Military Technical College
Kobry El-Kobbah,
Cairo, Egypt

5th International Conference
on Electrical Engineering
ICEENG 2006

Region of interest-based medical image compression with application to MRI brain image

*Ashraf D. Elbayoumy** *Simon J. Shepherd**

Abstract:

VoIP represents the future of digital voice communications and many carriers are preparing for the VoIP revolution. However, a number of outstanding issues need to be settled. The most important are security, compression, packet size optimization, quality of service and performance in heterogeneous networks. We have addressed all of these issues [1,2,3,4] and here we summarise our key findings in each of these areas.

Keywords:

VoIP, QoS, TEA, AMR

* Advanced Signals Laboratory, School of Engineering Design & Technology, University of Bradford, BD7 1DP, UK, ademahmo, S.J.Shepherd @bradford.ac.uk

1. Introduction:

Voice over IP– the transmission of voice over packetswitched IP networks – is one of the most important emerging trends in telecommunications. As with many new technologies, VoIP introduces both security risks and opportunities. VoIP has a very different architecture than traditional circuit-based telephony, and these differences result in significant security issues. Lower cost and greater flexibility are among the promises of VoIP for the enterprise, but VoIP should not be installed without careful consideration of the security problems introduced.

In recent work [1], it has been shown that the cryptoengine is a severe bottleneck in the Voice over IP network. Our results showed that the computationally lighter algorithms achieved better throughput than the more expensive ones. We presented the Tiny Encryption Algorithm (TEA) as a fast and secure algorithm for securing VoIP.

The incorporation of TEA or some other low complexity encryption algorithm could help alleviate the bottleneck, but this is not a scalable solution because it does not address the principal cause of delay. One of the most critical aspects in transmitting voice over IP networks is the increased packet size stemming form the encryption process.

A solution targets this problem is to compress the internal header of a packet down to approximately four bytes. This is possible because much of the data in the internal headers of a packet remains constant or is duplicated in the outer header. One thing must be considered when using this technique is the tremendous strain put on end-point CPU's as opposed to the crypto-engine. The endpoint CPU may be computationally slow (in the case of a simple VoIP with a low-cost embedded processor) or may be performing many more operations than just VoIP (in the case of a PC-based phone). In either case, the actual time required to perform the compression may take much longer than the time saved in the crypto-engine. So the question of the power of the endpoint CPU's is very important here.

We presented [2] a simple protocol that detects the endpoint CPU capabilities and according to this important information we choose dynamically the most suitable bitrate and decide if we will perform header compression, or not. Our results show that the end-to-end delay depends mainly on the receiving end-point. The higher capability PC's when acting as receivers, usually achieve a low average end-toend delay (65-150 ms) under low jitter conditions. The lower capability PC's achieved an average end-to-end delay from 150 ms to over 400 ms.

Quality of Service is fundamental to the operation of a VoIP network. Despite all the money VoIP can save users and the network elegance it provides, if it cannot deliver at least the same quality of call setup and voice relay functionality and voice quality as a traditional telephone network, then it will provide little added value. Unfortunately, the implementation of various security measures can degrade QoS. These complications range from delaying or blocking of call setups by firewalls to encryption-produced latency and delay variation (jitter).

However, current IP networks are based on best-effort services. They lack stringent QoS control. QoS control is an important issue in Voice over IP (VoIP) applications because of the need to meet both technical and commercial requirements.

QoS control mechanisms for VoIP should aim to make optimum use of available network/terminal resources and to minimise the effects of network impairments on voice quality. Several approaches exist to realise QoS control, but most seek to control the information flow from the audio/video sources, adaptively, in accordance with significant changes in the network. An important class of QoS control technique involves rate control (i.e. QoS control is achieved by automatically adjusting the send bit rate depending on network congestion conditions).

However, current rate control mechanisms are based largely only on network impairments such as packet loss rate or delay during congestion. The strategy is to control the sender behavior, using the network impairments, from the receiver or the network node but this may not be sufficient to provide optimum QoS, in terms of the voice quality delivered, because the control information didn’t consider the end-point CPU capability.

We proposed a new QoS control scheme [3] that combines the strengths of adaptive rate and end point CPU capability to provide a superior QoS control performance, in terms of perceived speech quality. Our results show that the new QoS control method achieved the best performance under different network congestion conditions and different end-point CPU capabilities.

In this paper the actual media is considered. Real transmissions done with several speech coders, because packet loss and end-to-end delay may be different according to the speech coder used to transmit data. Our goal is to present a comprehensive security solution to provide end-toend protection for VoIP applications running over heterogeneous networks without harming the perceived voice quality.

Finally, we present experimental results of comparing block and stream ciphers when used to secure VoIP in terms of end-to-end delay and subjective quality of perceived voice. Our [4] results show that end-to-end delay and subjective quality of perceived voice are better in case of additive stream ciphers than in block ciphers.

2. Preliminaries:

In this section we describe the problems occurred when voice is transmitted on a computer network and how they change when functions that guarantee confidentiality and authentication of the communication are introduced. The technologies involved are: VoIP: the application for digitizing, compressing, and converting voice into IP packets, and transmitting them over IP networks. Variable Rate Speech Coding (VBR), and header compression. In the following we briefly recall the basic techniques involved in each technology.

2.1. Voice over IP

In recent years, we have witnessed a growing interest in the transmission of voice using the packet-based protocols. Voice over Internet protocol (VoIP) is a rapidly growing technology that enables the transport of voice over data networks such as the public Internet. The following steps are performed:

- Digitization of the analog signal;
- Packet generation of the digital signal according to the TCP-UDP/IP protocols;
- Transmission of the packets on the network;
- Packet reception and analog signal reconstruction at the destination.

When sending voice traffic over IP networks, a number of factors contribute to overall voice quality as perceived by an end user. The factors determine voice quality include the choice of codec, echo control, packet loss, delay, delay variation (jitter), and the design of the network. Packet loss causes voice clipping and skips. Some codec algorithms can correct for some lost voice packets. Typically, only a single packet can be lost during a short period for the codec correction algorithm to be effective [2]. If the end-to-end delay becomes too long, the conversation begins to sound like two parties talking on a Citizens Band radio. A buffer in the receiving device tries to compensate for jitter (delay variation). If the delay variation exceeds the size of the jitter buffer, there will be buffer overruns at the receiving end, with the same effect as packet loss anywhere else in the transmission path.

Coding and packetization result in delays greater than users typically experience in terrestrial switched circuit networks. As we have seen, standard speech codecs are available for output coding rates in the approximate range of 64 kb/s to 5 kb/s. Generally, the lower the output rate, the more complex the codec. Packet design involves a tradeoff between payload efficiency (payload/total packet size) and packetization delay (the time required to fill the packet). For IPv4, the RTP/UDP/IP header is 40 bytes. A payload of 40 bytes would mean 50% payload efficiency. At 64 kb/s, it only takes 5 ms to accumulate 40 bytes, but at 8 kb/s it takes 40 ms to accumulate 40 bytes. A packetization delay of 40 ms is significant, and many VoIP systems use 20-ms packets despite the low payload efficiency when using low bit rate codecs.

Delay and jitter are two of the most critical factors that affect the quality of audio transmission. In real-time, if the packets are delayed too long, they have to be discarded, which would appear as loss of packets. Similarly, in the presence of jitter, it is not possible for the receiving site to play an audio packet as soon as it receives it, thus negatively affecting the quality of real-time voice communication.

Quality of Service is fundamental to the operation of a VoIP network. Unfortunately, the implementation of various security measures can degrade QoS. These complications range from delaying or blocking of call setups by firewalls to encryption-produced latency and delay variation (jitter). QoS issues are central to VoIP security. If QoS were assured, then most of the same security measures currently implemented in today’s data networks could be used in VoIP networks. But because of the time-critical nature of VoIP, and its low tolerance for disruption and packet loss, many security measures implemented in traditional data networks just are not applicable to VoIP in their current form.

2.2. VARIABLE RATE SPEECH CODING (VBR)

Adapting packet size to network conditions can reduce packet loss and end-to-end delay, which improve the speech quality. In a VoIP system the rate of the single speech sources is dynamically adapted to the workload conditions. The speech codecs adopted in H.323 for VoIP are G.711 (PCM), G.723.1 (MP-MLQ & ACELP), G.729 (CSACELP), and GSM-FR. These codecs have fixed bit rates so they cannot adapt themselves flexibly to the network condition that may change very rapidly. Recent papers have given various examples of variable rate coders, the most significant being the Adaptive Multi-Rate (AMR) coder, which was devised by ETSI for the third-generation (3G) mobile system.

The AMR speech codec is based on the algebraic code excited linear prediction (ACELP) algorithm, and consists of a multi-rate speech codec, a source controlled rate

(SCR) scheme including a voice activity detector (VAD) and a comfort noise generation system. The multi-rate speech codec is a single integrated speech codec with 8 source rates from 4.75 kbps to 12.2 kbps. The speech codec is capable of switching its bit rate every 20 ms length of speech frame depending upon the channel conditions. Mode switching can occur any time (frame-based). Thus, the AMR codec is well suited to rate control.

The adaptive rate QoS control scheme is shown in Fig. 1. In the scheme, the send rate of the AMR codec is adjusted in accordance with the network conditions to achieve the best possible QoS. The bit rate control mechanism is based on individual network parameters (e.g. packet loss rate and delay) or on the predicted speech quality (e.g. MOS score). The bit rate control module is used to adapt the send bit rate in accordance to the feedback information. The basic idea is that the AMR codec can reduce its bit-rate (if possible) when there is network congestion and increase its bit-rate when no congestion is detected [5].

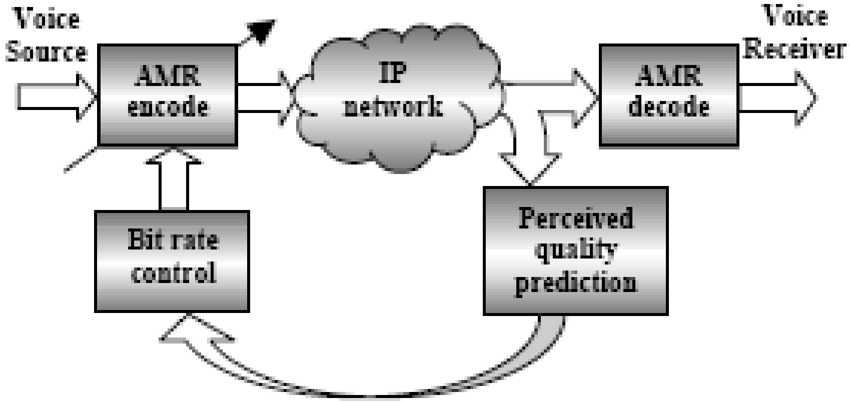


Figure (1): *Rate-adaptive QoS control scheme*

However, current rate control mechanisms are based largely only on network impairments such as packet loss rate or delay during congestion. The strategy is to control the sender behavior, using the network impairments, from the receiver or the network node but that may not be sufficient to provide optimum QoS. A second important class of QoS control technique exploits knowledge of the end-point CPU capabilities will be used to improve the voice quality delivered.

2.3. Header Compression

One of the most critical aspects in transmitting voice over networks that implement security protocols is the increase in packet size stemming from the encryption process.

The compression of packet headers results in bandwidth usage comparable to that of plain IP. This in turn, results in considerably less jitter and latency. The crypto-engine performance also improves.

The basic idea behind header compression is based on the observation that most of the fields belonging to the internal headers of a packet, i.e., IP original header, UDP and RTP headers carry values that either remain constant over the entire life of the connection, or change but the second order difference, i.e., the difference of consecutive differences computed on packet parameters is zero. Furthermore, some other fields contain information already present in the external headers.

There are a number of different methods of header compression that are defined [6,7], but the general principles of operation are very similar and essentially comprise the following elements:

- The full header is sent with the first datagram of the communication and stored by the receiver;
- Each field can be classified as UNCHANGING, RANDOM changes, DELTA changes or inferred as DEFAULT;
- The only segments of header information that need to be sent in every header are fields that change often and randomly, such as checksums or authentication data;
- For fields that are incremented from the previous value (DELTA), only the delta increment is sent.

The main problem with the header compression scheme is related to the occurrence of transmission errors as they can be detected by the CRC or by the checksum value in the IP/UDP headers, or because of packet loss. Because of the real-time nature of voice transmission, no error correction is possible and packets are lost. In this case a resynchronization process is required.

3. PROPOSED ADAPTIVE QOS CONTROL SCHEME:

As shown in Fig. 2, the proposed adaptive QoS control scheme is based on the AMR speech codec. The bit rate control module is used to adapt the send bit rate in accordance to the feedback information. The feedback can be expressed as end-to-end delay or loss rate at the receiver. However, the loss of a single packet, or of batches of packets, is an indication of severe network congestion: any action taken following this notification may be belated, thus leading to a prolonged congestion. It is better to rely on the end-to-end delay as feedback information because a delay increase is often followed by packet loss within the next few round-trip times. Therefore, being able to detect changes in the delay measure patterns, and acting on such indication, can prevent packet loss.

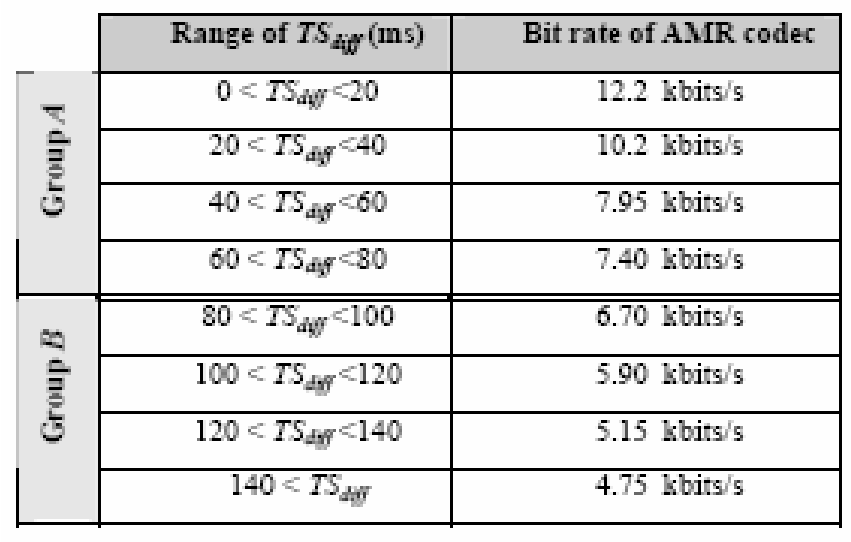
To determine the bit rate of an AMR codec, when applied to VoIP system, the network conditions should be monitored. As a measure of monitoring network conditions, in this paper, we define a parameter of TSdiff as given in eq. (1).

$$\text{TSdiff} = \text{timestamp}_i - \text{timestamp}_{i-1} - \text{Tsfixed} \qquad (1)$$

Where i and TSfixed, respectively, denote a frame number and increments of timestamp that is determined by the analysis frame size of the speech signal in the transmitting side. By computing eq. (1) in the receiving side, transmission delay of data packets, i.e. network conditions, can be estimated.

To apply an AMR codec to VoIP systems we need a table for bit-rate assignment depending on TSdiff values. Considering the allowable network delay for voice traffic as 150 ms, we simply divided TSdiff values linearly into 8 states as shown in table 1.

Table (1): *Relationship between values of TSdiff and assigned bit rate of an AMR codec*



	Range of TS_{diff} (ms)	Bit rate of AMR codec
Group *A*	$0 < TS_{diff} < 20$	12.2 kbits/s
	$20 < TS_{diff} < 40$	10.2 kbits/s
	$40 < TS_{diff} < 60$	7.95 kbits/s
	$60 < TS_{diff} < 80$	7.40 kbits/s
Group *B*	$80 < TS_{diff} < 100$	6.70 kbits/s
	$100 < TS_{diff} < 120$	5.90 kbits/s
	$120 < TS_{diff} < 140$	5.15 kbits/s
	$140 < TS_{diff}$	4.75 kbits/s

We divide the 8 states shown in Table 1 into two groups; Group A (the upper four states, Group B (the lower four states). According to the CPU capability, we decide which group will be available for the AMR speech codec to be used. If the endpoint CPU is computationally slow Group A is used. Switching from one bitrate to another in the same group could be every 20ms (length of speech frame) according to channel conditions but changing from one group to another needs 5 minutes to receive a new value from the CPU capability detector. If Group A is used that means the endpoint CPU is computationally slow so the header compression will not be performed. The header compression performed only when Group B is used. The reason that we detect the CPU capability every 5 minutes that the CPU may be performing many more operations than just VoIP (in the case of a PC-based phone) so sometimes it will be busy.

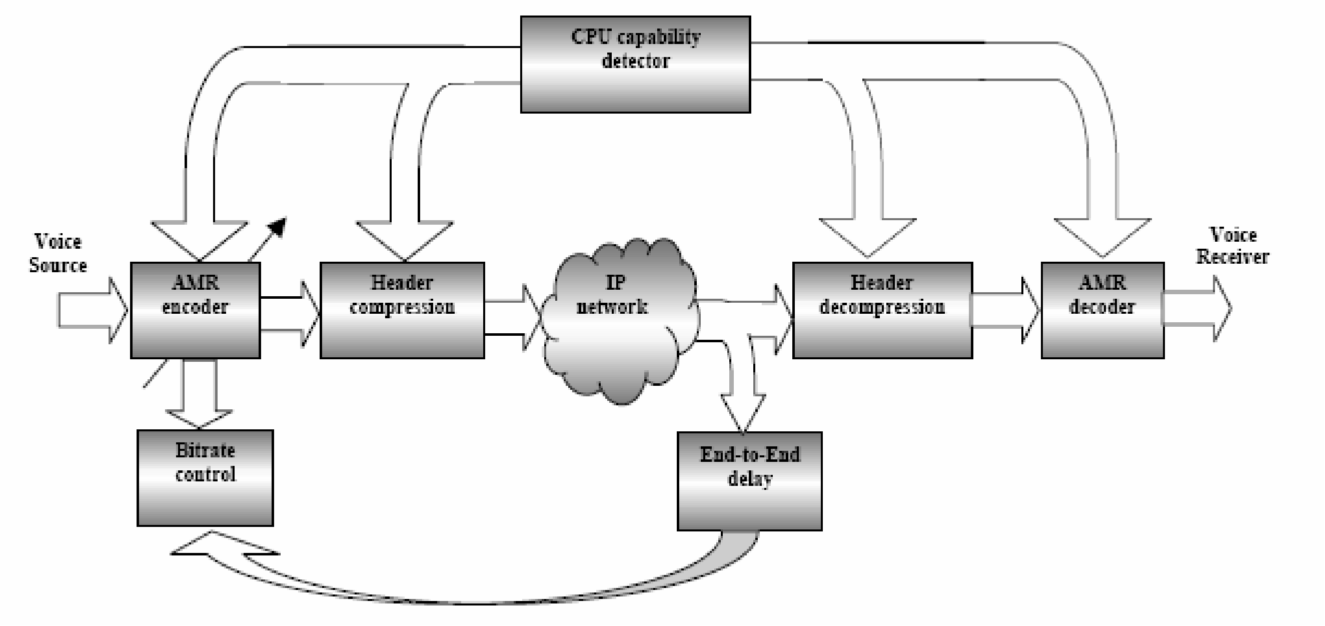


Figure (2): *Proposed adaptive QoS Control Scheme*

4. EXPERIMENTAL ENVIRONMENT:

An active VoIP QoS application measurement was performed with visual C++ software that transmits and receives full duplex VoIP streams between two hosts connected via an IP network. The program reads and encapsulates audio packets from the microphone of Host 1 and sends the packets to a remote Host 2. The two hosts have different CPU capabilities and are synchronized using NTP (Network Timing Protocol) because time synchronization provides receivers with precise information about end-to-end delays [8].

Packets are time-stamped and sequence-numbered so that various path criteria such as latency, jitter, packet loss, out of order packets, can be calculated.

5. EXPERIMENTAL RESULTS:

A. Crypto-engine:

In order to measure the maximum encoding rate, when different cryptographic algorithms are used to encrypt the packet payload, we performed the following experiment. We considered the cryptographic algorithms DES, 3DES, IDEA, TEA (all implemented in software) and for each case we generated 4 packet flows with packets of size 60, 100, 250, 1000 bytes, respectively. Each flow starts from 0 pps and increases its rate of 25 pps every 30 s in order to saturate the crypto-engine. Fig. 3 graphs the measured throughput as a function of the global traffic flow.

The straight line is the throughput for transmission of packets in the clear; therefore it increases linearly with traffic. The figure shows that when encryption is performed, throughput levels off or decreases after reaching a maximum value, which depends on the algorithm. It also shows that longer packets significantly improve the crypto-engine performance. TEA, then IDEA, DES achieves the best performance, and the last is 3DES.

In order to investigate how the proposed QoS control scheme affect perceived speech quality under different network conditions and different endpoint CPU's capabilities, we performed the following experiments with real voice traffic instead of a synthetic flow of packets using our visual C++ program. We considered a TEA encrypted phone call between two PC's with different CPU's capabilities, computer A is P4 2GHz in Bradford, UK, and computer B is P2 500MHz in Cairo, Egypt. Both PC's have Windows 2000 as an operating system. The interarrival time between consecutive packets is plotted in Fig. 4. The spikes in Fig. 4 are due to late packets. For Fig. 4 (a) (b) (c) packets are sent from computer A to computer B.

As shown in Fig. 4(c) adapting to both network conditions and end-point CPU capability reduce the number of spikes that means the number of late packets decreased hence improving the performance.

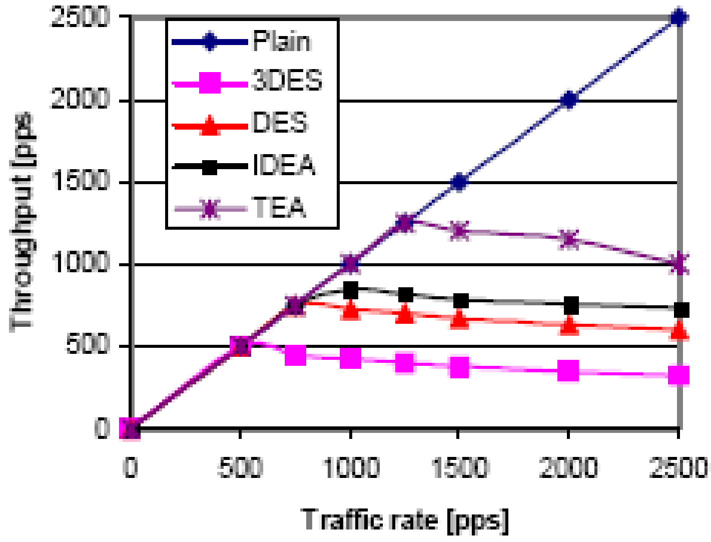


Figure (3): *Throughput of the crypto-engine in pps as a function of linearly increasing traffic in pps for plain and encrypted traffic*

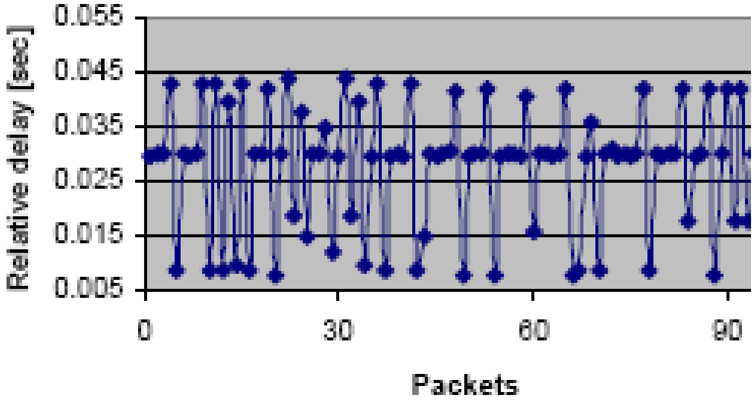


Figure (4-a): *Individual packet interarrival time for voice traffic without considering any adaptation*

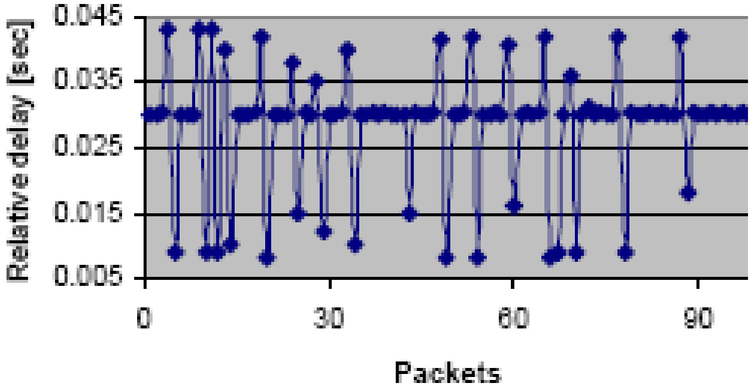


Figure (4-b): *Individual packet interarrival time for voice traffic when adapting to network conditions only*

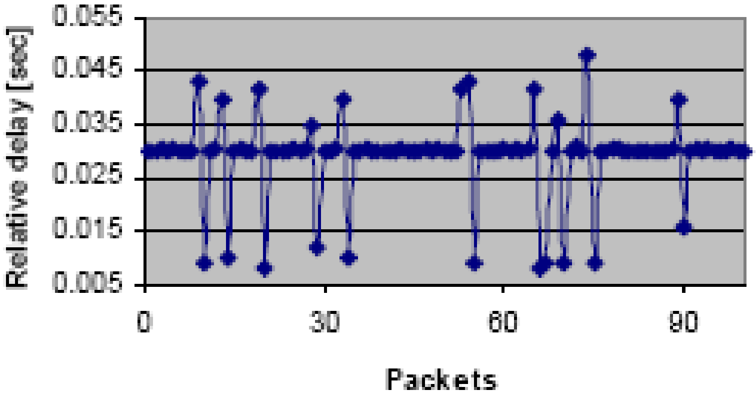
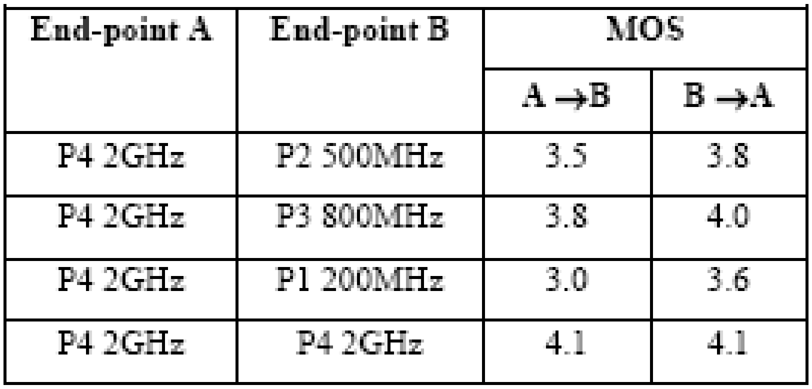


Figure (4-c): *Individual packet interarrival time for voice traffic when adapting to both network conditions and end-point CPU capability*

In order to investigate the effect of different end-point CPU's capabilities on the received VoIP quality, we implement the mean opinion score (MOS) test. In voice communications, particularly Internet telephony, MOS provides a numerical measure of the quality of human speech at the destination end of the circuit. The scheme uses subjective tests (opinionated scores) that are mathematically averaged to obtain a quantitative indicator of the system performance. It ranges from 1 to 5, 1 being the worst case.

All MOS values shown in this paper result from a large number of experiments, conducted under the same conditions. In most cases the results from 10 or 20 repeated experiments are averaged out to get a reliable value. In all cases the standard deviation for these series of repeated measurements is about 0.1 MOS. As show in Table 2, we achieved a better MOS when the higher capability PC acts as a receiver that because the end- to-end delay depends mainly on the receiving end-point.

Table (2): *The MOS values for different end-point CPU's capabilities*



End-point A	End-point B	MOS	
		A →B	B →A
P4 2GHz	P2 500MHz	3.5	3.8
P4 2GHz	P3 800MHz	3.8	4.0
P4 2GHz	P1 200MHz	3.0	3.6
P4 2GHz	P4 2GHz	4.1	4.1

C. Packet Loss:

To determine the influence of packet loss on the perceived quality for our proposed adaptive QoS control scheme a number of experiments have been conducted, where amounts of packet loss of 0%, 1%, 2%, 5% and 10% were introduced successively. The MOS values resulting from these measurements are shown in Fig. 5.

As shown in Fig. 5 adapting to both network conditions and end-point CPU capability improves the packet loss rate which impact on voice quality, compared to adapting to network conditions only.

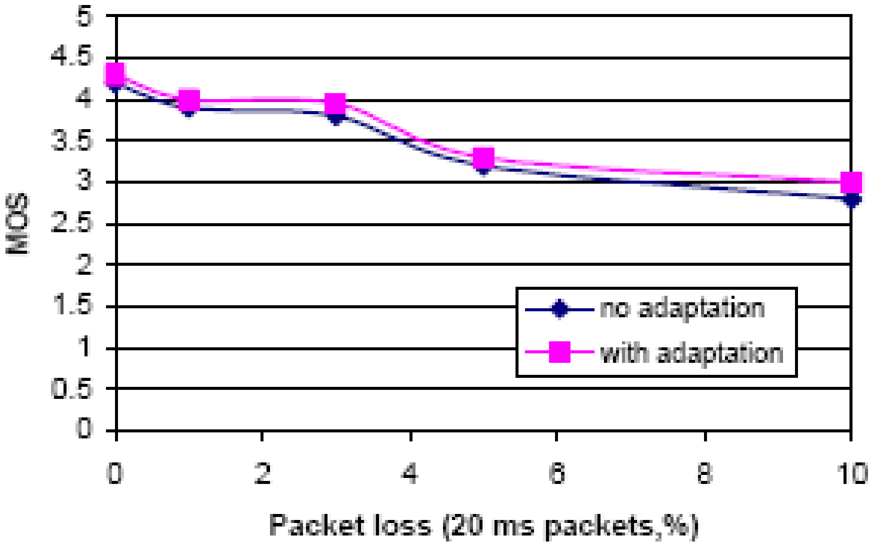


Figure (5): *The effect of packet loss on the perceived quality*

D. Block or stream ciphers for securing VoIP?

In order to compare between block and stream ciphers when used to secure VoIP, all packets are encrypted using the cryptographically powerful but computationally efficient AES cipher running in block cipher mode then in stream cipher mode.

In a stream cipher mode the encryption is done by bitwise XORing the payload of the packet with a generated keystream segment. This is called an additive stream cipher.

AES in counter mode acts as a keystream generator producing a pseudo-random keystream of arbitrary length that is applied in a bit-wise fashion to the RTP payload by means of a logical XOR function, thus working as a classical stream cipher. AES itself is a block cipher with a block size of 128 bits and a key size of 128, 192, or 256 bits. In order to work as a pseudo-random generator AES is loaded at the start of each RTP/RTCP packet with a distinct initialisation vector (IV) that is derived by hashing a 112-bit salt_key, the synchronisation source identifier (SSRC) of the media stream, and the packet index. Encrypting this IV results in an output of 128 pseudo-random bits.

Next the IV is incremented by one and again encrypted, thus generating the next 128 bits of the keystream. By counting the IV up by increments of one as many keystream blocks can be generated as are required to encrypt the whole RTP/RTPC payload.

Our objective and subjective tests show that, the end-toend delay and subjective quality of perceived voice are better in case of stream ciphers than in block ciphers because stream cipher mode has the big advantage that the keystream can be precomputed before the payload becomes available thus minimizing the delay introduced by encryption. And ofcource by using a stream cipher instead of block cipher there is no need to pad the payload up to multiple of the block size which would add 15 overheaded bytes to the RTP packet in the worst case.

One thing must be considered when using stream ciphers that additive stream ciphers do not provide any security service other than privacy. In particular, they do not provide message integrity. If message integrity is required, it can be provided through the use of an authentication transform. Such a transform SHOULD be used.

An additive stream cipher is vulnerable to attacks that use statistical knowledge about the plaintext source to enable key collision and time-memory tradeoff attacks [9]. These attacks take advantage of commonalities among plaintexts, and provide a way for a crypto- analyst to amortize the computational effort of decryption over many keys, thus reducing the effective key size of the cipher. Simply increasing the size of the keys used can provide protection against such attacks. We encourage the use of keys that are as large as possible, and note that in many cases increasing the key size of a cipher does not affect the throughput of a cipher.

Block cipher encryption has the advantage that it is not as vulnerable to typical plaintext attacks as is encryption with an additive stream cipher. This is because block cipher encryption (CBC mode) is randomized through the use of an unpredictable IV. However, additive stream cipher encryption can achieve the same level of security as CBC mode encryption through an increase in key size [9]. This strategy of "putting all of the randomization in the key" provides an encryption method that can be as secure as CBC, while providing the advantages of using additive stream cipher outlined above.

The debate over the relative merits of block vs. stream ciphers for VoIP will no doubt be an ongoing matter. Shamir [10] has long predicted the death of stream ciphers, but current research such as ours into the engineering practicalities of secure VoIP suggest otherwise. As with many practical situations, not everything is black and white. Block ciphers have a place and so do stream cipher. Our research points to an optimum compromise that may give the best of both worlds.

6. Conclusion:

VoIP will play a key role in the future of digital voice communications. Our research has covered the four main critical issues facing VoIP carriers and our findings are as follows:

Encryption algorithm ~ lightweight encryption algorithms are superior to heavier ones. Conventional algorithms such as the DES, which were originally designed for hardware, are very computationally demanding when implemented in software – even for fast, modern CPUs. The newer generation of fast, software-optimised algorithms such as SEAL, TEA and so on are vastly superior while offering just as high a level of security. Our research has also indicated that stream ciphers with resynchronization capability are preferable to block algorithms for securing VoIP.

Quality of Service ~ we have introduced a new end-to-end QoS algorithm based on end-point CPU capability detection that decides dynamically on optimum packet size, whether to perform such operations as header compression, and so forth. This has gone a long way towards mitigating the effects of having equipment with significantly differing capability at each end of the link, especially in light of the non-obvious result that the receiving end-point is the dominant factor in the overall link capability.

We believe that this work has gone a long way towardsresolving the outstanding issues facing the successful commercial implementation of secure VoIP.

References:

[1] A. Elbayoumy, S. Shepherd, “A High grade secure VoIP system using The Tiny Encryption Algorithm”, *Proceedings of* 7th Annual International Symposium on Advanced Radio Technologies, Boulder, Colorado, 1-3 March 2005.

[2] A. Elbayoumy, S. Shepherd, "QoS Control Using an Endpoint CPU Capability Detector in a Secure VoIP System", *Proceedings of 10th IEEE Symposium on Computers and Communications*, La Magna del Mar Menor, Spain, 27-30 June 2005.

[3] Using an Endpoint CPU Capability Detector ", *Proceedings of ITA05 International conference on Internet Technologies and applications* Wrexham, North Wales, UK, Wednesday 7th - Friday 9th September 2005.

[4] A. Elbayoumy, S. Shepherd, “Stream or Block Ciphers for Securing VoIP?”, Advanced Signals Laboratory, University of Bradford, UK, 2005, Pre-print.

[5] Jeong Wook Seo, Se Jeong Woo, and Keun Sung Bae, “A Study on the Application of an AMR Speech Codec to VoIP”, IEEE, 2001.

[6] R. Barbieri, D. Bruschi, E. Rosti, “Voice over IPsec: Analysis and Solutions”, Computer Security Applications Conference, 2002. Proceedings. 18th Annual, 9-13 Dec. 2002.

[7] D. Nguyen, J. Lequang, “cRTP Performance Enhancement”, Cisco System [ENG 102721].

[8] H. Melvin, L. Murphy, “Time Synchronization for VoIP Quality of Service”, IEEE Internet Computing, June 2002.

[9] Implications on Internet Security”, Seventh Annual Workshop on Selected Areas in Cryptography , 2000.

[10] http://www.iris.re.kr/ac04/data/Asiacrypt2004/Invited%20Talk%201_Adi%20Shamir.pdf