoding data for application transmission) and the management dialogue. Transport Layer creates logical connections between endpoints. Transport protocols divide application data into smaller units called *packet*s, which are transmitted over the network. Internet Layer creates logical connections between computers. Internet layer protocols “route” encapsulated packets called *datagrams* to their destination according to IP address. Internet layer tries to deliver the data using the best path, known as *Best effort delivery*. If the transmission of data failed, the sender is informed about it and he must ensure the re-transfer data. Link Layer describes the standards for the physical medium and electrical signals.

Table2. 1: TCP/IP and OSI models[3].

TCP/IP	OSI
4. Application Layer HTTP, IMAP, RTP, RTSP, SIP, SSH, TLS/SSL, etc.	7. Application Layer HTTP, FTP, SIP, SMTP, Telnet, etc.
3. Transport Layer TCP, UDP, DCCP, SCTP, RSVP, etc.	6. Presentation Layer MIME, XDR
	5. Session Layer Named Pipes, NetBIOS, SAP
2. Internet Layer IP (IPv4, IPv6), ICMP, ICMPv6, IGMP, IPsec, etc.	4. Transport Layer TCP, UDP, PPTP, SCTP, SSL, TLS
	3. Network Layer IP, ICMP, IPsec, IGMP
1. Link Layer ARP/InARP, NDP, OSPF, Tunnels (L2TP), PPP, Media Access Control (Ethernet, DSL, ISDN, FDDI) etc.	2. Data Link Layer ARP, CSLIP, SLIP, Frame relay, PPP
	1. Physical Layer RS-232, Ethernet, POTS, DSL

The traditional way of processing and transmitting analog data over end-to-end network that does not use IP communication technology is shown in figure 2.1.

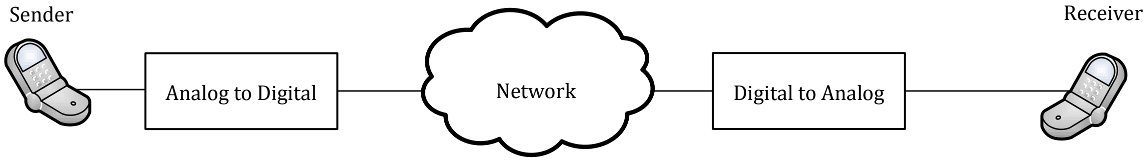


Figure2. 1: Traditional way of transmitting voice.

Data processing or *Encapsulation* (from higher to lower layer on the sender side and from lower to higher layer on receiving side) on nearly all computers is based on TCP/IP model, an example shown in figure 2.2. Useful information or *Payload* (in this case it is an analog voice), is converted from analog to digital form at the sender side using A/D converter. Then the data will be encoded according to the available codecs (PCM, ADPCM, G. 711, G. 719, G. 722, etc.) in the encoder. A codec is a device or computer program (JAVA) [15], usually installed as software on a server, or directly in hardware (ATA, IP phone, etc.), able to encode or decode digitalized data stream. The word codec is a combination of the words "COmpressor-DECompressor" or, more commonly, "COder – DECoder"[17]. Codec usually fulfill three tasks (in some rare cases doing only the last task): Coding – Decoding, Compression – Decompression and Encryption – Decryption.

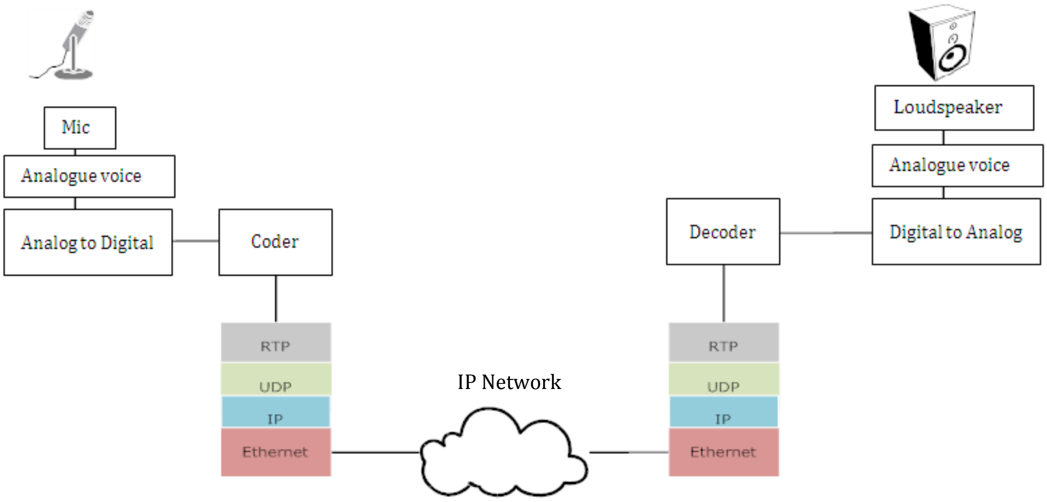


Figure2. 2: Transmitting voice over IP network.

The next process, on The Application layer, is "wrapping" payload into packets using appropriate protocols (RTP – *Real-time Transport Protocol*). Lower layer, Transport Layer, provides the necessary transmission protocols. This will be done through a process that is known by the term *encapsulation*, see figure 2.3. In computer networks this term means that data from lower respectively, higher layer is hidden into new object to which access is restricted to the members of that very layer. Thus, the task of each layer is packaging the data from the higher layer into its own "package". As seen in figure 2.3. Transport Layer in fact unaware of the fact, that the data from Application Layer is basically a converted voice from analog to digital form plus UDP protocol.

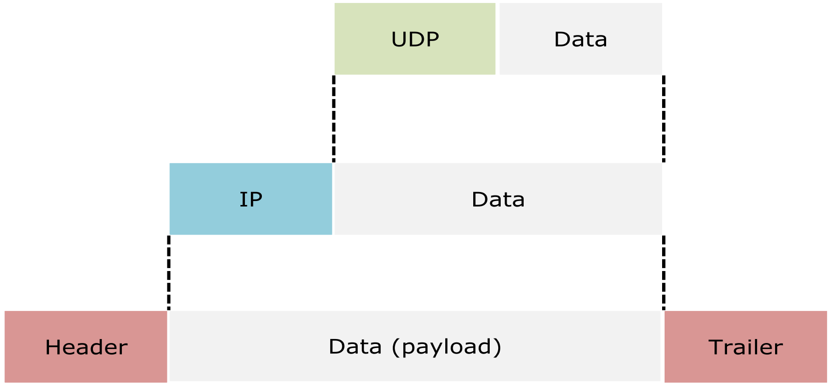


Figure2. 3: Encapsulation data throughout transport, internet and link layer[1].

On Link layer, Data frame is the basic unit for data transfer, which consists of a *Header*, *Payload* and *Trailer*. Data frame is the only packet that has a header and trailer. Other data frames from higher layers have only a header. Data frame header carries a link address of the sender, link address of the receiver and some other management information. In the trailer, Data frame carries among other things a checksum of the packet. Its purpose is detecting the random errors that may have taken place during data transmission. The transmitted data are then usually carried by the network layer packet. On the receiving side, by the inverse procedure, we obtain the original analog voice.

2.2 Internet Protocol – IP

IP is the primary protocol on Internet layer in TCP/IP model. Its role is to deliver packets from the source node to a destination based on their IP addresses. IP address must be a unique identity of the device (network card computers, routers, switches, etc.) in either *Wide* (WAN) or *Local Area Network* (LAN). Each datagram contains source and destination address, so the Internet layer can deliver datagram from the source to its destination. For the purpose of forwarding datagram, the Internet Protocol defines a way of addressing and appropriate structures for datagram encapsulation. IPv4 addresses are 32-bit in decimal format. Thus, each address consists of four octets. The value of each octet can range from 0 to 255, when converting it to binary octet; will consist of eight bits, each of which has a certain value. As for the structure of IPv4, address is divided into three parts, as shown in figure 2.4. Any addresses in the network must necessarily be unique and identifies only one device.

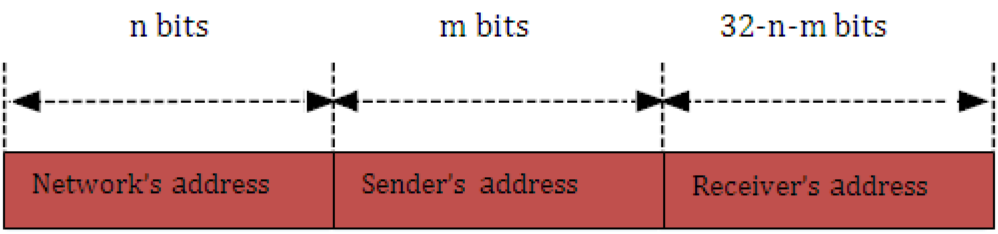


Figure2. 4: IPv4 address structure [8].

Previously, the IPv4 address consisted of only two parts: *ID network* – is used to identify the network or subnet, and *ID host*. Since the network address consisted always only of the first eight bits and the rest of it is for the interface network, there were wasted addresses. Therefore, this classification has not being used.

The *Subnet mask* determines the boundary between the addresses of subnet and interface. As in the case of IPv4 addresses, subnet mask is also a 32-bit, written in the same format as the IPv4 address. Because the part, which contains the Subnet address, can be in different sizes, configurations must be included in each network interface. In short, the subnet mask determines whether the target computer on a local or wide network.

Table 2.2 shows the IPv4 addresses allocation, where each class contains information about how much of IPv4 addresses is reserved for identifying the network and the interface. From the part reserved for the interface address it is possible, if necessary, to subtract a few bits to identify the subnet. Today, the division into classes has been replaced by *Classless Inter-Domain Routing* (CIDR), which is more flexible.

Table2. 2: IPv4 classes [13].

Class	**Network Prefix**	**Mask**	**Network bits**	**Computers bits**	**Number of networks**
A	0	255.0.0.0	7	24	128 (2^7)
B	10	255.255.0.0	14	16	16 384 (2^{14})
C	110	255.255.255.0	21	8	2 097 152 (2^{21})
D	1110	multicasting			
E	1111	Reserved for later use			

It is worth to mention that when we send datagram over IP network, the routers at each node route the packets by using routing protocols such as *Routing Information Protocol* (RIP) or *Open Shortest Path First* (OSPF) based on routing table (either static or dynamical), if the destination is not in that router's vicinity.

Nowadays, mainly used for routing is *Classless Inter-Domain Routing* (CIDR). If a datagram comes to a node, the router will look up to the routing table of the datagram and find the ports through which it can get into network, to which IPv4 destination address belongs. If there are many options, it will select one of them, based on additional information (e.g., according to set metrics, or the congestion ways, etc.). If the datagram is arriving to port, which leads to the network, where the datagram was directed, the router will drop that datagram.

At present, Internet Protocol version 4 (IPv4) is dominant, although the replacement, Internet Protocol version 6 (IPv6), is actively deployed around the world. IPv6 addresses are quite different from IPv4 addresses. They are 128-bit, normally written in hexadecimal format with the colon notation for groups of 16 bits ranging from 0000 to FFFF. Often IPv6 is written in reduced form, either by omitting the leading zeros for each individual group or combining groups of zero and instead write only :: (this replacement, however, can only be used once within the same address). Table 2.3 summarizes various forms of IPv6 entry.

Table2. 3: Various formats of IPv6.

IPv6 address	Description
2001:feca:0000:0000:0000:0000:f563:3574	Standard format.
2001:feca:0:0:0:0:f563:3574	Omitting of leading zeroes of each group.
2001:feca::f563:3574	Omitting of leading zeros of each group and the merger of consecutive zero groups.

2.2.1 IP datagram

Generally, the term packet refers to each message in packet format, while the term datagram is generally used for packets of the "unreliable" service. *Reliable services* are those that warn the user in case of failure of delivery, while *unreliable services* do not. Combination of TCP and IP provide reliable service while UDP and IP provide unreliable service. All these protocols use packets, but UDP packets are generally called datagrams.

IPv4 header consists of:

i. 4 bits containing IP version whether is IPv4 or IPv*6,*

ii. 4 bits containing Internet Header Length, which is multiplied by the length of the header 4 bytes (e. g. 5 means 20 bytes)

iii. 8 bits containing Type of Service - *Quality of Service*,

iv. 16 bits containing packet length in bytes,

v. 16 bits containing an identification tag that is used to recover packet from several fragments,

vi. 3 bits containing the item that defines whether the packet can be fragmented or not (DF: *Do not fragment*) and an item to determine whether further fragments the packet follow (MF: *More fragment*),

vii. 13 bits that contain the fragment offset field to determine the position of the fragment in the original packet,

viii. 8 bits containing 8-bit TTL,

ix. 8 bits containing protocol (TCP, UDP, ICMP, etc. …),

x. 16 bits containing *Checksum* header,

xi. 32 bits IP address of the source,

xii. 32 bits IP address of the destination.

As shown in table 2.1 *Internet Control Message Protocol* (ICMP) belongs to network layer. ICMP is an essential part of the IP protocol suite, as defined in RFC-792 [16]. The main task of this protocol is sending error messages, e. g. *the service is not available* or that *the host or router has not been reached*. ICMP messages are created in the Internet layer, usually from an IP datagram which generates the ICMP response. IP protocol encapsulates the suitable ICMP message with a new IP header (for ICMP message to get back to the original sender) and sends the resulting datagram by the usual way. In the computer network, a new IP header has been introduced *Time-To-Live* (TTL). TTL is set to the default value (usually 64) and automatically reduced by 1 when passing through any router. After reaching zero, the datagram is dropped and the sender is informed by ICMP message. Program *traceroute* performs transmitting UDP datagrams with a special set of IP header - TTL flag. Related utility is program *ping* implemented using the ICMP "*Echo Request*" and "*Echo Reply*" messages. ICMP header begins after the IPv4 header.

In the table 2.4 we can see the ICMP structure, where:

Type – refers to ICMP type.

Code – this field is set from 1 to 15, each number has a different purpose.

Checksum – contains error data checking.

ID – contains the ID value that should be returned in case of *ECHO REPLY*.

Sequence – This field contains a sequence value that should be returned in the case of ECHO REPLY.

Table2. 4: ICMP packet [14].

Bity	0–7	8–15	16–23	24–31
0	**Type**	**Code**	**Checksum**	
32	ID		**Sequence**	

2.3 Transmission Control Protocol – TCP

Due to network overload, traffic load balancing, or other unpredictable behavior of the network, IP packets may get lost or delivered out of order. If IP packet reaches its destination with an invalid *checksum*, TCP rejects it and does not acknowledge its delivery. Therefore, the sender, after time out, retransmits the corrupted packet. TCP can also rearrange packets order that have been sent in the wrong order, and even helps to minimize network overload in order to reduce the incidence of other problems. Since each end of a TCP connection has a limited total of buffer space, a receiving TCP side allows the other side to only send as much data as the receiver can manage. By doing so, a fast host can take an appropriate amount of buffers as well as a slower host [14].

Once the TCP protocol finally prepares a perfect copy of the original transmitted data, it passes datagram to higher Application Layer. The main task of this protocol is to ensure the delivery of packets. Therefore, TCP datagrams sometimes take a relatively long time to reach its destination. TCP protocol is not suitable for real-time applications like *Voice over IP* (VoIP), online games, etc.

Transport Layer protocols, most commonly TCP and UDP, use *port* (an application or process software, used by Transport Layer, serving as a communications endpoint) for *host-to-host* communication. Every port is identified by its number, universally known as the *port number*. *Internet Assigned Numbers Authority* (IANA) is responsible for maintaining the official assignments of port numbers for specific applications.

Construction of a TCP packet consists of the segments; header and data section. TCP header contains 10 fields; mandatory and optional field. Data section follows the header, its content is Payload. Length of data section is not included in the TCP packet header. However, it may be calculated by subtracting the combined length of the header encapsulated TCP and IP header of the total IP packet length (specified in the IP packet header). Table 2.7 shows the structure of TCP header, in which:

Source Port (16 bits) – identifies the sending port,

Destination port (16 bits) – identifies the receiving port, Sequence number (32 bits) - has a dual role:

- If SYN is set, then this is the original sequence number.
- If SYN is empty, then it is accumulated sequence number of the first data byte for the current section.

Acknowledgment Number (32 bits) – If set to ACK, then the value of this field is another sequence number that the receiver expects,

Data offset (4 bits) – specifies the size of the TCP header in 32-bit words. The minimum header size is 5 words and the maximum is 15 words,

Reserved (4 bits) – for future use and should be set to zero,

Flags (8 bits) – contains eight sub-bit,

Windows size (16 bits) - specifies the number of bytes (the serial number in the acknowledgment field) that the receiver is currently willing to accept,

Checksum (16 bits) – used for error checking and data header,

Urgent pointer (16 bits) – if set to URG, this is the 16-bit offset field sequence number indicating the last byte of urgent data, *Options* (0-320 bits divisible by 32) – The length of this field is determined by the data offset field. Options 0 and 1 are eight-bit or byte. Other options indicate the total length of the option (expressed in bytes).

Table2. 5: TCP header [14].

Bit offset	0 – 15			16 – 31
0	**Source port**			**Destination port**
32	**Sequence number**			
64	**Acknowledgment number**			
96	**Data offset**	**Reserved**	**Flags**	**Windows size**
128	**Checksum**			**Urgent pointer**
160 ...	**Options (if data offset > 5)** **...**			

2.4 User Datagram Protocol – UDP

UDP uses an uncomplicated transmitting model without underlying handshaking dialogs for supplying reliableness, ordering, or data wholeness. Hence, UDP provides unreliable service and datagrams may reach out of order, or being missing without observation. UDP takes for granted that error checking and repairing are either not required or performed in the practical application, avoiding the overhead of such processing at the network interface level. Applications with Time-sensitivity frequently use UDP (VoIP, *Domain Name System* (DNS), IP tunneling protocols, and many online games), because it is preferred to wait for the dropped packets, which may not be an option in a real-time system.

UDP header consists of 4 fields. Using two of them are optional in IPv4 (light red background table2.7). In IPv6, only the source port is optional. UDP header has:

Source Port Number – This field identifies the sending port, which should be meaningful, if used. If it is not, then it should be zero.

Destination Port Number – This field identifies the destination port.

Length – 16-bit field indicates the length in bytes of the whole datagram: header and data. Minimum length is 8 bytes, because it is a header. The size field contains the theoretical limit of 65 535 bytes (8 byte header + 65 527 bytes of data) for the UDP datagram. The practical limit for the length of the data that are stored in the base IPv4 is 65 507 bytes.

Checksum – 16-bit checksum field calculates error checking of headers and data. The algorithm for calculating the checksum is different for transport over IPv4 and IPv6. If the checksum is omitted in IPv4, the field uses the value of all-zeros. This field is not available for IPv6.

Table2. 6: UDP packet structure[14].

Bity	0–15	16–31
0	**Source Port Number**	**Destination Port Number**
32	**Length**	**Checksum**
64	**Data**	

2.5 SIP protocol

In *Voice over Internet Protocol* technology, one of the most essential protocols is *Session Initiation Protocol* (SIP). It is a signaling protocol defined by *Internet Engineering Task Force* (IETF) used for moderating communication sessions. The protocol can be used for preparation, supervision and drop the connection between two or more parties, serving four main purposes; establishment of user location – i. e., translate from user's name to network address, call management, allowing for changing properties of a session, and feature negotiation, for the members of session to agree on the features they will support. Inside the signaling message, SIP encapsulates messages of another protocol, which specifies the encoding method used for multimedia data transmission, its parameters and port numbers, on which data should be transmitted or received. Usually, *Session Description Protocol* (SDP), which is also text based protocol, is used for this purpose.

SIP devices can establish a session directly with each other, but more common is that they apply to one or more SIP proxy servers. These servers also can fulfill (and most likely they do) function of the SIP PBX, to which different participants are registered.

Quite often during communication occurs dropping call. The principle of solving this issue is very similar to http, which uses error messages logs. In addition different error messages have numeric and text label, for example, 200 – OK, 100 - Trying, 180 - Ringing, 486 - Busy Here etc. as shown in the figure 2.5.

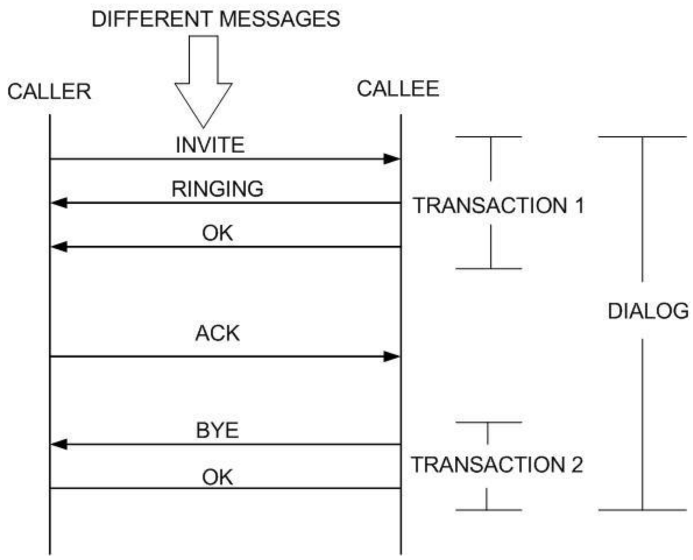


Figure2. 5: Example of SIP communication [10]

Errors are divided into the following categories:

1xx - Progress - a process without a problem, but not completed yet,

2xx - Success - process was completed without problems,

3xx - Redirection - process in progress,

4xx - Client error - request is incorrect and cannot be processed by the server,

5xx - Server Error - request is probably fine, but the error is on the server side,

6xx - Fatal error - quite fatal error, that cannot be processed in any way.

3 QUALITY OF SERVICE

Quality of Service (QoS) can be interpreted as the ability of allowing the transport of traffic with special requirements. It may also refer to ensuring a certain level of performance, for the sensitive and important information to arrive in time and safely. Protocols of QoS working on slowing unimportant packets, and in case of extreme network load, throwing them out. The device, which perceives the priorities of different packets and according to their priorities, classifies them, known as the router. However, in this thesis we will not deal with what is in the active components (router, switch, bridge, etc.) takes place. Therefore, this situation is represented by cloud as seen in figure 3.1.

3.1 QoS definition

QoS can be defined in two terms; QoS experienced by the end user (such as IP telephony, computer or other communication – terminal – equipment), and in term of network. As illustrated in figure 3.1, which shows end-to-end connection, the end user may also be a person, who uses the terminal device. From the perspective of network, QoS refers to the ability of a network to provide a certain level of quality. QoS is defined in several ways and there is no standard definition. It always depends on where, how and why to use it. Combining all of these definitions is really the best definition. Technically, QoS is a set of techniques to manage bandwidth, delay, jitter, and packet loss in the network. It is possible to manipulate one or all of them.

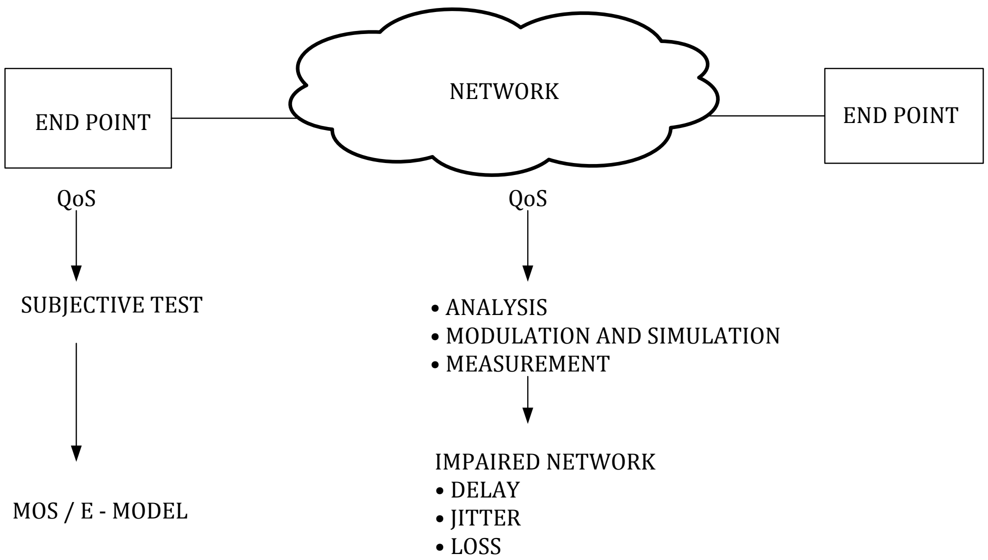


Figure3. 1: QoS definition [1].

3.2 Significant role of QoS

At the beginning of telecommunication technology, there were basically two separated networks; network for voice, and the other one is for data. Each one of them was, with a simple goal, aimed to transmit a certain type of information. Older network known as *Public switched telephone network* (PSTN), which was introduced with the invention of the telephone. This network was designed to transmit only voice. On the other hand, IP networks were designed for data transmission.

During the whole period of telephone communication, the network is dedicated to a single call. Once the call is terminated, circuits (*Circuit switched Network*) will be ready to create the next call. The quality of voice transmission depends on quality of the *end-to-end* network during a call, such as transmission loss, circuit noise, echo, etc. Therefore, the initial telephone network was designed in specific way to have impaired parameters in acceptable level. The voice was (and still is) a communication *real-time service*.

IP network, on the other hand, was a completely different type than the telephone network. Firstly, it was designed to transmit data that was (and mostly still is), unlike voice service, a *non-real time service*. In addition, data can be stored in the network buffer for later delivery. If the information has been received with an error, the network is capable of resending them again. Data services are sometimes referred to as "*store-and-forward*" services. Because earlier IP network has been transmitting only one kind of information, network can be designed to operate in so-called Best Effort mode, allowing processing all packets equally. Developers have begun to invent a new network that would integrate both types of services (real and non-real time services).

In the mid-20th century, these two independent networks have begun to unite under the term of Voice and data convergence [1]. Voice and data convergence is currently represented in some systems such as WiMAX, which must meet the requirements for data, voice, video conferencing, etc., with support for multi-services and QoS management. WiMAX supports IPv4, IPv6, ATM, Ethernet, etc. [19]. In the converged network, best effort (Best Effort) is not the right choice to meet a different requirement of their applications or services, even though it may be appropriate for traditional Internet applications such as file transfers (FTP), web browsing (HTTP), e–mail. QoS is simply a technology that provides a solution to this technical problem.

There are four parameters in network traffic can affect the required quality: Bandwidth, Delay, Jitter and Packet loss.

3.3 QoS for real – time services

User ranking is determined by subjective and objective tests. Parameter that can affect the quality of services in real time may be one or a combination of: the quantization noise, **bandwidth**, the ratio of bit errors, delay, jitter, **loss rate**, **codec options**, Echo control, network design and more. However, in this thesis, attention will be paid to the bold marked parameters which are highly interdependent. This unfortunately may lead to the case, when improvement of one parameter causes weakening to the other one and vice versa.

For testing and ranking the quality of transmitted voice signal over telecommunication equipment and systems, there are two basic methods, subjective and objective. Methods are described in details in ITU-T Series P.800, P.830 and P.862.

3.3.1 Delay

The delay affects mainly the real-time services such as voice or video. One-way delay is the amount of time measured from the moment the speaker started speaking, until hearing his sound on the receiving side. Two-way delay, of course, will be the sum of two one-way delays. ITU-T Recommendation G.114 recommends a maximum of a 150 ms one – way delay.

The main causes for delay include:

Source coding – A/D and D/A converter or Frame delay, Packetization delay, *Channel coding* – Detection and correction of errors or interleaving, jitter, buffer delay, Propagation, delay and more.

Used codec contributes in delays caused by source coding, packetization, channel coding, and jitter buffer. The network, on the other hand, contributes in delays caused Propagation delay, which defines the time that the packet needs to get from one end, throughout the medium, to the other end. Maximum energy speed on electrical or optical interface approaches the speed of light. For such delays, unfortunately, we cannot do anything because we are limited to the laws of physics! However, the only variable that affects Propagation delay is the length of cable.

3.3.2 Jitter

In computer network and mostly in networks based on IP, such as Internet, jitter means a difference or variation in delay between selecting packets with any lost packets being ignored in end-to-end network, known also as packet delay variation (PDV). We can generally state that the jitter affects each two packets in row having different amounts of delay. The question is whether the applied jitter pressed onto enough, relegated to the service. However, some traffics, such as digitalized voice, requires that the packets must be transmitted in a single "period", i. e., packets should arrive to the destination with same distance time between them - isochronous traffic.

3.3.3 Packet loss

Data in real-time service, such as voice, packet loss may result in very noticeable performance problems or jitter, leading to a voice cutting or hop. That reduces the speech intelligibility and video quality. Here we list some of the main loss sources: *Bit error rate* - due to impairing of transmission line parameters such as "fading", *circuit noise*, *packet collision*, *processing errors*, *buffer overflows*, *random packet delay* and more.

3.3.4 Bandwidth

The Main goal is to reduce the bandwidth requirements as low as possible. To do so we can consider RTP header compression, silence suppression, and RTP multiplexing [11].

In VoIP network there are three headers transmitting voice samples through network. We are talking about IP, UDP and RTP headers. Header of IPv4 is 20 octets, 8 octets for UDP header, and 12 octets for RTP header. The total length is 40 octets (320 bits). If we would like to calculate a required IP bandwidth for G. 729 for instance, given that packet duration is 30 ms that means

$^1/_{30\ m} \cdot 320 = 10666{,}7$ Header bits per second. That gives us 27 kbps (16 000 + 10 666.7) IP bandwidth required. Table 3.1 shows some other examples.

Table3. 1: IP Bandwidth for different Codecs [11].

Codec	Bandwidth(kbps)	Packet duration(ms)	IP Bandwidth(kbps)
G. 711	64	30	75
G. 726	32	30	43
G. 729	16	30	27

3.4 QoS measurement

There are many approaches to QoS measurement; primarily known subjective and objective measurements. Subjective tests, groups of people (subjects) evaluate the QoS under given conditions and circumstances. Most commonly subjective method is *Mean Opinion Score* (MOS) as specified in ITU – T Recommendation P.800. As for objective tests, there are various computational models or applications for QoS evaluation. Most notable objective methods are *Perceptual Speech Quality Measure* (PSQM, Recommendation P.861) and *Perceptual Evaluation of Speech Quality* (PESQ, Recommendation P.862). Our goal, however, is to prepare an appropriate laboratory, which enables the user to perform a subjective test.

3.4.1 Subjective tests

Subjective tests or also *Mean Opinion Score* are intended for laboratory that simulates the network conditions for testing purposes. When we are performing subjective tests it is necessary to select appropriate conditions, environment, and test subjects. It is also essential to correctly manage the test procedures. Test principle is as follows: one test subject (human) is placed on a workstation supplied with headset. The subject will listen to some sentences from headset, and then he evaluates the quality of that voice. The room size shall not be less than 20 m^3 and the noise must be kept as low as possible.

MOS is expressed as a number ranging from 1 to 5 where 1 is the lowest rate and 5 is the highest rate. Quality assessment is performed by predetermined sentences in the table 3.2:

Table3. 2: Rating scheme for MOS tests [ITU – T rec. P.800].

MOS	**Quality**	**Impairment**
5	Excellent	Imperceptible,
4	Good	Perceptible but not annoying,
3	Fair	Slightly annoying,
2	Poor	Annoying,
1	Bad	Very annoying.

For all the subjects participating in the subjective test, the MOS will be calculated according to the following equation 3.1 [18]:

$$\mathrm{MOS} = \frac{(N_E \times 5) + (N_G \times 4) + (N_F \times 3) + (N_P \times 2) + (N_B \times 1)}{N_t} \tag{3.1}$$

$$N_t = N_E + N_G + N_F + N_P + N_B \tag{3.2}$$

Where N is the number of subjects, the N index indicates the first letter of evaluation of quality (e. g. N_E the number of people who give 5 score – Excellent) and N_t is the total number of subjects.

The empirical tests of subjective evaluation, showed that the results of the individual subjects may vary significantly depending on; surrounding noise, hearing status of the evaluators, the test environment, etc. If the noise level is lower less than 12 dB, the signal becomes completely incomprehensible. Recommendations require for the noise level to be as low as possible, ideally should be 60 dB lower than the sample under test. Recommendations require for the room, in which test is conducted, to be larger than 20 (30) m^3. This room should also be low-reflection. In smaller rooms may lead to distortion due to reflection[12].

3.4.2 Objective test

Proposal for objective measurement methods based on human perception began in 80's. Among the most essential algorithms we have *Perceptual Audio Quality Measure* (PAQM), PSQM, *Noise-To-Mask Ratio* (NMR), Perceval, *objective audio signal evaluation* (OASE), and *Perceptual Objective Measurement* (POM). With the exception of PSQM all these algorithms have been proposed for estimating the quality wideband codec, and so they were rather intended for television and radio broadcasting.

With gradual development of speech coding, especially voice over IP, developer have developed new algorithms testing the quality of speech transmission, because PSQM did not cover the entire area of interference (noise). Other algorithms were PSQM99 and PAMS, but these algorithms are not good enough to revise the standard. In 2000 ITU-T Rec standardized algorithm PESQ (Perceptual Estimation of Speech Quality). We will not perform objective test in this thesis.

4 LABORATORY OF QOS

Our goal is to design a QoS laboratory, which has a core PC equipped with two network cards and software allowing loss packet, jitter and delay control. Integral part of the lab equipment is also SIP PBX software. This QoS lab should allow subjective tests. Laboratory proposal shall be virtually executed. In the proposed laboratory test most commonly used codecs virtually the user group.

4.1 Laboratory proposal

QoS laboratory design according to figure 4.1. has four main components; two virtual IP phones (x-lite), WANem unit (WANem – Wide Area Network emulator), which enables us to change QoS parameters – packet loss, jitter and delay, and the last component is SIP PBX (3CX Phone System for Windows by 3CX Ltd), whose main task is signaling and storing users profile.

Connection flow is shown in the same figure 4.1. Both end users will be connected with each other via WANem unit. In this case host computer (on which WANem is installed) will have two active network interfaces. Thus, PC1, respectively PC2 with WANem will create a local area network 1 (LAN1), respectively local area network 2 (LAN2). This way we can apply various changes on tested network. There are other methods for the same purpose mentioned in the manual of the manufacturer.

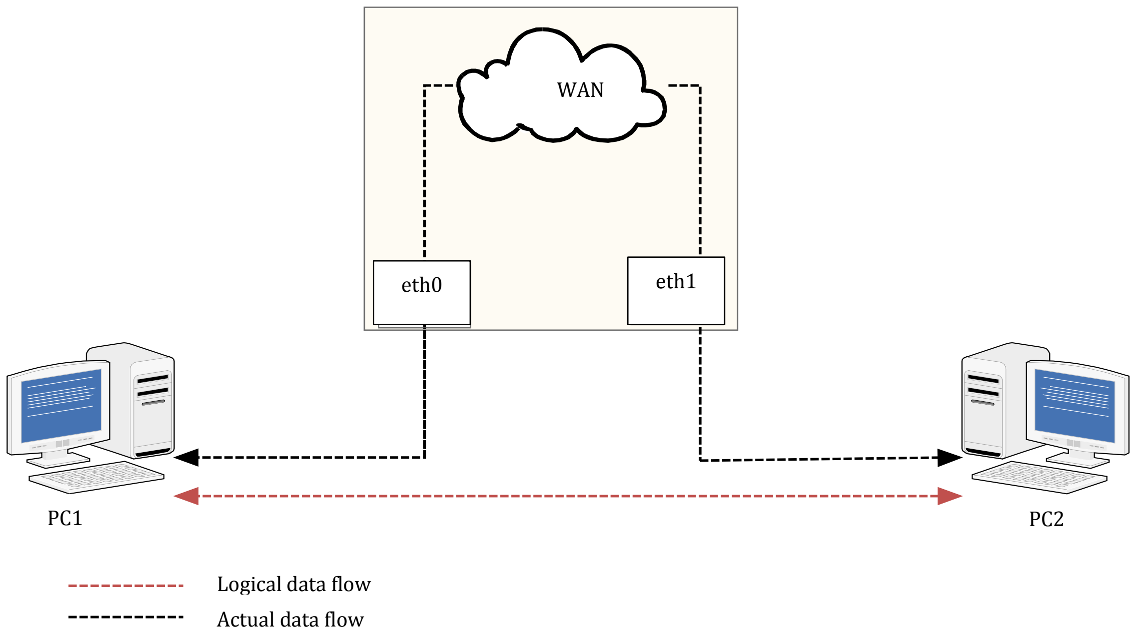


Figure4. 1: Data flow concept.

A control unit WANem issued by *Performance Engineering Research Center* (PERC) is basically an additional program running on a Debian – Linux platform, which can simulate (imitate) a WAN network in development or under testing. The host program, on which laboratory components are functioning, is VirtualBox from Oracle, which creates a virtual environment for all laboratory components. In this environment it is possible to "create" more useful devices for various purposes. This way, we can build the whole laboratory facilities into a single computer.

4.2 Adjustment of laboratory component

After installing VirtualBox and lunching it, we enter the computer's name and the type of operating system running on the computer. Lab computer names are PC1 and PC2. Then we set the amount of system RAM, which is subtracted from the total amount of host computer's memory. For better efficiency, it is recommended 512 megabytes for XP windows. Next step is selection virtual hard disk to be utilized as a boot disk. After confirming those options, this device will appear on the main VirtualBox window. Three devices are needed for the purpose of this experiment. It is all illustrated in figure 4.3.

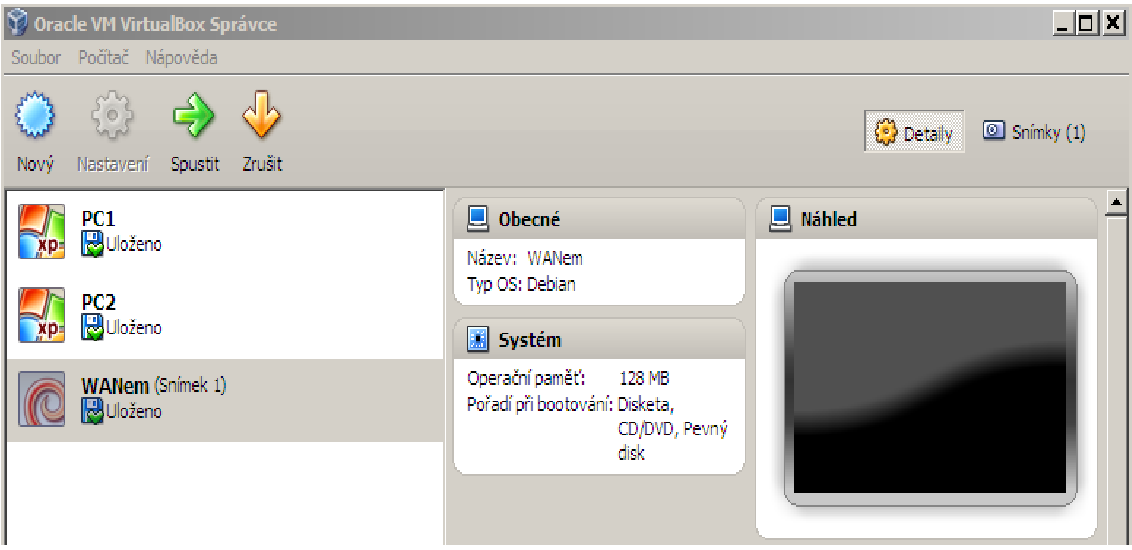


Figure4. 2: Screenshot of the VirtualBox main window.

The most essential procedure is setting the network interfaces of QoS lab devices. Computer PC1 and PC2 are on two separated local area networks as required in subsection 4.1. To do so, we click on "Network" button. A window with three variable attributes (i. e. the connection method, network interface card and MAC address.) for network configuring will show up. VirtualBox allows allocation up to four network card for each device.

We will configure the network settings for PC1, respectively PC2 as following;

- Connection: Internal network name: intnet1 respectively Intnet2,
- Advanced: default settings.

Network interfaces settings for WANem will be the same, just with three additional network cards. The first is connected with the PC1 through network intnet1, the second is connected with the PC2 through network intnet2, in this way each of them will be logically placed on a separated local area network (Intranet) and the last card will be connected via the Network Bridge through witch we change network parameters.

XP windows will be running on both PC1 and PC2. Two softwares for QoS lab are required; X-Lite and SIP PBX - just on one computer. Then we can use useful software (Wireshark) that allows us to monitor (capture) sending and receiving packets. This allows the user to measure delay, jitter and packet loss or graphically display them.

One of the useful features that VirtualBox provides is DHCP (Dynamic Host Configuration Protocol) server. This feature allows us configuring network interfaces automatically by using DHCP. To modify DHCP server we use program VBoxManage.exe running on DOC platform. The following commands execute the necessary settings.

Network's card on LAN intnet1:

```
cd "C:\Program Files\Oracle\VirtualBox"

C:\Program Files\Oracle\VirtualBox\VBoxManage dhcpserver add --
netname intnet1 --ip 169.254.55.1 --netmask 255.255.255.0 --
lowerip 169.254.55.100 --upperip 169.254.55.200 --enable
```

Network's card on LAN intnet2:

```
C:\Program Files\Oracle\VirtualBox\VBoxManage dhcpserver add --
netname intnet2 --ip 169.254.242.1 --netmask 255.255.0.0 --
lowerip 169.254.242.13 --upperip 169.254.300.20 --enable
```

After running all devices and if the settings of active elements (network cards, RAM, hard disk, etc.) are correct, three windows will show up, each window shows its own workstation environment. By now we are done on the host computer.

We move onto next step, which is adjustment of each workstation separately. Starting with WANem, we insert the CD into the CD ROM drive of our chosen PC. The important commands are listed in the manual of WANem, or via the command help. However, for this experiment we will use the following commands;

On WANem screen we enter the following commands

```
assign 169.254.55.100 eth1
assign 169.254.242.13 eth2
```

WANem assigns the network card of PC1 respectively PC2, which is the LAN1 respectively LAN2, the first respectively second network card of WANem device. On this basis, WANem will construct a routing table, by which network traffic is set.

It is necessary to navigate packets on PC1 to the desired destination. For that we are going to use *route add* command. The command runs on cmd.

```
C:\> route add 169.254.55.100 mask 255.255.255.255 169.254.245.20
```

First IP is for a computer, to which our data is hiding, and the second IP is for WANem (also known as the Gateway). We do the same on the second computer.

```
C:\> route add 169.254.242.13 mask 255.255.255.255 169.254.55.120
```

Figure 4.3 shows the actual and logical data flow direction in details, where discrete line represents the logical connection between devices, while the solid line represents the actual connection. SIP PBX interacts with PC2 via PC1 Network Interface Controller (NIC). Jitter, delay and other QoS parameters will be set by a remote computer via eth0. Graphical User Interface (USI) of WANem, which is shown in Annex A, gives us the ability to apply the required settings on sending packets.

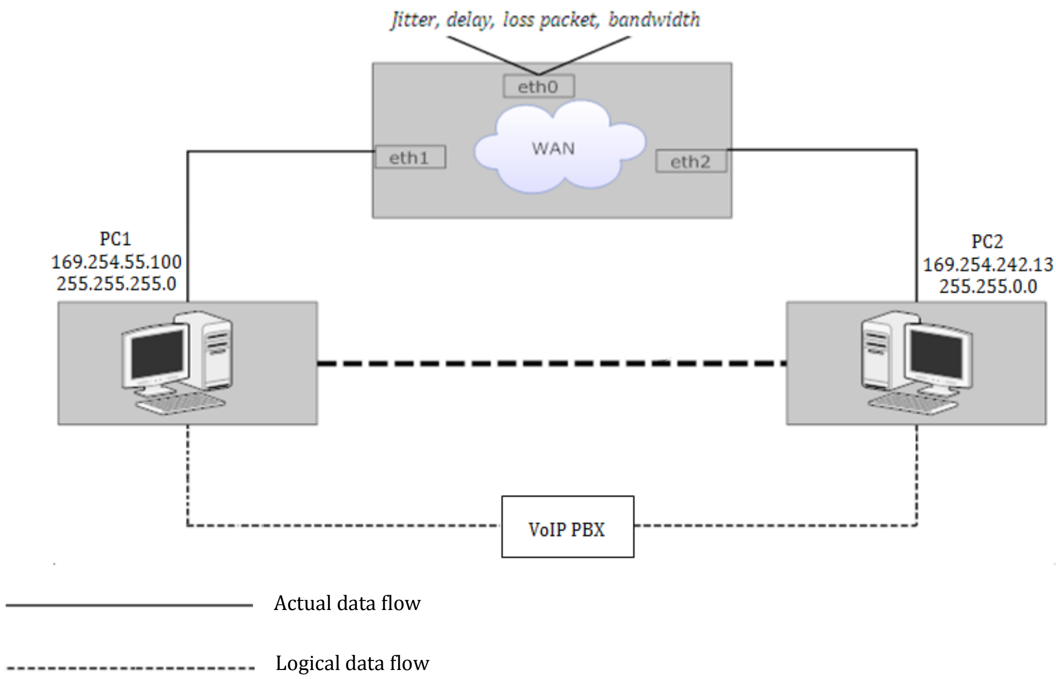


Figure4. 3: Data flow direction in QoS laboratory.

On a remote computer, using Internet Explorer, Firefox, or other functionally similar program, we go to URL http://wanemip/WANem, where wanemip is the IP address of eth0 of WANem and hit Enter button, a GUI shows up, with five options in it as shown in Annex A.

Figure4. 4: Screenshot of two end-users when they are connected properly.

4.3 Procedures for QoS measuring

4.3.1 Subjective test measurement

In this experiment we will use a simplified method of Absolute Category Rating (ACR), in which subjects give their opinion using an absolute quality scales (excellent, good, etc.). We will turn a blind eye on some of the ACR recommendation. To avoid confusion we can call it Modified ACR or M-ACR. Subjects participating in the subjective tests are chosen randomly from the community school, with stipulations that:

1. They have not been directly involved in work connected with valuation of the performance of speech coding, or related work; and
2. They have not participated in any subjective test whatsoever for at least the previous six months and not in a conversation test for at least one year.
3. The balance numbers of male and female subjects was not taken into consideration.

The experiment is conducted in a laboratory, in which exist several machines (PCs, monitors, routers with cooling system, etc.). That means it is unmanageable to meet the required specification in ITU Recommendation P. 800 regarding environmental noise. However, to minimize the noise from the machines as low as possible, we use a headset with noise cancelation.

According to ACR, as recommended in P. 800 Annex B, samples of speech should be prepared in recorded or stored form to avoid any unwanted variability in the speech source.

Figure4. 5: Example of the speech segment in program Audacity.

The speech material (*Segment 1 - 16 bit PCM, 44.1 kHz project rate -*) as shown in figure 4.5 is consisted of simple, meaningful and short sentences which are easy to understand with duration 9 to 12 seconds. These sentences were made up into random order with duration 3 to 4 seconds in a way that there is no obvious connection of meaning between them. According to Recommendation P. 84 Annex C, if N sentences per talker are used there will be N * (N-1) possible sentence combinations per talker. Since we are using 3 sentences the possible combinations will be 6. In order to avoid the complicity and save the time we will use in the whole experiment just one segment with 3 sentences in it for M-ACR method. This segment shall contain several silent periods; between the sentences 1 to 1.5 seconds, and at the begging and the end 1 to 2 seconds gab.

With table 4.1 in mind we will conduct 6 different tests for each subject. Assuming each test takes 15 seconds, changing parameters takes 30 to 60 seconds and answering the questions between 3 to 6 seconds, then the M-ACR for one subject shall take approximately 6.5 minutes. Thus, for 10 subjects the experiment takes 1.1 hour. Because it's inconvenient for the experiment to be conducted in one session, it is prudent to sub-divide the experiment into two or more sessions. Ideally no session should last for more than 20 minutes and in no case should a session exceed 45 minutes [Recommendation P. 800 Annex B], consequently we will have 2 sessions.

As was mentioned before, we conducted 6 tests for each subject. Each test has a different network condition as shown in table4. 1. Tests from 1 to 3 were conducted using G. 711 μ-law and from 4 to 6 using G. 711α-law. We have chosen those codecs, because they are related, which makes it easy to compare them with each other.

To summarize our experiment results we will use MOS formula (equation 3. 1) to evaluate the subjective tests for QoS laboratory. All the results from subjective test are moved to Annex C.

MOS calculation example:

$$MOS_{for\ test1} = \frac{(2 \cdot 5) + (3 \cdot 4) + (5 \cdot 3) + (0 \cdot 2) + (0 \cdot 1)}{10} = 3.7$$

Table4. 1: Numerical values of different network parameters.

	Codec	Bandwidth [kbps]	Loss packet [%]	Duration [s]	MOS for M-ACR[-]	Test number
Segment 1 (16 bit PCM, 44.1 kHz project rate)	G. 711 µ-law	80	0	1:40	3.7	1
		30	0	1:40	2.2	2
		80	50	1:40	1.2	3
	G. 711 α-law	80	0	1:40	3.2	4
		30	0	1:40	1.6	5
		80	50	1:40	1	6

So, for a network with no loss packet and 80 kbps of bandwidth, this network yields MOS of 3.7 for G. 711 µ-law and 3.2 for G. 711 α-law. However, if the bandwidth is lower than what codec requires (using the concept from subsection 3. 3. 4) the network yields MOS of 2.2 for G. 711 µ-law and 1.6 for G. 711 α-law. A bad performance the network shows when the loss packet is about 50 %.

Figure 4.6 shows the relationship between the network under testing and its MOS evaluation. Network 1 shows best performance while network 6 shows the worst performance.

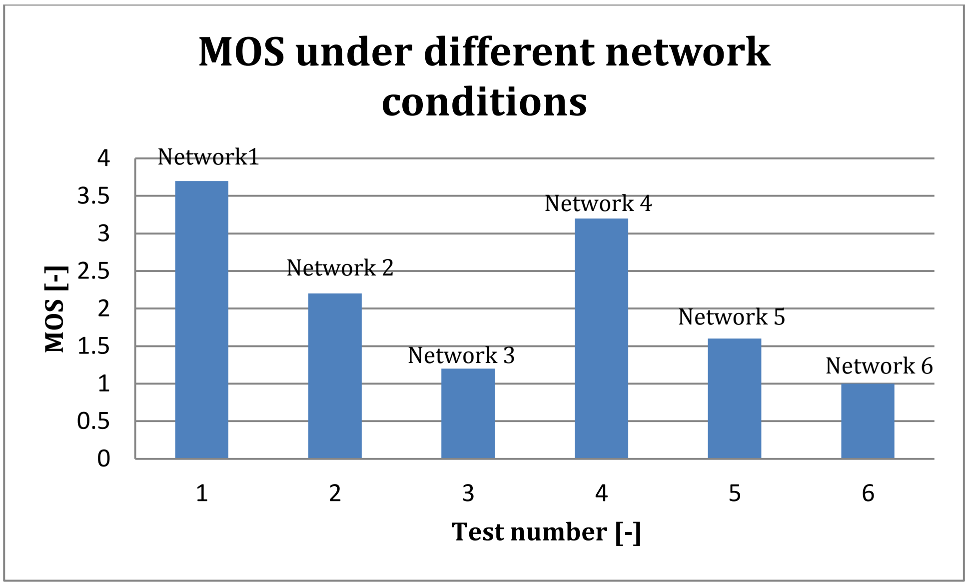


Figure4. 6: MOS evaluation under different conditions.

5 CONCULATION

Our goal from this study was to suggest a laboratory proposal for QoS evaluation. In the field of QoS measurement, there are many approaches; Subjective and Objective measurements. However, for the sake of this BC assignment, we included only subjective measurement.

QoS laboratory design was created in a virtual PC environment. For this purpose, we used VirtualBox software. Basically, all of laboratory facilities were built in that software, which gives us the mobility advantage (if the software is installed on a mobile device - laptop for instance). Two of the laboratory facilities represent end – users, and the third represents WAN emulator.

Annex C shows the results that we got from subjective tests. Each table has four attributes;

Segment: The speech material that we used for the experiment it is included in submitted Bachelor documents.

Score: Goes from 5 (Best quality) to 1 (Bad quality).

Quality: Quality evaluation in words.

Impairment: Description of the evaluated quality.

For the sake of simplicity we modified ACR method to meet our requirement, in fact we named it M-ACR. The experiment was conducted in a laboratory, in which exist several facilities (PCs, monitors, routers with cooling system, etc.). That means it is unmanageable to meet the required specification in ITU Recommendation P. 800 regarding environmental noise. However, to minimize the noise from the machines as low as possible, we use a headset with noise cancelation. We conducted subjective test, using M-ACR method, to prove only that our design is functioning. Judging from the results we got from subjective tests, it is clear that our design is working as we expected without any problems.

LITERATURA

[1] A Handbook for Successful VoIP Deployment: Network Testing, QoS, and More by John Q. Walker, *NetIQ Corporation*.

[2] Arindam Paul, QoS in Data Network: Protocols and Standards.

[3] DOSTÁLEK, Libor, KABELOVÁ, Alena. Velký průvodce protokoly TCP/IP a systémem DNS. 3. vyd. Praha: Computer Press, 2002. 542 s. ISBN 80–7226–675–6.

[4] Gen2 Ventures, *Clearing the Way for VoIP*, White Paper, www.gen2ventures.com

[5] Jan Kubr, Y36SPS QoS, 2008.

[6] Kun, I., Park, *QoS in Packet Networks*, the MITRE Corporation USA, 2004. LESTARI, A.A., YAROVOY, A.G., LIGTHART, L.P. Numerical and experimental analysis of circular–end wire bow–tie antennas over a lossy ground. *IEEE Transactions on Antennas and Propagation*. 2003, vol. 52, no. 1, p. 26–35.

[7] Hubert Zimmermann, Invite paper, IEEE Transactions on Communications, April 1980.

[8] Sven Ubik: QoS a diffserv – Úvod do problematiky, Technická zpráva TEN–155 CZ číslo6/2000.

[9] RFC 675, *Specification of Internet Transmission Control Protocol*, V. Cerf et al. (December 1974)

[10] Http://www.siptutorial.net/SIP/relation.html

[11] Bandwidth requirements for Voice over IP transmission, http://www.erlang.com/bandwidth.html (accessed May 11, 2012).

[12] http://access.feld.cvut.cz/view.php?cisloclanku=2007030002

[13] RFC 791, *Internet Protocol*, Information Sciences Institute (September 1981)

[14] W. Richard Stevens. The protocol TCP/IP illustrated, volume 1, 9780201633467.24290.

[15] Volker R., Alberto S., Peter E. CODEC Tutorial, November 6, 2007

[16] RFC 792, INTERNET CONTROL MESSAGE PROTOCOL, DARPA INTERNET PROGRAM PROTOCOL SPECIFICATION (September 1981)

[17] Wendell Odom, Michael J. Cavanaugh. Cisco QOS Exam Certification Guide, Second Edition Published by: Cisco Press ISBN1–58720–124–0.

[18] H. R. Tizhoosh, http://pami.uwaterloo.ca/tizhoosh/observer.htm, June 1997

[19] Stanislav H., Josef F., Vit S., Radiove a mobilni komunikace, Elektronicka Skripta

LIST OF SYMBOLS AND SHORTCUTS

ADPCM–Adaptive differential pulse – code modulation

ARP – Address Resolution Protocol

ATM– Asynchronous Transfer Mode

CIDR–Classless Inter – Domain Routing

COPS– Common Open Policy Service

CoS – Class of Service

CSLIP– Compressed Serial Line Internet Protocol

DARPA– Defense Advanced Research Projects Agency

DCCP– Datagram Congestion Control Protocol

DF– Don't fragment

Diffserv– Differentiated services

DNS– Domain Name System

DSL– Digital Subscriber Line

FDDI– Fiber Distributed Data Interface

FTP– File Transfer Protocol

GSM– Global System for Mobile Communications

HTTP– Hypertext Transfer Protocol

IANA– Internet Assigned Numbers Authority

ICANN– Internet Corporation Assigned Numbers Authority

ICMP– Internet Control Message Protocol

IETF– Internet Engineering Task Force

IGMP – Internet Group Management Protocol

IMAP– Internet Message Access Protocol

IntServ – Integrated Services

IP – Internet Protocol

ISDN – Integrated Services Digital Network

ISO – International Organization for Standardization

MF – More fragment

MIME – Multipurpose Internet Mail Extensions

MOS – Mean Opinion Score

NDP – Neighbor Discovery Protocol

OSI – Open System Interconnection

OSPF – Open Shortest Path First

PCM – Pulse–code modulation

PPP – Point–to–Point Protocol

PPTP – Point–to–Point Tunneling Protocol

QoS – Quality of Service

RFC – Request for Comments

RSVP – Reservation Protocol

RSVP – Resource reservation Protocol

RSVP –Resource reservation Protocol

RTI – Real–Time Intolerant

RTP – Real Time Protocol

RTP – Real Time Protocol

RTS – Real Time Services

RTSP – Real Time Streaming Protocol

RTT – Real–Time Tolerant

SCTP – Stream Control Transmission Protocol

SCTP – Stream Control Transmission Protocol

SIP – Session Initiation Protocol

SLIP – Serial Line Internet Protocol

SSH – Secure Shell

SSL – Secure Socket Layer

TCP – Transmission Control Protocol

TLS – Transport Layer Security

ToS – Type of Service pole

TTL – Time to Live

UDP – User Datagram Protocol

VoIP– Voice over IP

WiMAX – Worldwide Interoperability for Microwave Access

WWW – World Wide Web

RTSP – Real – Time Streaming Protocol

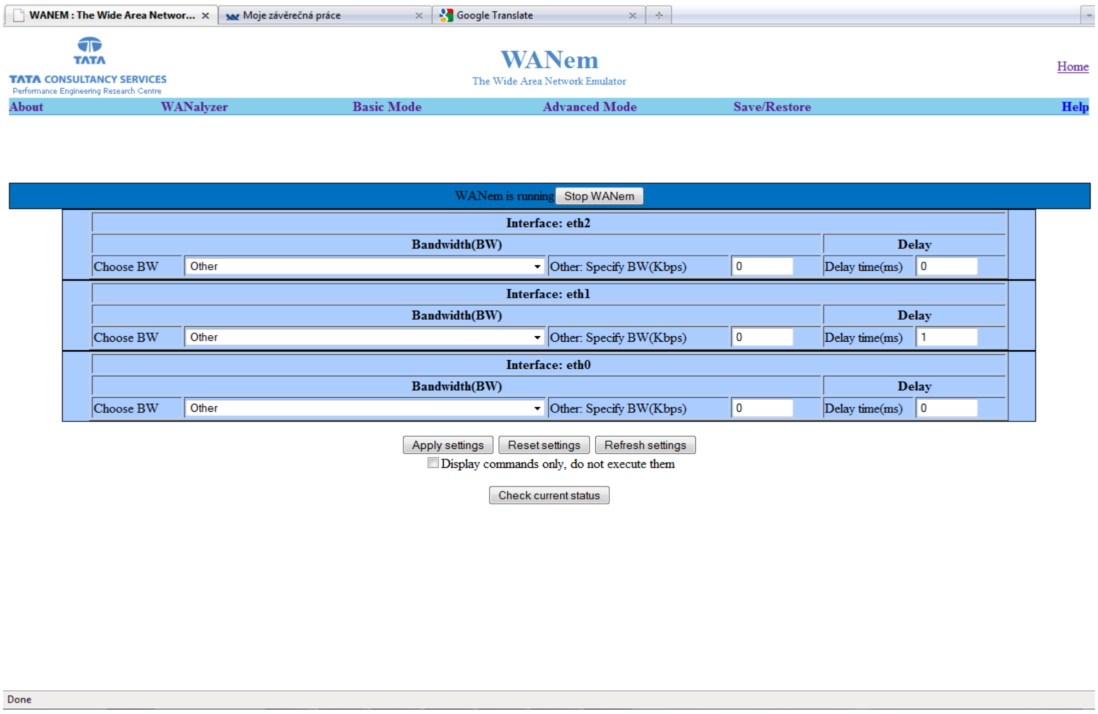
WANem – Wide Area Network emulator

DHCP – Dynamic Host Configuration Protocol

USI – Graphical User Interface

LIST OF ANNEXES

A. Screenshot of Graphical User Interface of WANem



B. The form we used for QoS test

<table>
<tr><td colspan="2">SUBJEKTIVNI TEST (MOS)</td></tr>
<tr><td>VĚK</td><td>POHLAVÍ</td></tr>
<tr><td colspan="2">Jak byste hodnotil srozumitelnost hlasu (zaškrtněte prosím příslušnou odpověď):</td></tr>
</table>

5	Vynikající	Nepatrný šum,
4	Dobrá	Patrný, ale nerušicí šum, srozumitelnost bez potíže,
3	Průměrná	Je rozumět ale s potížemi,
2	Nízká	Velmi obtížná srozumitelnost,
1	Špatná	Reč je nesrozumitelná.

C. Subjective test results

C.1. The assessment of subject number 1.

Segment	Assessment			Test Number
	Score	Quality	Impairment	
Segment 1 (16 bit PCM, 44.1 kHz project rate)	3	Fair	Slightly annoying	1
	2	Poor	Annoying	2
	1	Bad	Very annoying	3
	3	Fair	Slightly annoying	4
	1	Bad	Very annoying	5
	1	Bad	Very annoying	6

C.2. The assessment of subject number 2.

Segment	Assessment			Test Number
	Score	Quality	Impairment	
Segment 1 (16 bit PCM, 44.1 kHz project rate)	4	Good	Perceptible but not annoying	1
	2	Poor	Annoying	2
	1	Bad	Very annoying	3
	3	Fair	Slightly annoying	4
	1	Bad	Very annoying	5
	1	Bad	Very annoying	6

C.3. The assessment of subject number 3.

Segment	Assessment			Test number
	Score	Quality	Impairment	
Segment 1 (16 bit PCM, 44.1 kHz project rate)	4	Good	Perceptible but not annoying	1
	3	Fair	Slightly annoying	2
	2	Poor	Annoying	3
	3	Fair	Slightly annoying	4
	2	Poor	Annoying	5
	1	Bad	Very annoying	6

C.4. The assessment of subject number 4.

Segment	Assessment			Test number
	Score	Quality	Impairment	
Segment 1 (16 bit PCM, 44.1 kHz project rate)	3	Fair	Slightly annoying	1
	2	Poor	Annoying	2
	1	Bad	Very annoying	3
	3	Fair	Slightly annoying	4
	1	Bad	Very annoying	5
	1	Bad	Very annoying	6

C.5. The assessment of subject number 5.

Segment	Assessment			Test Number
	Score	Quality	Impairment	
Segment 1 (16 bit PCM, 44.1 kHz project rate)	3	Fair	Slightly annoying	1
	2	Poor	Annoying	2
	1	Bad	Very annoying	3
	2	Poor	Annoying	4
	1	Bad	Very annoying	5
	1	Bad	Very annoying	6

C.6. The assessment of subject number 6.

Segment	Assessment			Test number
	Score	Quality	Impairment	
Segment 1 (16 bit PCM, 44.1 kHz project rate)	5	Excellent	Imperceptible	1
	2	Poor	Annoying	2
	1	Bad	Very annoying	3
	4	Good	Perceptible but not annoying	4
	2	Poor	Annoying	5
	1	Bad	Very annoying	6

C.7. The assessment of subject number 7.

Segment	Assessment			Test number
	Score	Quality	Impairment	
Segment 1 (16 bit PCM, 44.1 kHz project rate)	4	Good	Perceptible but not annoying	1
	2	Poor	Annoying	2
	1	Bad	Very annoying	3
	3	Fair	Slightly annoying	4
	1	Bad	Very annoying	5
	1	Bad	Very annoying	6

C.8. The assessment of subject number 8.

Segment	Assessment			Test number
	Score	Quality	Impairment	
Segment 1 (16 bit PCM, 44.1 kHz project rate)	3	Fair	Slightly annoying	1
	2	Poor	Annoying	2
	1	Bad	Very annoying	3
	2	Poor	Annoying	4
	3	Fair	Slightly annoying	5
	1	Bad	Very annoying	6

C.9. The assessment of subject number 9.

Segment	Assessment			Test Number
	Score	Quality	Impairment	
Segment 1 (16 bit PCM, 44.1 kHz project rate)	5	Excellent	Imperceptible	1
	3	Fair	Slightly annoying	2
	2	Poor	Annoying	3
	5	Excellent	Imperceptible	4
	2	Poor	Annoying	5
	1	Bad	Very annoying	6

C.10. The assessment of subject number 10.

Segment	Assessment			Test Number
	Score	Quality	Impairment	
Segment 1 (16 bit PCM, 44.1 kHz project rate)	3	Fair	Slightly annoying	1
	2	Poor	Annoying	2
	1	Bad	Very annoying	3
	4	Good	Perceptible but not annoying	4
	2	Poor	Annoying	5
	1	Bad	Very annoying	6

D. TCP header in details.

<table>
<tr><th>Bit offset</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th><th>15</th><th>16</th><th>17</th><th>18</th><th>19</th><th>20</th><th>21</th><th>22</th><th>23</th><th>24</th><th>25</th><th>26</th><th>27</th><th>28</th><th>29</th><th>30</th><th>31</th></tr>
<tr><td>0</td><td colspan="16">Source port</td><td colspan="16">Destination port</td></tr>
<tr><td>32</td><td colspan="32">Sequence number</td></tr>
<tr><td>64</td><td colspan="32">Acknowledgment number</td></tr>
<tr><td>96</td><td colspan="4">Data offset</td><td colspan="4">Reserved</td><td>CWR</td><td>ECE</td><td>URG</td><td>ACK</td><td>PSH</td><td>RST</td><td>SYN</td><td>FIN</td><td colspan="16">Window Size</td></tr>
<tr><td>128</td><td colspan="16">Checksum</td><td colspan="16">Urgent pointer</td></tr>
<tr><td>160
...</td><td colspan="32">Options (if Data Offset > 5)
...</td></tr>
</table>