

171050

T.C.



GEBZE YÜKSEK TEKNOLOJİ ENSTİTÜSÜ
MÜHENDİSLİK VE FEN BİLİMLERİ ENSTİTÜSÜ

H323 PROTOKOLÜNÜN İNCELENMESİ VE
AÇIK KAYNAK KODLU OPENH323
KÜTÜPHANESİ KULLANILARAK VOIP
İSTEMCİ YAZILIMI GELİŞTİRİLMESİ

Mesut TOMRUK

YÜKSEK LİSANS TEZİ

ELEKTRONİK MÜHENDİSLİĞİ

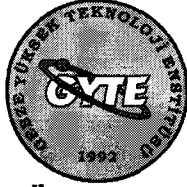
ANABİLİM DALI

TEZ DANIŞMANI

Yrd. Doç. Dr. Oğuz KUCUR

GEBZE

2005



**GEBZE YÜKSEK TEKNOLOJİ
ENSTİTÜSÜ**

YÜKSEK LİSANS JÜRİ ONAY FORMU



Mesut Tomruk'un tez çalışması, G.Y.T.E. Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 20.06.2005 Tarih ve 2005/22 sayılı kararıyla oluşturulan jüri tarafından Elektronik Anabilim Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir.

JÜRİ

ÜYE

(TEZ DANIŞMANI) : Yrd. Doç. Dr. Oğuz Kucur

ÜYE

: Yrd. Doç. Dr. Serdar Süer Erdem

ÜYE

: Yrd. Doç. Dr. İlyas Kandemir

ONAY

G.Y.T.E. Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 25.08.2005 Tarih ve 2005/28 sayılı kararı.



ÖZET



Bu çalışmada, Voice Over IP (VoIP) sistemlerinde kullanılan H323 protokolu incelenmiştir. Ayrıca açık kaynak kod geliştirme topluluğu tarafından geliştirilmiş olan openH323 yazılım kütüphanesi kullanılarak bir Voice Over IP istemci yazılım uygulaması geliştirilmiştir. VoIP bir çok telekomünikasyon firması tarafından geleneksel devre temelli telefon görüşmelerinin maliyetlerini düşürmek amacıyla kullanılmaktadır. H323, VoIP sistemlerinde kullanılmak üzere geliştirilmiş bir şemsiye protokoldür ve VoIP haberleşmesi için alt protokoller içerir. İstemci yazılımı geliştirilirken Microsoft Visual C++ 6.0 ve Microsoft Visual Basic 6.0 yazılım geliştirme araçları ve Active Template Library teknolojisi kullanılmıştır. Bu çalışmada H.323 protokolünün temelleri incelenmiş ve openH323 yazılım kütüphanesi ile birlikte MS Visual C++ 6.0 aracı ve Active T. Library yazılım teknolojisi kullanılarak VoIP yazılımı geliştirilmiştir. Geliştirilen yazılım ses iletimi ve band genişliği bakımından test edilmiş ve Speak Freely yazılımının değerlerine yakın değerler elde edilmiştir. Bu yazılım bu konuda ileride yapılabilecek çalışmalarda kullanılabilir.

Anahtar Kelimeler: H323, openH323, VoIP.



SUMMARY

In this work, H323 protocol which is used in Voice Over IP (VoIP) systems is investigated and a VoIP client software is developed by using openH323 software library, which is a software library was developed by open source community for H323 applications. VoIP is an emerging technology which is being adopted by many telecommunication companies to lower the high costs of circuit based telephone conversations. H323 is an umbrella protocol which provides several sub protocols for VoIP communications. Microsoft Visual C++ 6.0 and Microsoft Visual Basic 6.0 development tools and Active Template Library technology were used in develeopment of VoIP client software. The basics of H.323 have been studied and a VoIP software has been developed by using openH323 software library and MS Visual C++ 6.0 tool with Active T. Library software technology. This VoIP software has been tested in terms of voice transmission and bandwidth occupancy, and test results have been very close to those of Speak Freely software. This VoIP software can be used in future studies in this area.

Keywords: H323, openH323, VoIP.

TEŞEKKÜR



Yüksek Lisans eğitimim boyunca yardım ve katkılarını esirgemeyen değerli hocalarım ve danışmanlarım, Yrd. Doç. Dr. Oğuz Kucur'a ve Yrd. Doç. Dr. Serdar Süer Erdem'e teşekkürleri bir borç bilirim.

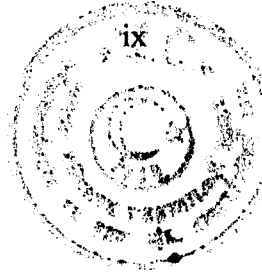


İÇİNDEKİLER DİZİNİ

	<u>Sayfa</u>
ÖZET	iv
SUMMARY	v
TEŞEKKÜR	vi
İÇİNDEKİLER DİZİNİ	vii
KISALTMALAR DİZİNİ	ix
ŞEKİLLER DİZİNİ	xi
TABLolar DİZİNİ	xii
1. GİRİŞ	1
1.1. Çalışmanın Amacı ve Katkısı	3
1.2. Tez'in Bölümleri	3
2. VOIP'E GENEL BAKIŞ	4
2.1 VoIP'in Uygulama Alanları	4
2.2 VoIP Uygulamalarında Önemli Konular	4
3. H.323	6
3.1 H.323 Elemanları	6
3.2 H.323 Alanı	8
3.3 H.323 Protokolleri	8
3.4 H.323 Terminal Özellikleri	11
3.5 H.323 Gateway Özellikleri	12
3.6 H.323 Gatekeeper Özellikleri	13
3.7 H.323 Çağrısı Oluşturulması	14
4. SIP	16
4.1. SIP Bileşenleri	16
4.1.1. Kullanıcılar	16
4.1.2. Ağ Sunucuları	17
4.2. SIP Mesajları	17
4.3. SIP Operasyonu	17
4.4. SIP ve H.323'ün Karşılaştırılması	19
5. GELİŞTİRİLEN İSTEMCİ YAZILIMI	21

5.1 Yazılımın Bileşenleri	21
5.2 Yapılan Testler	27
6. SONUÇLAR	32
KAYNAKLAR	33
ÖZGEÇMİŞ	34





KISALTMALAR DİZİNİ

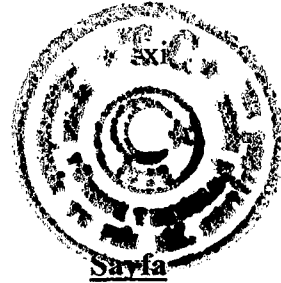
Kısaltma

Açıklama

ACF:	Admission Confirm
ADPCM:	Adaptive Differential PCM
ARJ:	Admission Reject
ARQ:	Admission Request
ATL:	Active Template Library
ATM:	Asynchronus Transfer Mode
BCF:	Bandwidth Confirm
BRJ:	Bandwidth Reject
BRQ:	Bandwidth Request
COM:	Component Object Model
CS-ACELP:	Conjugate Structure Algebraic CELP
DLL:	Dynamic Link Library
IETF:	Internet Engineering Task Force
IP :	Internet Protocol
IPX:	Internet Packet Exchange
ISDN:	Integrated Services Digital Network
ITU:	International Telecommunications Union
LAN:	Local Area Network
LD-CELP:	Low Delay Code Excited Linear Prediction
MC:	Multipoint Controller
MCU:	Multipoint Control Unit
MFC:	Microsoft Foundation Classes
MP:	Multipoint Processor
MPOA:	Multiprotocol Over ATM
PC:	Personal Computer
PCM:	Pulse Code Modulation



PSTN:	Public Switched Telephone Network
QoS	Quality of Services
RAS:	Registration, Admission, Status
RMOA:	Realtime Multimedia Over ATM
RSVP	Resource ReSerVation Protocol
RTP :	Real Time Transfer Protocol
RTCP :	Real Time Control Protocol
SCN:	Switched Circuit Network
SIP:	Session Initiation Protocol
SSL:	Secure Socket Layer
TCP:	Transport Control Poptocol
UAC:	User Agent Client
UAS:	User Agent Server
UDP:	User Datagram Protocol
URI:	Uniform Request Identifier
URL:	Uniform Resource Locator
VTOA:	Voice and Telephony Over ATM
VoIP :	Voice Over IP
WAN:	Wide Area Network



ŞEKİLLER DİZİNİ

Sekil

Sayfa

3.1	H.323 elemanlarını içeren ağ şeması	8
3.2	Bir H.323 terminalinde bulunan protokollerin şeması	11
3.3	H.323 gateway protokol şeması örneği	12
3.4	H.323 gatekeeper protokol şeması	13
4.1	Örnek bir SIP çağrısı.	19
5.1	Yazılımı oluşturan parçalar	21
5.2	Kullanıcı arabirimi bağlantı ekranı	22
5.3	Kullanıcı arabirimi ayarlar ekranı	22
5.4	Test Bilgisayarları	27
5.5	İletilen ve Alınan Ses Dosyalarının Karşılaştırma Grafiği	28
5.6	İstemci Yazılım İçin GSM Kodek Bant Genişliği Grafiği	30
5.7	Speak Freely Yazılım İçin GSM Kodek Bant Genişliği Grafiği	31

TABLÖLAR DİZİNİ



Tablo

Sayfa

3.1	Ses kodekleri ve iletim için ihtiyaç duydukları bant genişlikleri	10
5.1	Ses İletimine Ait Karesel Hata Hesaplam Sonuçları Tablosu	28



1. GİRİŞ

İnternet giderek artan bir şekilde insanların iletişim faaliyetlerindeki yerini artırmaktadır. Uzun bir süreden beridir yazılı haberleşmenin çok önemli bir bölümü İnternet üzerinden elektronik postalar ile yapılmaktadır, bu dönüşüm ile birlikte mektup ve geleneksel postacılık sistemleri yavaş ve maliyetli bir seçenek olarak daha az tercih edilmektedir. Benzer bir dönüşüm telefon görüşmeleri için de yaşanmaktadır. Geleneksel devre temelli telefon görüşmelerinin yerini paket temelli bir haberleşme altyapısı kullanan İnternet üzerinden iletim yapan hizmetler almaktadır. Böylece insanların birbiri ile sesli görüşme yapabilmeleri için çok daha düşük maliyetli hizmetler sunulabilmektedir. Maliyet düşürücü yeni yöntemler ve araçlar rekabetçi piyasa koşulları yüzünden daha fazla tercih edilir duruma gelmektedir.

VoIP hizmetleri sunan firmalar giderek artmaktadır. Örneğin Cisco gibi ağ ekipmanı üreten firmalar müşterilerinin VoIP için kullanabilecekleri donanımları satmakta. Türkiye’de de yeni operatörler Türkiye ile yurtdışı arasında düşük fiyatlı telefon görüşmeleri yapma imkanı sağlamaktadır. Bilgisayar üzerinde çalışan ve ücretsiz olan Skype yazılımı giderek yaygınlık kazanmaktadır [Link5].

IP (Internet Protocol), İnternetin üzerine inşa edildiği paket temelli haberleşmeyi sağlayan bir protokoldür. Böylece bir iletişim devresi birden fazla kişi/cihaz tarafından daha verimli şekilde kullanılabilir. İşte IP protokolünün maliyet düşürücü etkisi buradan gelmektedir, bu yüzden IP üzerinden ses iletimi için yöntemler geliştirilmiştir. Sesin IP üzerinden iletimine VoIP (Voice Over IP) denmektedir. Ses ve görüntünün İnternet üzerinden iletilebilmesi için bir çok konuda gelişmeye ihtiyaç duyulmuştur. Doğası gereği analog olan ses ve görüntünün sayısallaştırılması, sayısal bilgilerin daha az kaynak harcanarak saklanıp iletilebilmesi için bilgi sıkıştırma teknikleri ve bunların uygulaması olan kodekler geliştirilmiştir. IP üzerinden gerçek zamanlı ses ve görüntü transferi esnasında senkronizasyon ve kayıp paketlerin telafisi için Real Time Transfer Protocol (RTP) [Schulzrinne,1996], RTP ile birlikte bilgi akışıyla ilgili analiz ve yönetim için Real Time Control Protocol (RTCP), bant genişliği ve servis kalitesi (Quality of Services

(QoS)) için Resource ReSerVation Protocol (RSVP) protokolleri geliştirilmiştir [Braden,1997].

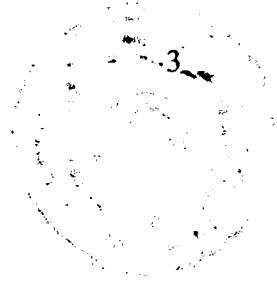
IP üzerinden ses ve görüntü iletimi esnasında iletimi başlatmak, yönetmek, çeşitli üreticilerin cihazları ve yazılımlarını birbiri ile uyumlu hale getirmek en önemli problem olarak ortaya çıkmıştır. Bu problemi çözmek için çeşitli protokoller geliştirilmiştir. Bunlardan birincisi International Telecommunications Union (ITU) tarafından geliştirilen H.323 şemsiye protokolüdür ve diğer alt protokolleri içermektedir [ITUT,1998]. İkinci protokol Internet Engineering Task Force (IETF) tarafından geliştirilmiş olan Session Initiation Protocol (SIP) [Handley,1999] protokolüdür [Mortada,2001].

Ayrıca, ATM Forum tarafından Asynchronous Transfer Mode (ATM) ağlar üzerinde çalışacak VoIP standartları belirlenmiştir. Bu standartlar şunlardır; Voice and Telephony over ATM (VTOA), Realtime Multimedia over ATM (RMOA) [Link2]. ATM ağları sanal olarak oluşturulan devreleri kullanır, amacı düşük gecikme süreleri ve yüksek iletim hızları elde etmektir [Link6]. ATM ağlarının düşük gecikme süreleri ve güvenilir iletim özellikleri VoIP uygulamaları için cazip hale gelmektedir.

Bu çalışmada VoIP ile haberleşmenin problemlerine çözüm olarak geliştirilen H.323 protokolü ve SIP protokolü incelenerek karşılaştırılır. Ayrıca H.323 protokolü ile uyumlu bir VoIP yazılımının geliştirilmesi ve kullanım testlerinin yapılması hedeflenmiştir.

OpenH323

H.323 protokokolüne uygun donanım ve yazılım geliştirmek istendiğinde hazır yazılım kütüphaneleri kullanılabilir. Bunlar içinde ticari olanlar bulunmakla birlikte ücretsiz olarak kullanılabilir açık kaynak kodlu olarak geliştirilmiş bir kütüphane openH323'tür [Link1]. Bu kütüphane ile H323 protokolü ile uyumlu olarak çalışabilecek H323 birimleri geliştirilebilir. openH32 açık kaynak kodlu bir kütüphane olduğu için bir çok geliştirici ve kullanıcı tarafından test edilerek geliştirilmektedir. openH323 kütüphanesi C++ programlama dili ile geliştirilmiştir. Bu kütüphane kullanılarak hem Windows hem de Linux işletim sistemleri için uygulama geliştirmek mümkündür.



1.1 Çalışmanın Amacı ve Katkısı

Bu tez çalışmasının amaçlarını şu başlıklar altında toplayabiliriz;

- 1- H.323 ve SIP protokolünü incelemek, karşılaştırmasını yapmak.
- 2- openH323 kütüphanesini incelemek.
- 3- MS Visual C++ yazılım geliştirme aracı, Active Template Library teknolojisini, MS Visual Basic yazılım geliştirme aracını ve openH323 kütüphanesini kullanarak bir H.323 istemcisi yazılımını gerçeklemek.
- 4- Çalışma sonucunda, ileride başka çalışmalar için kullanılabilecek bir istemci yazılımı elde etmek. Geliştirilen yazılım ile sesli H.323 görüşmesi testi yapmak.

Yazılımı geliştirilirken açık kaynak kodlu yazılımlardan faydalanılmıştır.

1.2 Tezin Bölümleri

Bu çalışmanın ikinci kısmında VoIP konusuna genel bir bakış yapıp VoIP'in uygulama alanlarına değinilmektedir. Ayrıca VoIP uygulamalarında karşılaşılan önemli konulardan bahsedilmektedir.

Üçüncü bölümde ise H.323 protokolü incelenirken H.323 protokolü ile ilgili kavramlardan ve özelliklerden bahsedilmektedir. Ayrıca bir H.323 çağrısının adımları belirtilmektedir.

Dördüncü bölümde SIP protokolü incelenip H.323 protokolü ile karşılaştırılması yapılmaktadır.

Beşinci bölümde, çalışma çerçevesinde geliştirilen H.323 protokolü ile uyumlu VoIP istemci yazılımı ile ilgili bilgiler verilmektedir. Yazılımın bileşenleri ve yapılan testler hakkında bilgi verilmiştir.

Altıncı bölümde ise çalışmanın sonuçları özetlenmiştir.



2. VoIP'E GENEL BAKIŞ

2.1 VoIP'in Uygulama Alanları

VoIP'in çeşitli uygulama alanları mevcuttur, bunların ilk akla geleni IP telefonu uygulamasıdır. IP telefonu uygulaması bilgisayarlar arasında yapılabileceği gibi PSTN şebekesi ile bilgisayarlar arasında da yapılabilir. Ayrıca IP telefonu sistemleri için özel olarak üretilmiş telefon cihazları da kullanılabilir. Diğer uygulama alanları şunlardır: telekonferans, videokonferans, uzaktan eğitim, radyo ve televizyon yayıncılığı.

2.2 VoIP Uygulamalarında Önemli Konular

Ses ve Görüntü Kalitesi

Ses ve görüntü kalitesine kullanılan sıkıştırma algoritması (codec) ve IP üzerinden bilgi paketlerinin iletimi esnasında oluşan gecikmeler etki eder. IP protokolünün yapısı gereği bilgi paketlerinin bir uçtan bir uca iletimi garanti edilememektedir. Bu etkilerin azaltılabilmesi için eko giderici algoritmalar, daha gelişkin kodekler ve Quality of Services (QoS) teknikleri kullanılabilir.

IP ağlarındaki paket gecikmeleri ve paket kayıpları VoIP için sorun yaratacak derecede etkin olursa ATM ağları ve uyumlu VoIP standartları ile daha yüksek kalitede ses iletimi mümkün olabilmektedir.

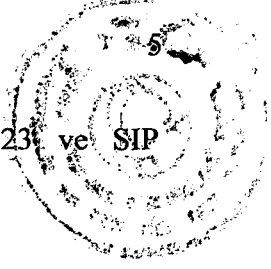
Uyumluluk:

Farklı üreticilerin ürünleri arasında uyumsuzluklar yaşanabilmektedir. Bu sorunun aşılması için sonraki bölümlerde değinilecek olan H.323 veya SIP protokollerine uygun ürün geliştirilmesi gerekmektedir.

Güvenlik:

Hem iletilen ses ve videonun hem de kullanılan sistemin kontrol mekanizmalarının kötü amaçlı kullanım ve dinleme amaçlı saldırılara karşı korunması gerekmektedir. Güvenlik için Layer 2 Tunneling yöntemi ve Secure

Socket Layer (SSL) şifreleme yöntemi kullanılabilir. Ayrıca H.323 ve SIP protokolleri de güvenlik için bazı donanımlar sağlamaktadır.



Public Switched Telephone Network (PSTN) Şebekesi İle Entegrasyon:

IP üzerinden iletilen ses trafiğinin geleneksel PSTN şebekesine entegrasyonu gerekmektedir. Bunun için çeşitli gateway ürünleri kullanılmaktadır.





3. H.323

H.323 protokolü IP tabanlı ya da Internet Packet Exchange (IPX) tabanlı yerel alan ağları (Local Area Network (LAN)) ve geniş alan ağları (Wide Area Network (WAN)) üzerinden ses, video ve data iletimi için gerekli elemanları, protokolleri ve mekanizmaları tanımlar. H.323 tabanlı bir haberleşme yalnızca ses iletimine (IP telephony) dayanabileceği gibi görüntü ve bilgi iletimini de (video telephony) mümkün kılabilir. Noktadan noktaya haberleşmenin yanısıra bir noktadan çok noktaya haberleşme biçimi de desteklenmektedir.

3.1 H.323 Elemanları

Aşağıda H.323 alanına ait elemanlar belirtilmiştir. [Trillium]

Terminaller

Terminaller son kullanıcının cihazlarıdır ve çift yönlü sesli ve/veya görüntülü haberleşme yapabilirler. Zorunlu olmamakla birlikte data haberleşmesini de destekleyebilirler. Haberleşme noktadan-noktaya veya noktadan-çok noktaya olabilmektedir. Terminal cihazı bir IP telefonu cihazı olabileceği gibi bilgisayar (Personal Computer (PC)) üzerinde çalışan bir VoIP yazılımı da olabilir.

Gateway

Gateway cihazları H.323 ağının başka tip bir ağı bağlantısını üstlenirler. Bu görevi yerine getirmek için bağlantı sağladığı iki ağın gerektirdiği protokoller arasında dönüşümü sağlarlar. Örneğin IP tabanlı H.323 ağı ile telefon ağı (PSTN) arasındaki bağlantıyı sağlayabilirler.

Gatekeeper

Her Gatekeeper sorumlu olduğu etki alanı içerisindeki bileşenler için otantikasyon ve kayıt (registration) servisini sağlar. Bileşenler; terminaller, Multipoint Control Unit (MCU)'lar ve gateway'ler olabilir. Gatekeeper kayıt ettiği bileşenler için ayrıca LAN adresleri ile IP adresleri arasında çevrim servisini de sağlar.



Ayrıca řu servisler zorunlu olmamakla birlikte gatekeeper tarafından sağlanabilmektedir; otorizasyon, bant genişlięi yönetimi, yardımcı servisler (supplementary services), katalog servisleri (directory services) ve çağrı yönetimi servisleri (call management services).

Multipoint Control Unit (MCU)

MCU konferans yapılabilmesi için gerekli desteęi sağlar. Bir MCU, Multipoint Controller (MC) ve gerektiğinde Multipoint Processor (MP) birimi/birimleri içerir.

Multipoint Controller (MC)

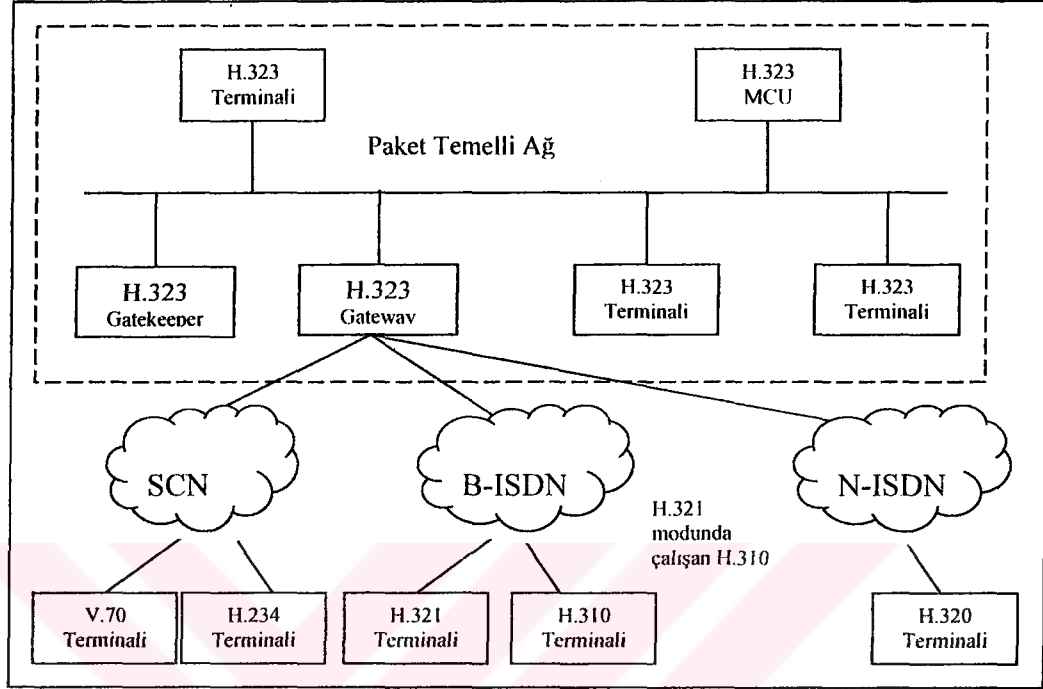
MC, H.245 protokolünü kullanarak terminaller ile haberleşir ve terminaller tarafından desteklenebilen özellikleri belirler (capability exchange mesajları ile). Ayrıca gerektiğinde konferans esnasında hangi ses ve görüntü stream'lerinin multicast yöntemi ile yayınlanacağını kontrol eder.

Multipoint Processor (MP)

Ses, görüntü ve data stream'larını işler. Gerektiğinde bunları anahtarlama (switching) ve karıştırma (mixing) işlemlerine tabi tutar.



Şekil 3.1’de H.323 elemanlarını içeren bir ağ yapısı gösterilmiştir.



Şekil 3.1 H.323 elemanlarını içeren ağ şeması

3.2 H.323 Alanı (Zone)

Bir H.323 alanı, bir gatekeeper elemanı tarafından yönetilmekte olan terminaller, gateway ve MCU’lar topluluğudur. Bir H.323 alanında yalnızca bir tane gatekeeper elemanı olabilir ve en az bir terminal elemanı bulunmalıdır.

3.3 H.323 Protokolleri

H.323 bir şemsiye protokoldür. Yani bazı protokolleri bünyesinde barındırır.
[Trillium]

H.323 şemsiyesi altında bulunan protokoller, yerine getirdiği işlevler ve zorunlu olup olmamaları açısından aşağıda listelendiği şekildedir:

Ses Kodekleri GSM (zorunlu değil)	G.711, G.722, G.728, G.723, G.729 (zorunlu)
Video Kodekleri	H.261, H.263 (Zorunlu değil)
Call Signaling	H.225 (zorunlu)
Control Signaling	H.245 (zorunlu)
Multipoint	H.323 (Zorunlu değil)
Data	T.120 (Zorunlu değil)
Security	H.235 (Zorunlu değil)
Supplementary Services	H.450 (Zorunlu değil)
CS Services	H.246 (Zorunlu değil)
Diğer protokoller	RTP, RTCP, IP, TCP, UDP
Ses kodekleri (Audio Codecs)	

Ses kodekleri ile iletilecek ses bilgisi sıkıştırılır. Böylece kısıtlı iletim bant genişliğine rağmen sesli haberleşme yapılabilmesi sağlanır. G.7xx serisi standartlar ile ses sıkıştırma ve açma işlemlerinin nasıl yapılacağı tarif edilir. Her kodekin farklı sıkıştırma oranı, algoritma komplekslik seviyesi, iletim bant genişliği ihtiyacı ve kalite karakteri vardır. Çeşitli kodeklerin ihtiyaç duydukları bant genişlikleri Tablo 3.1’de belirtilmiştir [Seydim,1999].



Tablo 3.1. Ses kodekleri ve iletim için ihtiyaç duydukları bant genişlikleri

Kodek	Açıklama	Bant genişliği
G.711	Pulse Code Modulation (PCM)	64 kb/s
G.726	Adaptive Differential PCM (ADPCM)	16,24,32,40 kb/s
G.728	Low Delay Code Excited Linear Prediction (LD-CELP)	16 kb/s
G.729	Conjugate Structure Algebraic CELP (CS-ACELP)	8 kb/s
G.723.1		5.3,6.4 kb/s
GSM		12,89 kb/s (13200 bps)

Görüntü Kodekleri (Video Codecs)

Görüntü kodekleri ile görüntü bilgisinin ne şekilde sıkıştırılıp açılacağı belirtilir.

Call Signaling H.225 Protokolü

Bu protokol ile Gatekeeper adı verilen birimin mesajları [RAS (Registration, Admission, Status)] tanımlanır. RAS mesajları ile şunlar yönetilir “registration”, “admission” ve “status”. Ayrıca Q.931 standardına dayalı olarak Call-set-up (çağrı başlatma) ve Call-termination (çağrı sonlandırma) mesajları ile RTP protokolünün kullanımı tanımlanır.

Control Signaling H.245 Protokolü

Kullanıcılar arası konferans kontrolünü ve “capability exchange” mesajlarını tanımlar. Capability exchange mesajları ile birimlerin (örneğin terminaller) desteklediği özelliklerin ve protokollerin bilgisinin karşılıklı olarak paylaşımı sağlanmaktadır.

Güvenlik (Security) H.235 Protokolü

Haberleşme yapacak H.323 kanallarının güvenliğini ve otantikasyonunu tarif eder.

Haberleşme Sırasında Kullanılan Diğer Protokoller

Media stream iletilirken RTP ve RTCP protokolleri kullanılır. RTP media stream taşır, RTCP ise taşınan bilgiye ilişkin durum ve kontrol bilgisini iletir.

En alt seviyede IP ve bunun üzerinde Transport Control Protocol (TCP) ve User Datagram Protocol (UDP) protokolleri kullanılır. TCP bağlantı tabanlı bir protokoldür ve haberleşme esnasında sürekli açık olup, iletilen bilginin yerine ulaştığını garanti etmek üzere tasarlanmıştır. Bu sebeple kontrol bilgisi iletiminde TCP kullanılmaktadır. UDP ise bağlantı tabanlı bir protokol değildir, dolayısıyla gönderilen bilginin iletimi garanti edilemez. TCP'nin garantili bir protokol olması sebebiyle UDP'ye kıyasla yük yaratan overhead sorunu vardır, bu yüzden iletilen bilginin en büyük kısmını oluşturan ses ve görüntü iletimi için UDP kullanılmaktadır. Şekil 3.2'de bir terminalde kullanılan protokollerin şeması verilmiştir.

Data	Control and Signaling		Audio/ Video	Registration
T.120	H.225.0 Call Signaling	H.245 Conference Control	RTP/RTCP	H.225.0 RAS
TCP			UDP	
Network Layer				
Data link Layer				
Physical Layer				

Şekil 3.2. Bir H.323 terminalinde bulunan protokollerin şeması

3.4 H.323 Terminal Özellikleri

H.323 terminali şu özellikleri desteklemelidir [Trillium] :

H.245: "Capability exchange" işlemini yapabilmek ve medya kanalı açabilmek için.

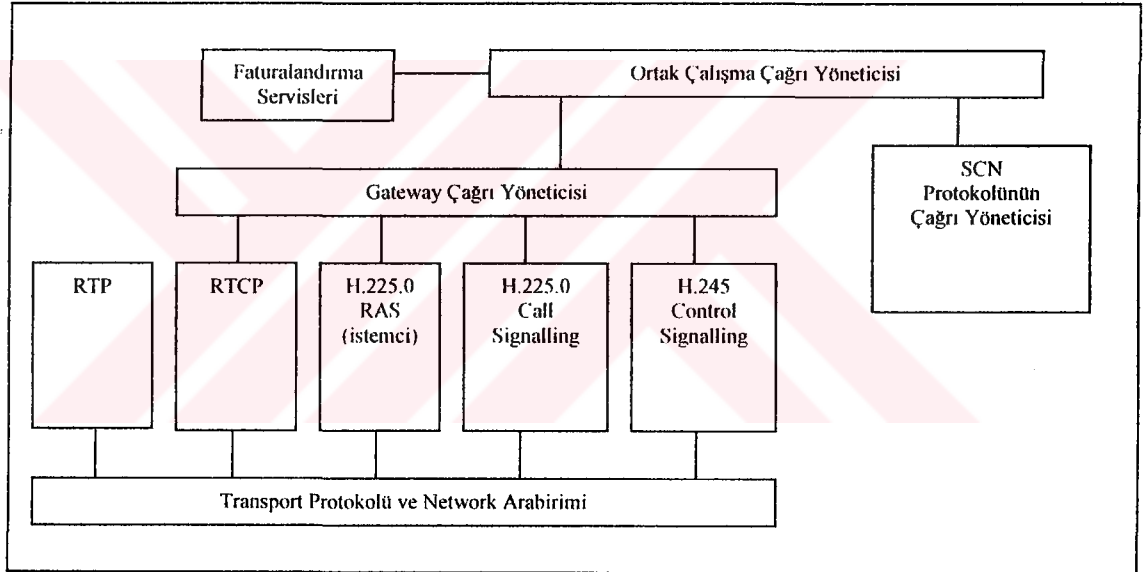
H.225: “Call signaling” ve “Call setup” işlemleri için.

RAS: Gatekeeper elemanına kayıt olmak ve diğer “admission” işlemleri için.

RTP/RTCP: ses ve video paketlerini doğru sırada gönderip alabilmek için .

3.5 H.323 Gateway Özellikleri

H.323 gateway elemanı bir H.323 ağı ile farklı bir ağı birbirine bağlayarak “call setup”, “call release” işlemleri için protocol dönüşümlerini yapar. Ayrıca medya formatı dönüşümlerini de yapar. Şekil 3.3’te IP ağını bir Switched Circuit Network (SCN) ağına (örneğin ISDN) bağlayan bir H.323 gateway örneği ve protokol şeması bulunmaktadır [Trillium] .



Şekil 3.3. H.323 gateway protokol şeması örneği

Gateway, H.323 tarafında “Control signaling” ve “Capability exchange” işlemleri için H.245 protokolünü, “Call setup” ve “Call release” işlemleri (Call signaling) için H.225 protokolünü, gatekeeper ile kayıt ve “admission” işlemleri için H.225 protokolünü ve RAS’ı kullanır. SCN ağı tarafında ise SCN’e özel protokolleri çalıştırır.

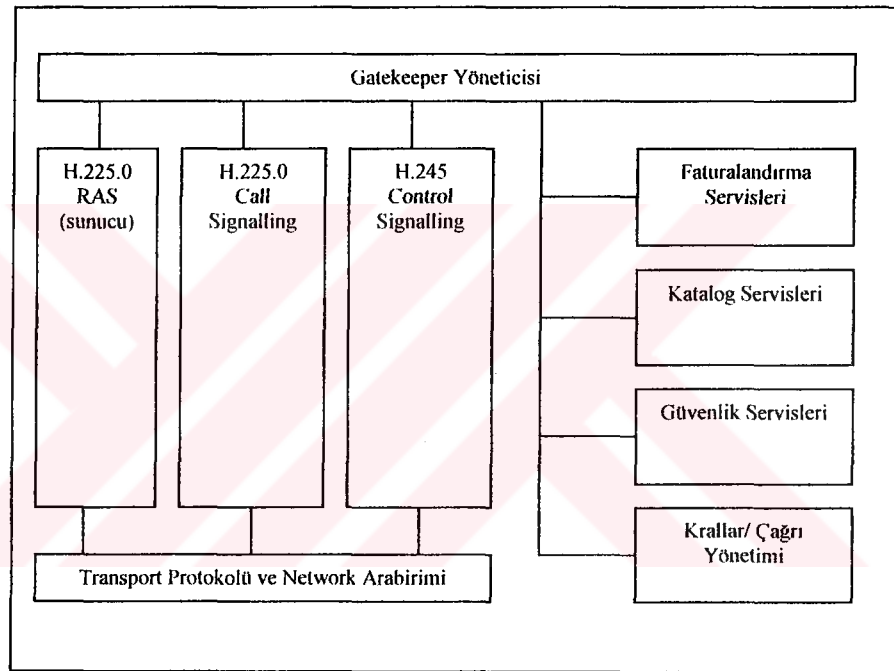
Terminaller, gateway ile H.245 ve H.225 protokolleri ile haberleşir. Gateway bu protokoller ile iletilen bilgileri H.323 olmayan ağa özgü protokollerdeki karşılığına çevirir. Çağrı başlatma (Call setup) ve çağrı sonlandırma (Call release)

işlemlerini her iki taraf için de uygular. Ayrıca ihtiyaç duyuluyorsa ses, video ve data format dönüşümlerini de yerine getirir.

Gatewayler, terminaller gibi bir end-point (uç birim) olarak gatekeeper'a kayıt olurlar. Fakat gatekeeper hangi end-point'in gateway olduğunu bilir.

3.6 H.323 Gatekeeper Özellikleri

Gatekeeper'ın yerine getirebileceği zorunlu ve seçimsiz görevler vardır. Şekil 3.4'te bir gatekeeper'ın bileşenlerini gösteren bir şema ve görevleri verilmiştir [Trillium].



Şekil 3.4. H.323 gatekeeper protokol şeması

Zorunlu Gatekeeper Görevleri

Adres Dönüşümü: H.323 terminalleri “alias” ile adreslenebilirler. H.323 ağı içinden gelen çağrılar için bu alias’ların IP adreslerine dönüştürülmesi işlemidir. Benzer şekilde H.323 olmayan ağlardan gateway üzerinden gelen çağrılar kendine has adresleme yöntemi ile H.323 ağına çağrı talebinde bulunur (Örneğin E.164 için 310-444-2121 olabilir), bu durumda da gatekeeper bu adresleri IP adreslerine dönüştürür [Trillium].

Katılım Kontrolü (Admission Control): Eğer bir H.323 alanında bir gatekeeper varsa H.323 ağına katılan end-point'ler kendilerini bu alana kayıt ettirebilirler. Bunun için Admission Request (ARQ), Admission Confirm (ACF) ve Admission Reject (ARJ) mesajları kullanılır. Bu mesajlara RAS mesajları adı verilir [Trillium].

Bant Genişliği Kontrolü (Bandwidth Control): Gatekeeper bant genişliği kullanımını kontrol etmek için Bandwidth Request (BRQ), Bandwidth Confirm (BCF) ve Bandwidth Reject (BRJ) mesajlarından oluşan RAS mesajlarını kullanır. Bu sayede izin verilen bant genişliğini aşacak çağrı talepleri olumsuz olarak cevaplanabilecek ve izin verilen sınırlar içinde kalacaktır [Trillium].

Zorunlu Olmayan Gatekeeper Görevleri

Çağrı Kontrol Sinyalleşmesi (Call Control Signaling): H.323 uç noktaları arasındaki H.225 çağrı sinyalleşmesi mesajları terminaller arasında doğrudan iletelebileceği gibi gatekeeper üzerinden de iletebilir. Gatekeeper'ın bu görevine çağrı sinyalleşmesi yönlendirme (call signaling routing) adı verilir [Trillium].

Çağrı Otorizasyonu (Call Authorization): Uç noktalar arası H.225 çağrı sinyalleşmesi mesajlarının yetkilendirilip yetkilendirilmediği gatekeeper tarafından kontrol edilebilir. Yetki mekanizması zaman bazlı kısıtlamalarla veya erişim bazlı olabilir. Yetkisiz uç noktalara red mesajı gönderilir. Erişim bazlı kısıtlamada arayan tarafın diğer tarafı aramaya yetkisi olup olmadığına bakılır [Trillium].

Çağrı Yönetimi (Call Management): H.323 gatekeeper elde ettiği bilgileri kullanarak çağrıları yönetebilir. Örneğin yük dağıtımını yapabilmek için çağrıları başka gatekeeper'lara yönlendirebilir [Trillium].

3.7 H.323 Çağrısı Oluşturulması (Call setup)

Bir H.323 çağrısı oluşturulurken şu adımlar gerçekleşir [Trillium]:

- Gatekeeper bulunması
- Gatekeeper'a kayıt olunması (uç nokta olarak)
- Uç noktanın çağrı oluşturma safhasına girmesi



- Uç nokta ve gatekeeper arasında “Capability exchange” mesajlaşmasının yapılması

- Çağrının başlaması

- Çağrının sonlanması





4. SESSION INITIATION PROTOCOL (SIP)

SIP protokolü IETF tarafından İnternet telefonu, İnternet multimedia konferans uygulaması ve multimedia dağıtım uygulamalarındaki “session”ların oluşturulması, değiştirilmesi ve sonlandırılmasını sağlamak için geliştirilmiştir. SIP bir “application layer” protokolüdür. SIP mesajları kullanıcı (client) tarafından üretilip sunucuya (server) iletilir ve sunucu mesajları işler. SIP ile kullanıcı isimlerinin adreslemesi ve yönlendirmesi yapılabilir. Bu sayede kullanıcıların kendilerine gelen çağrıları cevaplayabilmeleri mümkün olmaktadır. SIP protokolü bir çağrı için şu servisleri verebilir [RFC2543]:

- Kullanıcı yeri: Hangi uç sistemin çağrı için kullanılacağını belirlenmesi .
- Kullanıcının yetenekleri: Kullanılacak medyanın ve medya parametrelerinin belirlenmesi.
- Kullanıcı mevcudiyeti (User availability): Aranan tarafın çağrıya katılmayı isteyip istemediğinin belirlenmesi.
- Çağrının gerçekleşmesi: (Call setup): Çaldırma (ringing) ve çağrı parametrelerinin gerçekleşmesi.
- Çağrı yönetimi (Call handling): Çağrının yönlendirilmesi ve sonlandırılması.

SIP konferans kontrol servisi sunmaz. SIP metin tabanlı bir protokoldür.

4.1. SIP Bileşenleri

SIP bileşenlerini iki ana başlık altında inceleyebiliriz.

4.1.1. Kullanıcılar (User Agents)

Kullanıcı sistemde yer alan uç sistemdir. İki şekli vardır; User Agent Client (UAC) ve User Agent Server (UAS). UAC SIP isteğini (request) oluşturur ve UAS’a iletir. UAS isteği alır, işler ve cevaplar (response).



4.1.2. Ağ Sunucuları (Network Servers)

Üç tip ağ sunucusu vardır:

- Kayıt sunucusu (location server): Kullanıcıların yerlerine dair adres bilgilerini tutar ve günceller.
- Proxy server: İstekleri alır ve bir sonraki sunucuya (Next-hop server) iletir.
- Yönlendirici sunucu (redirect server): Kullanıcı isteklerini bir sonraki next-hop server'a iletmek yerine bu next-hop server'ın adresini verir.

4.2. SIP Mesajları

SIP uç sistemleri arasında kullanılan mesajlar aşağıdadır:

- INVITE: Bir kullanıcıyı çağrıya davet etmek içindir.
- BYE: Çağrıyı sonlandırmak içindir.
- ACK: INVITE mesajından sonra teyid içindir.
- OPTIONS: Çağrıda kullanılabilecek özelliklerin öğrenilebilmesi içindir.
- REGISTER: SIP kayıt sunucusuna kullanıcının yeri hakkında bilgi vermek içindir.
- CANCEL: Henüz başlatılmamış bir çağrının talebini iptal etmek içindir.

4.3. SIP Operasyonu

SIP çağrılarında kullanıcıların adresleri SIP mesajlarında yer alan Call ID ile ifade edilir. Bir SIP çağrısının gerçekleştirilmesi için arayan taraf aranan tarafa mesajlarını doğrudan ya da yönlendirici sunucular üzerinden gönderebilir [Handley,1999].



4.3.1. SIP Adreslemesi

SIP adresleri bir kullanıcıya ya da bir kullanıcı grubuna ait olabilir. sip:username@host formatındaki adrese SIP Uniform Resource Locator (URL) denir.

4.3.2. SIP Sunucusunun Bulunması

SIP kullanıcısı mesajlarını SIP sunucusuna proxy server üzerinden ya da Uniform Request Identifier (URI) ile tanımlanan IP adresi ve porta gönderebilir.

4.3.3. SIP Transaction

Bir kullanıcı UAC, bir SIP sunucusuna mesaj gönderip cevabını aldığı anda SIP transaction gerçekleşmiş olur. SIP mesajları TCP ya da UDP protokolleri ile iletilebilir.

4.3.4. SIP Daveti (SIP Invitation)

Bir SIP çağrısı gerçekleşirken önce arayan taraf INVITE mesajı gönderir, ardından aranan taraf ACK mesajı ile onay gönderir. Çağrıya ya da konferansa ait bilgiler “session descriptor” ile iletir. Eğer aranan taraf çağrıyı kabul ederse kendisine iletilen session descriptor’a benzer bir descriptor’u arayan tarafa iletir.

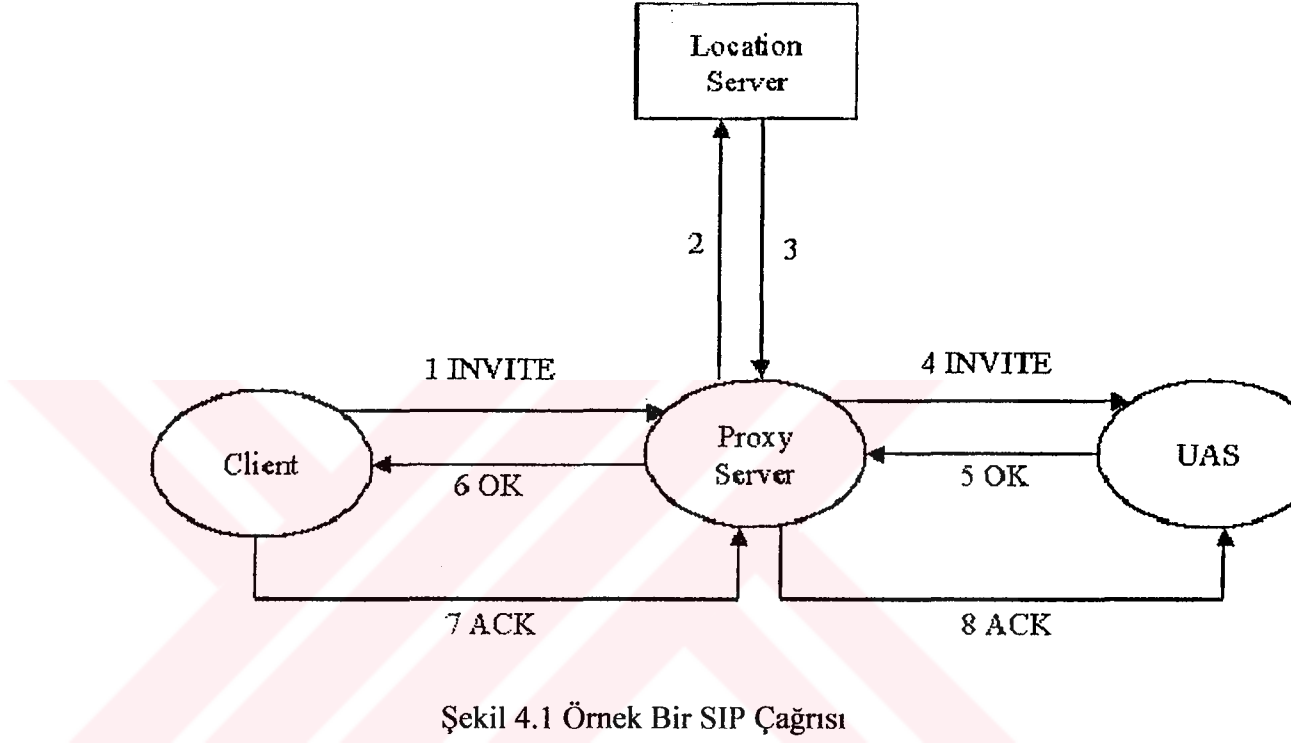
4.3.5. Mevcut SIP Çağrısının Değiştirilmesi

Mevcut çağrı parametrelerinde bir değişiklik gerektiğinde yeni parametreler ve aynı Call ID ile yeni bir INVITE mesajı gönderilir, böylece yeni parametreler ile yeni bir çağrı başlatılabilir.

4.3.6. Örnek Bir SIP Çağrısı

Şekil 4.2’de bir SIP çağrısının gerçekleşmesi örneklenmiştir. 1. adımda kullanıcı (UAC) proxy server’a INVITE mesajını gönderir. 2. adımda proxy server aranan tarafın yani SIP server’ın adresini öğrenmek için kayıt sunucusuna

(location server) başvurur. 3. adımda proxy server kayıt sunucusundan aranan tarafın adresini öğrenir. 4. adımda proxy server aranan tarafa INVITE mesajını iletir. 5. adımda UAS yani SIP server tamam cevabını proxy server'a iletir. 6. adımda proxy server tamam cevabını arayan tarafa yani UAC'a tamam cevabını iletir. 7 ve 8. adımlarda UAC'ın ACK mesajı UAS'a iletilir.



4.4. SIP ve H.323'ün Karşılaştırılması

H.323 ilk olarak LAN ve Integrated Services Digital Network (ISDN) sistemleri için geliştirildiği için İnternet üzerinden çalışacak VoIP uygulamaları için bazı sıkıntılar ortaya çıkmaktadır. SIP bu sıkıntıların aşılması için geliştirilmiştir.

H.323, SIP'e göre daha karmaşık bir protokoldür. H.323 mesajlar için ikilik gösterimi kullanır. SIP ise HTTP altyapısını kullandığı için basit bir yapısı vardır.

H.323'te yüzlerce mesaj varken SIP'te yalnızca 37 mesaj vardır.

H.323 ilk olarak tek LAN için tasarlandığı için ölçekleme sorunu vardır. SIP daha ölçeklenebilir bir yapıya imkan vermektedir.

SIP IETF gibi güçlü bir organizasyon tarafından geliştirilmiştir, ancak H.323 de çok yaygın bir kullanım tabanına ve pazar payına sahiptir.

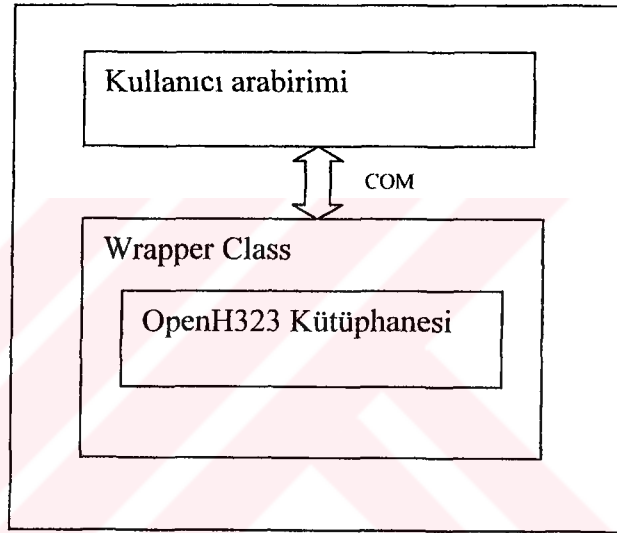


5. GELİŞTİRİLEN İSTEMCİ YAZILIMI

H.323 protokolü ile uyumlu sesli görüşmeyi sağlayan bir H.323 uç nokta yazılımı geliştirilmesi hedeflendi.

5.1. Yazılımın Bileşenleri

Yazılım Windows patformunda çalışabilecek şekilde tasarlandı. Yazılımın mimarisi Şekil 5.1’de belirtildiği gibidir.



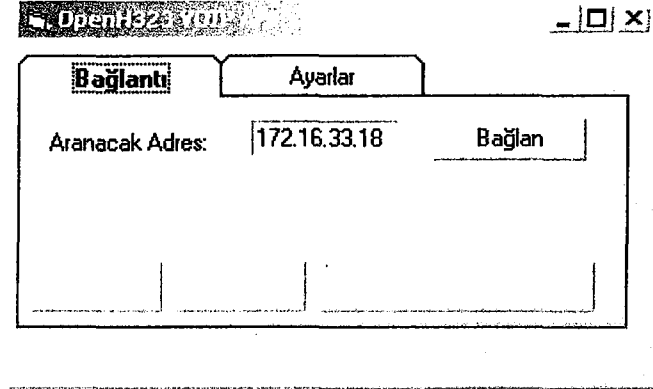
Şekil 5.1 Yazılımı oluşturan parçalar

5.1.1. Kullanıcı Arabirimi

Yazılım Windows patformunda çalışmak üzere MS Visual Basic 6.0 yazılım dili ile geliştirilmiştir. Yazılımın kullanılabilmesi için bilgisayarda ses kartı, mikروفon ve kulaklık donanımları bulunmalıdır. Kullanıcı arabirimi wrapper class olarak belirtilen ve bir alt seviyede çalışan yazılım bileşenine erişerek H.323 çağrısı yapabilir ve karşı taraftan gelen bir çağrıyı cevaplandırabilir.

Şekil 5.2’de belirtilen ekranda arama yapılmak istenen kullanıcının, yani karşı tarafın bilgisayarının IP numarası “Aranacak Adres” alanına girilir. Daha sonra “Bağlan” düğmesine basılır. Bu aşamada program “wrapper class” ile haberleşerek çağrı başlatmak üzere karşı taraf ile bağlantı kurulmasını sağlar. Eğer karşı taraf

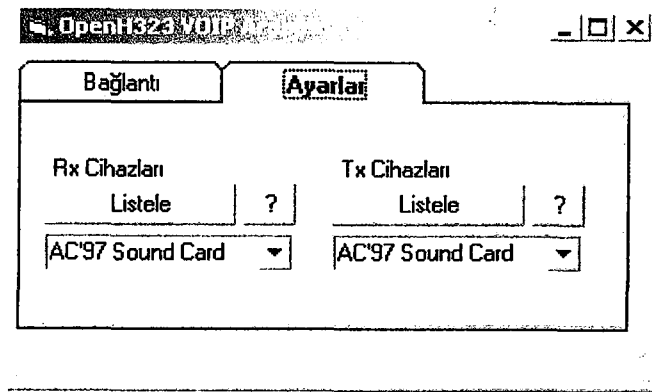
çağrıyı kabul ederse H.323 görüşmesi başlar. Arama yapan kullanıcının “Görüşmeyi Sonlandır” düğmesine basması ya da karşı tarafın görüşmeyi sonlandırmaya karar vermesi ile görüşme son bulur.



Şekil 5.2 Kullanıcı arabirimi bağlantı ekranı

H.323 çağrısı yapılabilmesi için TCP 1720 ve UDP 1719 portları kullanılmaktadır. Dolayısıyla programın çalışacağı bilgisayarda ve ağda bu portların kullanımına izin verilmiş olmalıdır. Ayrıca çağrı yapılacak karşı tarafın da bu portlara erişiminin açık olması gereklidir.

Şekil 5.3'te gösterilen ayar ekranından ses kartı ile ilgili ayarlar yapılmaktadır. Bilgisayar üzerinde birden fazla ses kartı ya da ses cihazı varsa bunlardan hangisinin mikrofon hangisinin ses çıkışı için kullanılacağı bu ekrandan ayarlanmalıdır, gerekli ayarlar yapılmazsa karşı tarafa ses iletilemeyebilir ya da karşı taraftan gelen ses duyulamayabilir.



Şekil 5.3 Kullanıcı arabirimi ayarlar ekranı

Wrapper class ile kullanıcı arabirimi programı arasındaki etkileşim fonksiyon ve event mekanizmaları ile sağlanır. Kullanıcı arabirimi programı, kişinin ekranları üzerinden verdiği komutları bu mekanizmalar yardımıyla gerçekleştirir.

İstemci yazılımından örnek Visual Basic program kodu aşağıda verilmiştir.

```
Private Sub OH3231_ReceivedCall(ByVal RemoteUserName As String)
    SBar.SimpleText = RemoteUserName + " Kullanıcısından çağrı var !!"
    lblArayan.Caption = RemoteUserName + "Arıyor!"
    cmdCevapla.Enabled = True
    cmdReddet.Enabled = True
    cmdConnectToDestination.Enabled = False
End Sub

Private Sub cmdCevapla_Click()
    OH3231.AnswerCall
End Sub

Private Sub cmdConnectToDestination_Click()
    SBar.SimpleText = "Bağlanıyor"
    SBar.Refresh
    OH3231.DestinationAddress = tbDestinationAddress.Text
    OH3231.Connect
End Sub

Private Sub cmdReddet_Click()
    OH3231.DontAnswer
End Sub

Private Sub cmdSonlandir_Click()
    OH3231.Hangup
End Sub

Private Sub OH3231_Connected(ByVal RemoteUserName As String)
    SBar.SimpleText = "Bağlandı -->" + RemoteUserName
    Call reset_form_to_conversation
End Sub
```

```

Private Sub OH3231_Disconnected(ByVal RemoteUserName As String)
    SBar.SimpleText = "Bağlantı kesildi -->" + RemoteUserName
    Call reset_form
End Sub

Private Sub OH3231_Error(ByVal Err As Integer, ByVal Error As String)
    MsgBox (Error + "Err num:" + CStr(Err))
End Sub

Private Sub OH3231_ReceivedCall(ByVal RemoteUserName As String)
    SBar.SimpleText = RemoteUserName + " Kullanıcısından çağrı var !!"
    lblArayan.Caption = RemoteUserName + "Arıyor!"
    cmdCevapla.Enabled = True
    cmdReddet.Enabled = True
    cmdConnectToDestination.Enabled = False
End Sub

Private Sub OH3231_RemoteRinging()
    SBar.SimpleText = "Çaldırılıyor"
    Call reset_form_to_conversation
End Sub

```

5.1.2. Wrapper Class

Wrapper class, en alt seviyede bulunan openH323 kütüphanesine erişimi kolaylaştırmak için geliştirilmiştir. OpenH323 kütüphanesi C++ yazılım geliştirme dili ile hazırlanmış olduğu için C++ dili dışındaki dillerden erişim mümkün olmamaktadır. Windows ortamında, yazılım bileşenlerinin birbiri ile etkileşimini ve bir yazılım dili ile geliştirilen bir bileşenin başka bileşenler tarafından kullanılabilmesi için Component Object Model (COM) teknolojisi kullanılabilmektedir.

Component Object Model (COM)

COM Microsoft tarafından geliştirilmiş bir yazılım geliştirme teknolojisidir. COM kullanmanın sağladığı avantajlar şunlardır:

- Yazılımlar birbirinden bağımsız bileşenler şeklinde geliştirilebilmektedir.

- Bileşenler daha kolay genişletilebilir ve yönetilebilir.
- Karmaşık algoritmalar mantıksal birimler olarak bileşenler şeklinde geliştirilebilir.
- Bileşenler farklı uygulamalarda tekrar tekrar kullanılabilir.
- Hazır geliştirilmiş bileşenler dışarıdan temin edilerek uygulama geliştirme süresi ve maliyeti düşürülebilir.
- Bileşenler farklı bilgisayarlara yüklenerek dağıtık biçimde çalışma imkanı yaratılabilir.
- Arayüz tabanlı programlama imkanı sağlar.

COM bileşenleri VC++ ya da Visual Basic ile hazırlanabilir. Visual Basic ile kod geliştirme çok daha hızlı ve kolaydır fakat çalışma esnasında performans düşüşüne ve alt seviyede bazı kontrollerin yitirilmesine sebep olur. C++ ile geliştirme yaparken COM'un gerektirdiği disiplini uygulamak için alternatifler mevcuttur.

- En zor yöntem yapıların kendimiz tarafından oluşturulması
- Microsoft Foundation Classes (MFC) kütüphanesinin kullanılması. MFC, Windows için program geliştirmek isteyen programcılara yardımcı olması için Microsoft tarafından geliştirilmiş bir yazılım kütüphanesidir.
- Active Template Library (ATL) kullanılması. ATL, COM bileşenleri oluşturmak için kullanılan bir "template library" (şablon kütüphanesi) dir. ATL sayesinde COM bileşenleri ve özel bir COM bileşeni olan "Activex Control" daha hızlı bir şekilde geliştirilebilir. Geliştirme esnasında VC++ dili kullanılmaktadır.

ATLin MFC'ye göre avantajları:

MFC kütüphanesi sınıf hiyerarşisi ile oluşturulmuştur. ATL ise birbirinden ayrı çeşitli şablonlar (template) ile meydana getirilmiştir. MFC kullanımında, kullanılmayan özellikleri içeren fonksiyonlar da sınıf hiyerarşisi gereği yazılımın içinde gömülmek zorunda kalmaktadır. ATL daha az hafıza ihtiyacı olan daha küçük boyutta yazılım geliştirme imkanı sağlamaktadır.

MFC kütüphanesi bileşenleri yazılıma Dynamic Link Library (DLL) olarak bağlantırlmakta, dolayısıyla yazılımın çalıştırılacağı bilgisayarda Run-time DLL kütüphanesi dağıtımını zorunlu kılmaktadır. ATL ise yazılım içinde kaynak kodu olarak derlendiği için ayrıca bir kütüphane dağıtımını gerekli kılmaz.

Activex Control Kullanımının Sağladığı İmkanlar:

Activex Control özel bir COM bileşeni olmakla birlikte şu imkanları sağlar:

Geliştirilen bileşen farklı yazılım geliştirme dilleri içinde kolayca kullanılabilir (Örnek; Visual Basic, Delphi, VC++).

Internet Explorer yardımıyla web sayfaları içinde kullanılabilir. Bu özellik İnternet üzerinden kullanılacak uygulamaların kolay dağıtım ve yüklenmesine imkan sağlar. Örneğin belirli bir web sayfası açıldığında kullanıcıdan izin istemek suretiyle gerekli yazılım kolayca kullanıcı bilgisayarına yüklenebilmektedir.

Active Template Library teknolojesi ve MS Visual C++ 6.0 yazılım dili kullanılarak "Activex DLL" olarak oluşturulan wrapper class diğer yazılım dilleri içinde de kullanıma izin vermektedir.

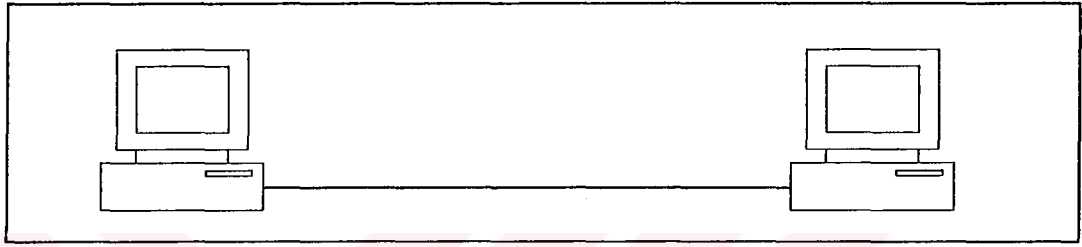
Wrapper class openH323 kütüphanesinde bulunan fonksiyon ve event mekanizmalarını ATL şablonlarına uygun olarak içerecek ve bir üst yazılımın (kullanıcı arabirimi) erişebileceği şekilde hazırlanmıştır. OpenH323 kütüphanesinin 1.13 versiyonu kullanılmıştır.

5.2. Yapılan Testler

Sesli Görüşme Testi

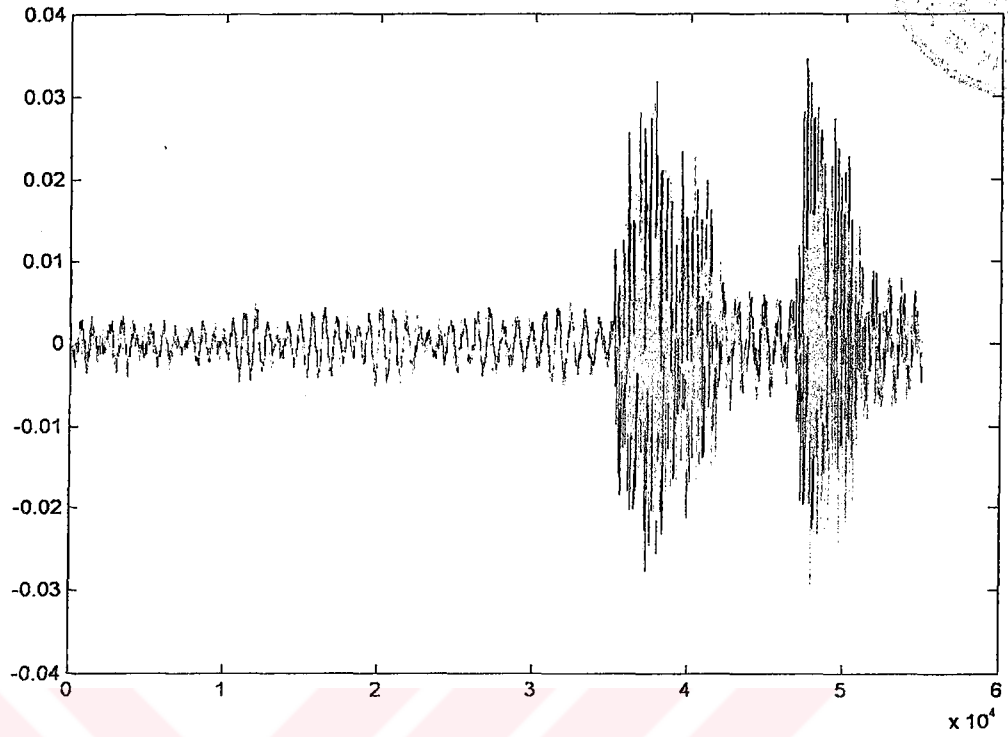
Wrapper class ve bunu kullanan kullanıcı arabirimi programı tamamlandıktan sonra test edildi. Test için iki bilgisayar kullanıldı bu bilgisayarlar aynı bilgisayar ağına TCP/IP ile bağlı bilgisayarlardı. Kıyaslama yapabilmek için “Speak Freely” isimli yazılımın 7.2 versiyonu kullanıldı [Link4].

GSM kodeki kullanarak sesli görüşme gerçekleştirildi.



Şekil 5.4. Test Bilgisayarları

İki yazılımın aynı ses iletimi için test edilmesi için önceden 10 adet ses dosyası kaydedildi. Bu ses dosyalarının her biri tek tek iletilerek alıcı tarafta kaydedildi. Daha sonra gönderilen ve alınan ses dosyaları Matlab programına yüklenerek karesel hata hesaplamaları yapıldı. Şekil 5.5’te istemci yazılımı ile iletilen ve alınan ses dosyasının üst üste bindirilmesi ile elde edilen grafiğin bir parçası verilmiştir. Bu iletim oldukça başarılı olmuştur çünkü gönderilen ses dosyasına ait grafiğe çok yakın bir grafiğe sahip bir ses dosyası alınmıştır.



Şekil 5.5. İletilen ve Alınan Ses Dosyalarının Karşılaştırma Grafiği

Tablo 5.1’de karesel hata hesaplama sonuçları tablosu verilmiştir. Her iki yazılım için de yakın sonuçlar elde edilmiştir. İstemci yazılım için $1.22E-04$ ortalama sonucu elde edilirken Speak Freely yazılımı için $1.28E-04$ ortalama sonucu elde edilmiştir.

Tablo 5.1. Ses İletimine Ait Karesel Hata Hesaplam Sonuçları Tablosu

Ses Dosyaları	OpenH323 İstemci Yazılımı	Speak Freely Yazılımı
1. Ses Dosyası	8.81E-05	9.19E-05
2. Ses Dosyası	1.30E-04	1.43E-04
3. Ses Dosyası	8.56E-05	8.46E-05
4. Ses Dosyası	7.06E-05	6.75E-05
5. Ses Dosyası	8.32E-05	8.21E-05
6. Ses Dosyası	1.47E-04	1.46E-04
7. Ses Dosyası	1.67E-04	1.79E-04
8. Ses Dosyası	1.43E-04	1.49E-04
9. Ses Dosyası	1.78E-04	1.95E-04
10. Ses Dosyası	1.25E-04	1.38E-04
Ortalama	1.22E-04	1.28E-04



Bant Genişliği Kullanım Testi

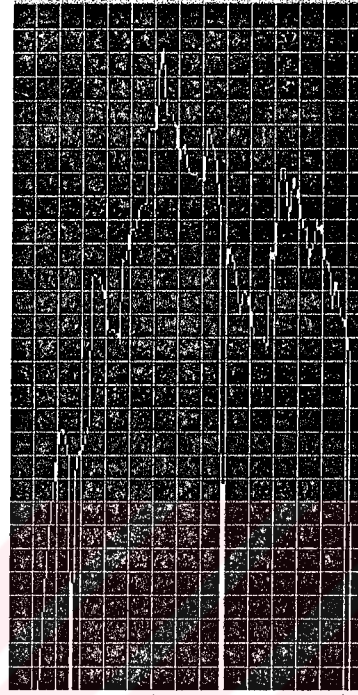
openH323 ile geliştirilen istemci yazılımı benzer bir yazılım ile kıyaslanarak test edildi. Kıyas için İnternet üzerinden ücretsiz olarak dağıtılan ve popüler olan “Speak Freely” isimli yazılımın 7.2 versiyonu kullanıldı [Link4]. Her iki yazılım da GSM kodeği kullanılarak test edildi.

GSM kodek kullanan yazılımlar aynı LAN içindeki iki ayrı PC’ye kuruldu. Ayrıca bant genişliği kullanımını tespit için NetLimiter [Link3] yazılımının deneme sürümü kullanıldı. İstemci yazılımları ile sesli görüşme yapılırken bir yandan da bant genişliği kullanımı grafikleri kaydedildi. Test için 150 saniyelik bir müzik kaydı içeren ses dosyası kullanıldı.

GSM kodeği için 13200 bit/s bant genişliği belirtilmektedir. Ancak paket tabanlı ağlarda UDP protokolü ile transfer edilirken ekstra bitler gönderilmesi gerektiği için bu bant genişliğinden daha yüksek bant genişliği ihtiyacı doğmaktadır. Bir UDP paketi içinde 4 GSM kodek frame’i gönderilmektedir. Her bir GSM bilgi paketi i 33 byte büyüklüdedir. RTP,UDP ve IP protokollerinden kaynaklanan ekstra iletim maliyeti 40 byte büyüklüğündedir. Her 132 byte GSM bilgisi için 172 byte toplam iletim gerekmektedir dolayısıyla bant genişliği ihtiyacı 17200 bit/s (2,1 kByte/s) olarak hesaplanabilir.



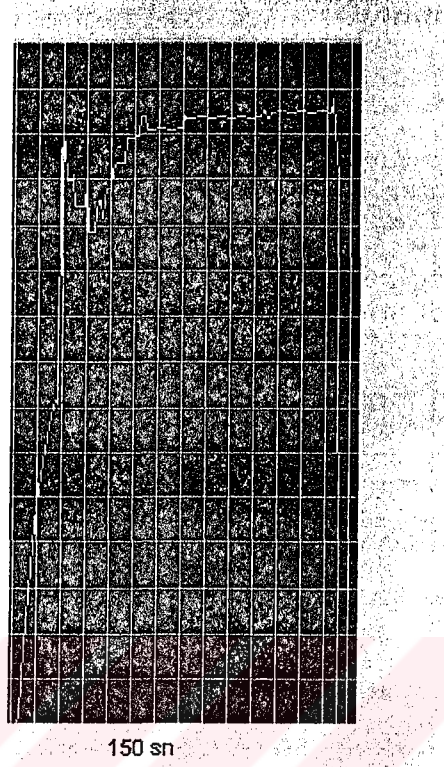
İstemci yazılımı için elde edilen grafik şekil 5.6 ile verilmiştir.



Şekil 5.6. İstemci Yazılım İçin GSM Kodek Bant Genişliği Grafiği

GSM kodek için bant genişliği kullanım oranı 17200 bit/s yani 2,1 kB/s olarak belirlenmişti. İstemci yazılım için yapılan testlerde elde edilen sonuçlar bu bilgi ile paralellik taşımıştır. Konuşma esnasında doğal olarak oluşan sessizlikler esnasında bant genişliği kullanımının azaldığı gözlenmiştir.

Speak Freely yazılımı için elde edilen grafik şekil 5.7 ile verilmiştir.



Şekil 5.7. Speak Freely Yazılım İçin GSM Kodek Bant Genişliği Grafiği

OpenH323 kütüphanesi ile geliştirilen istemci yazılımının nispeten daha az bant genişliği kullandığı görülmüştür.



6. SONUÇLAR

Bu çalışmada IP tabanlı ağlarda VoIP ile ses iletimi problemlerin gidermek, ağ üzerinden tetmin edici bir görüşme yapma imkanı sağlamak için geliştirilmiş H.323 ve SIP protokolleri incelendi. SIP protokolünün basit ve hızlı yapısı ile İnternet üzerinden kullanılan uygulamalarda giderek artan bir popülerliği olduğu belirlenmiştir. Diğer yandan H.323'ün de uzun süreden beri kullanılan ve özellikle yerel ağlar üzerinde çalıştırılan IP telefonu uygulamalarında daha fazla kullanıldığı ve üretici firmalar tarafından desteklendiği görülmüştür.

Bu çalışmada açık kaynak kodlu bir kütüphane olan OpenH323'ün kullanıldığı bir istemci yazılımı geliştirilmiştir. Bu yazılım geliştirilirken MS Visual Basic 6.0 yazılım dili, MS Visual C++ 6.0 dili ve MS Active Template Library kullanılmıştır.

Geliştirilen yazılım fonksiyon bakımından test edilerek iki bilgisayar arasında sesli görüşme yapma görevini başarıyla yerine getirdiği görülmüştür. Bu amaçla Speak Freely yazılımı ile karşılaştırılmıştır. İletilen sesin orijinal ses ile kıyaslanması ile elde edilen ortalama karesel hata testinde de tatmin edici sonuçlar elde edilmiştir. Ayrıca ağ bant genişliği kullanımı bakımından öngörülen sınırlar içinde kaldığı tespit edilmiştir. Her iki testte de Speak Freely yazılımının test değerlerine yakın değerler elde edilmiştir.

İleride şu çalışmaların yapılabilceği düşünülmektedir: Elde edilen yazılımın ses iletimi esnasında çeşitli ağ koşulları altında ne kadar gecikmeye sebep olduğu incelenebilir. Elde edilen yazılımın ticari olarak satılan ve kullanılan IP telefonu ekipmanları ile birlikte çalışması başarımı açısından test edilebilir. VoIP sistemlerinin klasik PSTN ağları ile birlikte çalışması konusu incelenebilir. VoIP servislerinin ticari hizmet olarak sunumu esnasında ihtiyaç duyulan esnek ücretlendirmenin nasıl yapılabilceği konusu araştırılabilir. Kullanıcıların güvenli ses iletimi ihtiyacının karşılanmasına çalışılırken bu konuyla ilgili devletlerin istihbarat konusundaki ihtiyaçlarının ve yaptırımlarının nasıl uygulanabileceği araştırılabilir.



KAYNAKLAR

- [Schulzrinne,1996] Schulzrinne H., et al, IETF RFC 1889, "RTP: A Transport Protocol for Real-Time Applications", Ocak 1996.
- [Braden,1997] Braden R., et al, IETF RFC 2205,"Resource ReSerVation Protocol (RSVP)", Eylül 1997.
- [ITU-T,1998] ITU-T, "Draft H.323v4", Şubat 2001
- [Handley,1999] Handley M., et al, IETF RFC 2543, "SIP: Session Initiation Protocol", Mart 1999
- [Mortada,2001] Mortada I., and Probst W., "Internet telephony signaling", Telematics and Informatics, 2001
- [Link1] Open H.323 kütüphanesi, <http://www.openh323.org>
- [Link2] ATM Forum sitesi , <http://www.atmforum.com>
- [Trillium] Trillium H.323 Tutorial
- [Seydim,1999] Seydim A., "Voice Over IP", 1999
- [Link3] NetLimiter sitesi , <http://www.netlimiter.com>
- [Link4] Speak Freely yazılımının sitesi , <http://www.speakfreely.org>
- [Link5] Skype yazılımının sitesi, <http://www.skype.com>
- [Link6] ATM hakkında bilgi, http://www.iec.org/online/tutorials/ip_int/topic04.html



ÖZGEÇMİŞ

Mesut Tomruk, 9 Eylül 1975 tarihinde Edirne’de doğdu. İlköğretimini Lüleburgaz Atatürk İlkokulu’nda tamaladıktan sonra sırasıyla Lüleburgaz Ortaokulu ve Lüleburgaz Lisesi’ni tamamladı. Lisans öğrenimini 2002 yılında İstanbul Teknik Üniversitesi Elektrik Elektronik Müh. Fak. Elektrik Müh. Bölümünde tamamladı. Lisans eğitimini tamamlarken bitirme çalışmasını “MS .Net Platformu İle Kitap Satışı İçin Elektronik Ticaret Sitesi Geliştirilmesi” konusu üzerinde hazırladı. 2002 yılından bu yana Gebze Yüksek Teknoloji Enstitüsü Elektronik Mühendisliği Bölümü’nde Yüksek Lisans eğitime devam etmektedir. Ayrıca 1997 yılından bu yana özel sektörde yazılım geliştirme uzmanı olarak çalışmaktadır.

