

**YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**TAŞINABİLİR SİSTEMLERDE SIP SİNYALLEŞMESİ
ARACILIĞI İLE VOIP İSTEMCİ UYGULAMASI**

Elektronik ve Haberleşme Müh. Nazan GÖNEN

**FBE Elektronik ve Haberleşme Anabilim Dalı Haberleşme Programında
Hazırlanan**

YÜKSEK LİSANS TEZİ

Tez Danışmanı: Yrd. Doç. Dr. Ünal KÜÇÜK

İSTANBUL, 2008

İÇİNDEKİLER

Sayfa

SİMGE LİSTESİ	v
KISALTMA LİSTESİ	vi
ŞEKİL LİSTESİ	ix
ÇİZELGE LİSTESİ	xi
ÖNSÖZ	xii
ÖZET	xiii
ABSTRACT	xiv
1. GİRİŞ	15
2. İNTERNET PROTOKOLÜ ÜZERİNDEN SES AKTARIMI (VoIP)	17
2.1 Genel Bilgiler	17
2.1.1 OSI Referans Modeli	17
2.1.1.1 Verinin Kapsüllenmesi	18
2.1.1.2 Katmanlar ve Açıklamaları	19
2.1.2 TCP/IP Referans Modeli	20
2.1.2.1 TCP/IP Referans Modeli Katmanları	21
2.1.3 Ağ Teknolojileri	23
2.1.3.1 PSTN(Kamusal Anahtarlama Telefon Ağı)	24
2.1.3.2 Paket Anahtarlama Ağlar	25
2.2 VoIP Tanımı ve Avantajları	25
2.3 VoIP Senaryoları	27
2.4 VoIP Çalışma Şekli	28
2.5 VoIP Sistemlerde Kullanılan Kodekler	28
2.5.1 Kodek Değerlendirme Parametreleri	30
2.5.1.1 Kalite ve Kapasite	30
2.5.1.2 Kodlama Gecikmesi	31
2.5.1.3 Karmaşıklık ve Maliyeti	32
2.5.1.4 Hata Duyarlılığı	33
2.6 VoIP Servis Kalitesi(QoS)	33
2.6.1 Gecikme(Delay)	33
2.6.2 Yankı(Echo)	34
2.6.3 Net Çıkış Hızı(Throughput)	34
2.6.4 Paket Kayıp Oranı	34
2.6.5 Gürültü	34
2.7 VoIP Protokolleri	35
2.7.1 H.323	36
2.7.2 MGCP	36
2.7.3 SIP	37
2.7.4 RSVP	37
2.7.5 RTP	38
3. OTURUM BAŞLATMA PROTOKOLÜ (SIP)	39
3.1 SIP Bileşenleri	42

3.1.1	SIP Kullanıcı Arabirimi.....	42
3.1.2	SIP Sunucusu.....	45
3.1.3	SIP Geçiti.....	46
3.2	SIP Adreslemesi	47
3.3	SIP Mesaj Yapısı	48
3.3.1	Başlangıç Satırı.....	48
3.3.2	Başlık Bölümü	48
3.3.2.1	Genel Başlık Kısımları	49
3.3.2.2	Talep Mesajına Özgü Başlık Kısımları	52
3.3.2.3	Cevap Mesajına Özgü Başlık Kısımları	53
3.3.2.4	Mesaj Gövdesi Başlık Kısımları.....	54
3.3.3	Mesaj Gövdesi	55
3.3.3.1	Oturum Tanımlama Protokolü(SDP).....	55
3.4	SIP Mesajları	59
3.4.1	SIP Talep Mesajları	59
3.4.1.1	INVITE.....	59
3.4.1.2	REGISTER	59
3.4.1.3	BYE	59
3.4.1.4	ACK.....	60
3.4.1.5	CANCEL	60
3.4.1.6	OPTIONS	60
3.4.1.7	REFER.....	60
3.4.1.8	SUBSCRIBE	60
3.4.1.9	NOTIFY	60
3.4.1.10	MESSAGE	61
3.4.1.11	INFO.....	61
3.4.1.12	PRACK.....	61
3.4.1.13	UPDATE	61
3.4.2	SIP Cevap Mesajları	61
3.4.2.1	1xx Cevap Mesajları.....	62
3.4.2.2	2xx Cevap Mesajları.....	62
3.4.2.3	3xx Cevap Mesajları.....	62
3.4.2.4	4xx Cevap Mesajları.....	62
3.4.2.5	5xx Cevap Mesajları.....	63
3.4.2.6	6xx Cevap Mesajları.....	64
4.	GERÇEK ZAMANLI VERİ TAŞIMA PROTOKOLÜ(RTP).....	65
5.	WAV SES DOSYASI FORMAT YAPISI.....	69
6.	PROJEDE KULLANILAN PROGRAMLAMA DİLLERİ VE ARAÇLAR	71
6.1	Java	71
6.2	Java2 Micro Edition (J2ME)	73
6.2.1	J2ME Yapılandırıcıları (J2ME Configurations).....	73
6.2.1.1	CLDC (Connected Limited Device Configuration)	74
6.2.1.2	CDC (Connected Device Configuration)	75
6.2.2	J2ME Profilleri (J2ME Profiles)	76
6.2.2.1	MIDP (Mobile Information Device Profile).....	78
6.2.3	JSR (J2ME Spesifikasyonları).....	78
6.3	Sun WTK Yazılım Geliştirme Kiti.....	79
6.4	Eclipse Geliştirme Ortamı	80

6.5	Ethereal Ağ Analiz Aracı	80
7.	TAŞINABİLİR SİSTEMLERDE SIP SİNYALLEŞMESİ ARACILIĞI İLE VOIP İSTEMCİ UYGULAMASI	82
7.1	Tasarlanan SIP İstemcisinin Çalışma Yapısı.....	83
7.1.1	Çevrimdışı Durumunda İken	86
7.1.1.1	Oturum Açma	86
7.1.1.2	Oturum Kapatma	99
7.1.1.3	Kişi Listesini Görüntüleme.....	99
7.1.1.4	Kişi Ekleme	100
7.1.1.5	Kişi Silme	101
7.1.1.6	Anlık İleti Gönderme.....	102
7.1.1.7	Arama Başlatma	103
7.1.1.8	Hesap Bilgilerini Düzenleme	104
7.1.2	Çevrimiçi Durumunda İken	106
7.1.2.1	Oturum Açma	106
7.1.2.2	Oturum Kapatma	107
7.1.2.3	Kişi Listesini Görüntüleme.....	111
7.1.2.4	Kişi Ekleme	120
7.1.2.5	Kişi Silme	125
7.1.2.6	Anlık İleti Gönderme.....	131
7.1.2.7	Arama Başlatma	140
7.1.2.7.1	SIP Mimari Yapısı	140
7.1.2.7.2	RTP Mimari Yapısı	142
7.1.2.7.3	Gerçeklenen Arama Senaryoları.....	146
7.1.2.8	Hesap Bilgilerini Düzenleme	168
8.	SONUÇLAR.....	170
	KAYNAKLAR.....	172
	EKLER	174
	ÖZGEÇMİŞ.....	177

SİMGE LİSTESİ

bps	Saniyedeki Bit Sayısı
Gbit/sn	Saniyedeki Gigabit Sayısı
kb	Kilobit
KB	Kilobyte
kb/sn	Saniyedeki Kilobit Sayısı
kHz	Kilohertz
MB	Megabyte
Mbit/sn	Saniyedeki Megabit Sayısı
MIPS	Milyon İşlem Bölü Saniye
ms	Milisaniye
µs	Mikrosaniye
sn	Saniye

KISALTMA LİSTESİ

ACELP	Algebraic Code Excited Linear Prediction
ADC	Analog-to-Digital Converter
ADPCM	Adaptive Differential Pulse Code Modulation
APCM	Adaptive Pulse Code Modulation
API	Application Programming Interface
ARP	Address Resolution Protocol
ASCII	American Standard Code for Information Interchange
ATM	Asynchronous Transfer Mode
BFI	Bad Frame Indication
CDC	Connected Device Configuration
CELP	Code Excited Linear Prediction
CLDC	Connected, Limited Device Configuration
CNAME	Canonical Name
CS-ACELP	Conjugate-Structured Algebraic Code Excited Linear Prediction
CS-CELP	Conjugate-Structured Code Excited Linear Prediction
CSRC	Contributing Source Identifier
CVM	Compact Virtual Machine
DAC	Digital-to-Analog Converter
DMOS	Degradation Mean Opinion Score
DSP	Digital Signal Processor
ETSI	European Telecommunications Standards Institute
FTP	File Transfer Protocol
GSM	Global System for Mobile Communications
HTTP	Hyper Text Transfer Protocol
ICMP	Internet Control Message Protocol
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol
IM	Instant Messaging
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
ISDN	Integrated Services Digital Network
ISO	International Organization for Standardization
ISP	Internet Service Provider
ITU	International Telecommunications Union
ITU-T	International Telecommunications Union Telecommunication Standardization Sector
J2ME	Java2 Micro Edition
J2SE	Java2 Standart Edition
JCP	Java Community Process
JNI	Java Native Interface
JRE	Java Runtime Engine
JSR	Java Specification Request
JVM	Java Virtual Machine
KVM	K Virtual Machine

LAN	Local Area Network
LCD UI	Limited Connected Device User Interface
LD-CELP	Low-Delay Code Excited Linear Prediction
LPC	Linear Predictive Code
MAC	Media Access Controller
MCU	Multipoint Control Unit
MEGACO	Media Gateway Control
MGCP	Media Gateway Control Protocol
MIDP	Mobile Information Device Profile
MMUSIC	Multi-Party Multimedia Session Control
MOS	Mean Opinion Score
MTU	Maximum Transmission Unit
MP-MLQ	Multi Pulse Multi-Level Quantization
OSI	Open Systems Interconnection
PCM	Pulse Code Modulation
PCMA	Pulse Code Modulation A-Law
PCMU	Pulse Code Modulation μ -Law
PDA	Personal Digital Assistant
POP3	Post Office Protocol version 3
PSTN	Public Switched Telephone Network
PT	Payload Type
QoS	Quality of Service
RARP	Reverse Address Resolution Protocol
RFC	Request For Comments
RMI	Remote Method Invocation
RMS	Record Management System
RSVP	Resource Reservation Protocol
RTCP	Real-Time Control Protocol
RTP	Real-Time Transport Protocol
RTSP	Real-Time Streaming Protocol
SIP	Session Initiation Protocol
SDK	Software Development Kit
SDP	Session Description Protocol
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
SONET/SDH	Synchronous Optical Network and Synchronous Digital Hierarchy
SS7	Signalling System 7
SSRC	Synchronisation Source Identifier
TCP	Transmission Control Protocol
UA	User Agent
UAC	User Agent Client
UAS	User Agent Server
UDP	User Datagram Protocol
URI	Uniform Resource Identifier
URL	Universal Resource Locator
UTF-8	8-bit Unicode Transformation Format
VAD	Voice Activation Detection
VoATM	Voice over ATM
VoIP	Voice over IP
WAN	Wide Area Network
WAV	Wave

Wi-Fi
WTK

Wireless Fidelity
Wireless Tool Kit

ŞEKİL LİSTESİ

	Sayfa
Şekil 2.1 Veri paketlerinin OSI katmanlarından geçerken uğradıkları değişim.....	18
Şekil 2.2 OSI ve TCP/IP referans modellerinin karşılaştırılması.....	21
Şekil 2.3 Rejeneratif repetörün blok diyagramı.....	35
Şekil 2.4 Çoklu ortam ağ protokolleri	35
Şekil 3.1 SIP protokol kümesi	40
Şekil 3.2 Cisco 7970 ve Nortel 1535 donanımsal IP telefon.....	42
Şekil 3.3 CounterPath X-Lite yazılımsal IP telefon	43
Şekil 3.4 Windows Messenger 5.1 ve Gizmo5 yazılımsal IP telefon.....	43
Şekil 3.5 Sony Ericsson P990i Wi-Fi üzerinden SIP destekli cep telefonu	44
Şekil 3.6 Nokia N95 Wi-Fi üzerinden SIP destekli cep telefonu	44
Şekil 3.7 SIP kullanıcı birimleri ve sunucuları.....	45
Şekil 3.8 SIP geçitleriyle kurulmuş hibrid ağ yapısı	47
Şekil 4.1 RTP başlık yapısı.....	66
Şekil 5.1 WAV dosya formatı (Craig, 2005).....	70
Şekil 6.1 CLDC mimari yapısı	74
Şekil 6.2 CDC mimari yapısı.....	75
Şekil 6.3 J2ME yapılandırıcı ve profil mimari yapısı	77
Şekil 6.4 JSR örnek listesi	79
Şekil 6.5 Sun WTK emülatör arayüzü.....	79
Şekil 6.6 Eclipse geliştirme ortamı arayüzü	80
Şekil 7.1 Taşınabilir SIP istemcisi.....	84
Şekil 7.2 Çevrimdışı durumda oturum açma emülatör çıktısı	88
Şekil 7.3 Oturum açma SIP mesajları akış şeması	89
Şekil 7.4 Çevrimdışı durumda oturum kapatma emülatör çıktısı.....	99
Şekil 7.5 Çevrimdışı durumda kişi listesi emülatör çıktısı.....	100
Şekil 7.6 Çevrimdışı durumda kişi ekle emülatör çıktısı.....	101
Şekil 7.7 Çevrimdışı durumda kişi sil emülatör çıktısı	102
Şekil 7.8 Çevrimdışı durumda anlık ileti gönder emülatör çıktısı.....	103
Şekil 7.9 Çevrimdışı durumda arama başlat emülatör çıktısı.....	104
Şekil 7.10 Çevrimdışı durumda hesap bilgilerini düzenle emülatör çıktısı.....	105
Şekil 7.11 Çevrimiçi durumda oturum açma emülatör çıktısı.....	107
Şekil 7.12 Çevrimiçi durumda oturum kapatma emülatör çıktısı.....	109
Şekil 7.13 Çevrimiçi durumda kişi listesi emülatör çıktısı.....	110
Şekil 7.14 Çevrimiçi durumda kişi listesi emülatör çıktısı.....	111
Şekil 7.15 İstemcinin kişi listesindeki kullanıcılar	112
Şekil 7.16 Kullanıcı durum değişikliği (Çevrimiçi --> Dışarıda).....	113
Şekil 7.17 Kullanıcı durum değişikliği (Çevrimiçi --> Dışarıda) SIP mesajları akış şeması	113
Şekil 7.18 Kullanıcı durum değişikliği (Çevrimdışı --> Çevrimiçi)	114
Şekil 7.19 Kullanıcı durum değişikliği(Çevrimdışı-->Çevrimiçi) SIP mesajları akış şeması	114
Şekil 7.20 Kullanıcı durum değişikliği (Yemekte --> Dönecek)	116
Şekil 7.21 Kullanıcı durum değişikliği (Yemekte --> Dönecek) SIP mesajları akış şeması..	116
Şekil 7.22 Kullanıcı durum değişikliği (Meşgul --> Çevrimdışı)	117
Şekil 7.23 Kullanıcı durum değişikliği (Meşgul --> Çevrimdışı) SIP mesajları akış şeması	118
Şekil 7.24 İstemcinin kişi listesindeki kullanıcıların durum bilgileri.....	119
Şekil 7.25 Kullanıcıların durum bilgileri güncellenmiş istemci kişi listesi.....	119
Şekil 7.26 Çevrimiçi durumda kişi ekle emülatör çıktısı	120
Şekil 7.27 Kişi ekle SIP Mesajları akış şeması	121

Şekil 7.28 Kişi eklendikten sonra güncellenen kişi listeleri emülatör çıktısı	124
Şekil 7.29 Kişi listesinde mevcut bir kullanıcıyı tekrar ekleme emülatör uyarı çıktısı	125
Şekil 7.30 Çevrimiçi durumunda kişi sil emülatör çıktısı	126
Şekil 7.31 Kişi sil SIP mesajları akış şeması	127
Şekil 7.32 Kişi silindikten sonra güncellenen kişi listeleri emülatör çıktısı	130
Şekil 7.33 Çevrimiçi durumunda anlık ileti gönder emülatör çıktısı	131
Şekil 7.34 Anlık ileti gönderilemedi emülatör çıktısı	132
Şekil 7.35 Anlık ileti gönderilemedi SIP mesajları akış şeması	132
Şekil 7.36 Karşı kullanıcının oturum açma ekranı	134
Şekil 7.37 İki istemci arasındaki anlık iletişime	135
Şekil 7.38 Anlık ileti gönderildi SIP mesajları akış şeması	135
Şekil 7.39 Çoklu anlık iletişime	140
Şekil 7.40 Arama başlat SIP mimari yapısı	141
Şekil 7.41 Arama başlat RTP mimari yapısı	142
Şekil 7.42 Arama başlat emülatör çıktısı	146
Şekil 7.43 Aranıyor – Çalıyor emülatör çıktısı	147
Şekil 7.44 Aranana kullanıcı aramayı kabul etmedi ekran çıktısı	148
Şekil 7.45 Arama kabul edilmedi emülatör çıktısı	149
Şekil 7.46 Arama kabul edilmedi SIP mesajları akış şeması	149
Şekil 7.47 İstemci aramayı iptal etti emülatör çıktısı	153
Şekil 7.48 İstemci aramayı iptal etti SIP mesajları akış şeması	153
Şekil 7.49 Aranana kullanıcı aramayı cevapladı ekran çıktısı	156
Şekil 7.50 Arama cevaplandı emülatör çıktısı	157
Şekil 7.51 Arama cevaplandı SIP mesajları akış şeması	158
Şekil 7.52 RTP paket akış şeması	161
Şekil 7.53 Arama özet bilgisi	162
Şekil 7.54 Protokol hiyerarşi istatistiği	162
Şekil 7.55 Arama sırasında sistem IO grafiği	163
Şekil 7.56 RTP akışları listesi	163
Şekil 7.57 VoIP aramalar listesi	163
Şekil 7.58 RTP akış analizi (A->B)	164
Şekil 7.59 RTP akış analizi (B->A)	164
Şekil 7.60 Jitter grafiği (A->B)	165
Şekil 7.61 Jitter grafiği (B->A)	165
Şekil 7.62 İstemci görüşmeyi sonlandırdı emülatör çıktısı	166
Şekil 7.63 İstemci görüşmeyi sonlandırdı SIP mesaj akışı	166
Şekil 7.64 Aranana kullanıcı görüşmeyi sonlandırdı emülatör çıktısı	167
Şekil 7.65 Aranana kullanıcı görüşmeyi sonlandırdı SIP mesaj akışı	168
Şekil 7.66 Çevrimiçi durumunda hesap bilgilerini düzenle emülatör çıktısı	169

ÇİZELGE LİSTESİ

	Sayfa
Çizelge 2.1 Konuşma kodeklerinin kapasiteleri ve kaliteleri	31
Çizelge 2.2 Konuşma kodeklerinde algoritmik gecikmeler	31
Çizelge 2.3 Konuşma kodeklerinde işlem karmaşıklığı	32
Çizelge 3.1 Başlık bölümü içeriği	49
Çizelge 3.2 SIP başlık kısımlarının kompakt formları	54
Çizelge 3.3 SDP içeriği	55

ÖNSÖZ

Her geçen gün gerekliliği ve kullanımı katlanarak artan “Taşınabilir Sistemlerde VoIP İstemci Uygulamaları” na yönelik hazırladığım bu projenin yapım aşamasında karşılaştığım zorlukların üstesinden gelmemde bana yardımcı olan aileme ve tez çalışması süresince benden desteğini esirgemeyen çok değerli hocamız Yrd. Doç. Dr. Ünal KÜÇÜK’ e sonsuz teşekkürlerimi sunarım.

ÖZET

Kablosuz iletişim sistemlerin, kablolu iletişim sistemlerin yerini almaya başladığı günümüz haberleşme sektöründe, taşınabilirlik her geçen gün daha çok önem kazanmaktadır. Bu proje ile taşınabilir sistemlerde SIP sinyalleşmesi aracılığı ile bir VoIP istemcisi geliştirmek amaçlanmıştır.

Bu uygulama, IBM' in geliştirmiş olduğu Eclipse geliştirme ortamında, J2ME platformunda, Sun WTK 2.5.2 emülatörü kullanılarak geliştirilmiştir. Sunucu – istemci mimari yapısı temel alınarak tasarlanmıştır. Sinyalleşme SIP protokolü ile sağlanırken, gerçek zamanlı veri transferi RTP protokol yapısı oluşturularak sağlanmıştır.

Bu projede, tasarlanan SIP ve RTP mimari yapısı ile istemci, hesap oluşturma, oturum açma, oturum kapatma, kişi listesini görüntüleme, anında kişi durum bilgisi değişikliğini alma, kişi ekleme, kişi silme, hesap bilgilerini düzenleme, anlık ileti gönderebilme ve arama başlatma uygulamalarını gerçekleyebilmiştir.

Anahtar Kelimeler: VoIP, SIP, RTP, J2ME, taşınabilir cihazlar

ABSTRACT

Nowadays, mobility gains much more importance day by day in telecommunication sector in which wireline communication systems are being replaced with wireless communication systems. With this project, the aim was to design a VoIP client on mobile devices by using SIP signalling.

This project was developed in Eclipse Integrated Development Environment with J2ME platform by using Sun WTK 2.5.2 emulator. The project was designed based on server - client architectural structure. While signalling was handled by SIP protocol, real-time data transfer was handled by developed RTP protocol structure.

In this project, the client gained the ability to create an account, login, logout, display the contact list, get presence of the contacts at the time the contacts changes their status, add contact, remove contact, edit account settings, chat, make a call within designed SIP and RTP architectural structure.

Keywords: VoIP, SIP, RTP, J2ME, mobile devices

1. GİRİŞ

Gelişen iletişim teknolojileri kapsamında, devre anahtarlama ağıların yerini paket anahtarlama ağı almıştır. Paket anahtarlama sistemlerinde, güvenilirlik, iletilen verinin kalitesi ve paket anahtarlama ağı ile devre anahtarlama ağı arasında kullanılan farklı standartların birbirleriyle uyumu gibi problemler ortaya çıkmıştır. Bu problemleri çözmek ve paket anahtarlama sistemlerinde işleyişi gerçekleştirmek için birçok protokol standardı geliştirilmiştir.

Bu protokollerden H.323, MGCP(Medya Geçidi Kontrol Protokolü - Media Gateway Control Protocol) ve SIP(Oturum Başlatma Protokolü - Session Initiation Protocol) gibi protokoller, kontrol ve sinyalleşme ile ilgili standartları tanımlamak; RTP(Gerçek Zamanlı Taşıma Protokolü - Realtime Transport Protocol), RTCP(RTP Kontrol Protokolü - Real-Time Control Protocol) ve RSVP(Kaynak Rezervasyon Protokolü - Resource Reservation Protocol) gibi protokoller de verinin kaliteli iletimini sağlamak için geliştirilmiştir.

H.323 protokolünün, IP uygulamaları için yeteri kadar kullanışlı ve esnek olmaması, SIP protokolünün geliştirilmesine sebep olmuştur. SIP protokolü, HTTP ve SMTP gibi metin tabanlı protokoller temel alınarak oluşturulmuş bir protokoldür. Bu metin tabanlı mimari yaklaşım, SIP protokolüne, ihtiyaçlara göre kural eklenip, değiştirilebilen esnek bir yapı kazandırmıştır. ASN.1 mesajlaşma yerine basit olan, HTTP/1.1 de yapılandırılmış ASCII kullanır. Böylece, SIP sinyalleşmelerinde, şifre çözümü ve hata denetimi kolaylaşmakta, ve küçük ayarlamalar ile web tipi uygulamalar SIP servisini destekleyebilmektedir. SIP, URL(Evrensel Kaynak Konumlandırıcı), E.164 (North American Numbering Plan Addressing) standartlarını tam desteklemektedir. Böylece, SIP modelinde kullanıcının e-posta adresi ile telefon adresi aynı olabilmektedir. Bu özellikleri sayesinde SIP protokolü, internet telefonculuğunda kullanılan en yaygın protokol haline gelmiştir. Bu yüzden, daha eski VoIP sistemleri H.323' ü kullanırken, günümüzdeki mevcut VoIP sistemlerinin çoğu SIP standardını izlemektedir. SIP protokolünün bu üstünlüklerinden dolayı, bu projede de sinyalleşme protokolü olarak SIP protokolü kullanılmıştır.

Gereksinimler doğrultusunda, günümüz haberleşme sektöründe taşınabilirlik her geçen gün daha çok önem kazanmaya başlamıştır. Kablolu iletişim sistemlerin, kullanıcının konumunu kısıtlaması nedeniyle kablosuz iletişim sistemlerine ihtiyaç duyulmuştur. Paket anahtarlama ağıların, devre anahtarlama ağıların yerini almasından sonra, kablosuz iletişim sistemlerinin

tasarlanması mümkün hale gelmiştir. Bu projede, tam da bu ihtiyaca cevap olacak şekilde, kablosuz sistemlerde paket anahtarlamalı bir uygulama gerçekleştirilmeye çalışılmıştır.

Bu proje ile taşınabilir sistemlerde SIP sinyalleşmesi aracılığı ile bir VoIP istemcisi geliştirmek amaçlanmıştır. Bu uygulama, IBM' in geliştirmiş olduğu Eclipse geliştirme ortamında, J2ME platformunda, Sun WTK 2.5.2 emülatörü kullanılarak geliştirilmiştir. Sunucu – istemci mimari yapısı temel alınarak tasarlanmıştır. Sinyalleşme SIP protokolü ile sağlanırken, gerçek zamanlı veri transferi RTP protokol yapısı oluşturularak sağlanmıştır. Bu projede, tasarlanan SIP ve RTP mimari yapısı ile istemci, hesap oluşturma, oturum açma, oturum kapatma, kişi listesini görüntüleme, anında kişi durum bilgisi değişikliğini alma, kişi ekleme, kişi silme, hesap bilgilerini düzenleme, anlık ileti gönderebilme ve arama başlatma uygulamalarını gerçekleyebilmiştir.

Bitirme tezi kitabının ileriki bölümlerinde internet üzerinden ses aktarımından, SIP protokol yapısından, RTP protokol yapısından, WAV ses dosya formatı yapısından, projede kullanılan programlama dilleri ile araçlardan, taşınabilir sistemlerde SIP sinyalleşmesi aracılığı ile VoIP istemci uygulamasının çalışmasından ve uygulamanın sonucuna ait istatistik verilerin analizlerden detaylı bir şekilde bahsedilecektir.

2. İNTERNET PROTOKOLÜ ÜZERİNDEN SES AKTARIMI (VoIP)

İnternet üzerinden sesin taşınması (VoIP) son yıllarda çok popüler bir konu haline gelmiştir. Özellikle, ekonomik açıdan çeşitli avantajlara sahip olan VoIP, varolan internet altyapısını kullanması ve ek maliyet getirmemesi açısından şirketler için çok cazip bir ortam oluşturmaktadır. Ayrıca ses ve veri hizmetlerinin aynı ağ üzerinden paket anahtarlama mantığıyla verilmesi mevcut band genişliğinin çok verimli kullanılmasını sağlamıştır.

VoIP gerçek zamanlı paket iletimini gerektiren bir sistemdir. Dolayısıyla kayıp paketlerin alıcı tarafından tekrar istenmesi söz konusu olamaz. Bu durumda kullanılacak en iyi protokol çifti IP (Internet Protocol)' nin üzerinde çalışan UDP (User Datagram Protocol) ve onun da üzerinde çalışan RTP (Real Time Transport Protocol) olarak görülür.

İşaretleşme fazında ise, ITU-T' nin geliştirmiş olduğu H.323 ile IETF' nin geliştirdiği SIP' in (Session Initiation Protocol) çekişmesi gözden kaçmamaktadır. SIP, H.323' e göre daha yeni bir protokol olduğu için, H.323' ün cevap veremediği ölçeklenebilirlik ve karmaşıklık problemlerini ortadan kaldırmıştır. Ayrıca SIP, metin bazlı bir yapıya sahip olduğundan bu protokol ile çalışan uygulama geliştirmek H.323' e göre daha kolaydır. Bütün olumlu yanlarına rağmen, VoIP' in aşması gereken çeşitli engeller halen bulunmaktadır. Bu engeller, hizmet kalitesinin aynı düzeyde tutulması ve sistemde açılan UDP portlarının yarattığı güvenlik açıklarıdır. Ayrıca veri ağlarıyla ortak kullanılması da VoIP' in veri ağının etkilendiği tüm olumsuzluklardan etkilenmesine yol açacaktır.

2.1 Genel Bilgiler

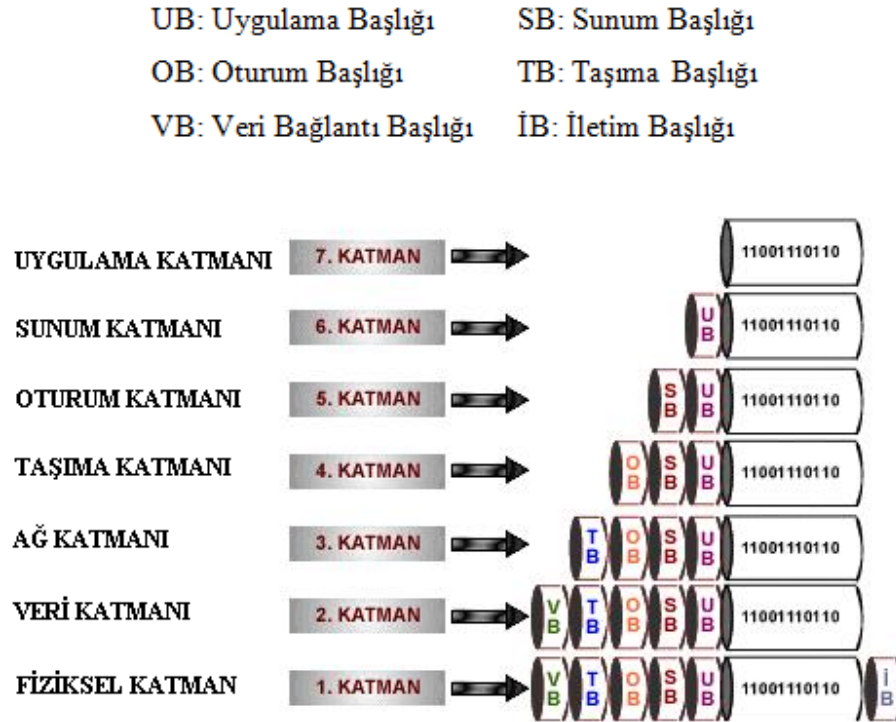
Bu bölümde VoIP konusunun rahatlıkla anlaşılabilmesi için gerekli olan genel bilgiler verilecektir. Bu bilgiler arasında, OSI referans modeli, TCP/IP referans modeli ve günümüzde kullanılan ağ teknolojileri hakkında bilgi verilecektir.

2.1.1 OSI Referans Modeli

OSI (Open System Interconnection) modeli, 1984 yılında ISO (International Standards Organization) tarafından geliştirilmiş bir modeldir. Bu modelde, hangi ağ protokolünün hangi kurallara bağlı olarak çalışacağını düzenlenir ve haberleşmek isteyen iki ağ bileşeni arasında yapılan bilgisayar haberleşmesini, her birinde sanal olarak farklı görevler tanımlanan ve sırayla her bileşenin kendi görevini yerine getirdiğinde ancak bir haberleşmenin oluşabildiği 7

katmanlı bir çalışma şeklidir. Diğer bir deyişle, farklı ağ elemanları (bilgisayarlar, ağ geçitleri veya yönlendiriciler gibi) kendi içlerinde anlaşmak için farklı haberleşme protokolleri kullansalar bile birbirleriyle konuşurken bu modelin kullanılması yoluyla birbirlerinden haberdar olabilirler ve karşılıklı olarak veri alışverişinde bulunabilirler.

OSI referans modeli 7 katmandan oluşmaktadır. Bu yapı, katmanlarda hangi protokollerin kullanılacağını ve verinin nasıl ele alınacağını tanımlar. Şekil 2.1’ de paketlerin katmanlardan geçerken geçirdikleri değişim gösterilmiştir.



Şekil 2.1 Veri paketlerinin OSI katmanlarından geçerken uğradıkları değişim

Her katmanın ayrı bir ismi ve ayrı bir görevi vardır. Bu katmanlar yukarıdan aşağıya doğru uygulama katmanı, sunum katmanı, oturum katmanı, taşıma katmanı, ağ katmanı, veri katmanı ve fiziksel katman olarak isimlendirilirler.

2.1.1.1 Verinin Kapsüllenmesi

Bir uygulama mesaj oluşturacağı zaman, veriye uygulama katman başlığı ekler ve bir alttaki seviyeye mesajı gönderir. Bu proses her katmanda yeni başlıklar eklenerek, mesaj en alt katmana gelinceye kadar devam eder. Veriye eklenen başlıklar verinin farklı isimlerde tanımlanmasına neden olur. Bunlar, mesaj, paket, segment, datagram, çerçeve (frame) ve hücre gibi adlar alırlar.

2.1.1.2 Katmanlar ve Açıklamaları

OSI referans modeli 7 katmandan oluşur. Bu modüler yapı yeni protokollerin geliştirilmesini ve sistemlere entegre olmasını kolaylaştırmıştır. Örneğin, işaretleri taşıyan fiziksel arayüz değişse bile bu referans modeli baz alan sistemde sadece bir veya iki katmanda çalışan protokoller değiştirilerek sistemler kolaylıkla gerçekleştirilebilir. Aşağıda OSI katmanları makine seviyesinden kullanıcıda çalışan program seviyesine doğru olmak üzere açıklanmıştır.

Fiziksel Katman: Bu katman verinin fiziksel bir haberleşme kanalında iletilmesini sağlar. Bitler, eğer kablolar üzerinden aktarılıyorsa elektriksel büyüklüklere; özellikle belirli gerilim değerlerine dönüştürülür. Eğer enformasyon, hava veya boşluk gibi bir ortamdan iletilecekse, bu durumda bitle genellikle yüksek frekanslı taşıyıcı dalgaya genlik, frekans veya faz değişimi şeklinde bindirilerek iletilir.

Veri (Bağlantı) Katmanı: Bu katman çerçevenin (frame) hatasız olarak iletiminden sorumludur. Değişken uzunluktaki veriye çeşitli uzunlukta 1' ler ve 0' lar ekleyerek çerçeve büyüklüğünü 64 bitte sabitler. Bu katmanda hem hata denetimi hem de hata düzeltmesi yapılır.

Ağ Katmanı: Bu katman veri paketinin düzgün yönlendirilmesinden sorumludur. Bu noktada bağlantı üç şekilde olabilir:

Kalıcı bağlantı: Operatör tarafından girilmiş sabit yönlendirme tablolarının oluşturduğu bağlantı türüdür.

Bağlantı yönelimli (Connection Oriented) bağlantı: Bu bağlantı türü oturum açılınca oluşturulur; oturumun sonlandırılmasıyla bağlantı da sonlanır. Haberleşme boyunca o oturum için bir virtüel devrenin kurulması söz konusudur.

Bağlantısız (Connectionless) bağlantı: Her paket gideceği hedef adresini içerir. Herhangi bir virtüel devre kurulmaz. Dolayısıyla yol kurma gecikmesi oluşmaz. Gerçek zaman uygulamalarında kullanılır.

Taşıma Katmanı: Bu katman, enformasyonun uçtan uca (end to end) doğru bir şekilde iletilmesinden sorumludur. Burada akış kontrol mekanizmaları devreye girer. Veri paketleri doğru şekilde yerine ulaşmazsa tampon bellekte (buffer) tutulan veri paketleri tekrardan gönderilir. Ayrıca bu katman kendisine gelen veri eğer doğru alınmışsa karşıdaki bilgisayara ya da ağ aygıtına bir onay mesajı da göndermekle yükümlüdür.

Oturum Katmanı: Bu katman farklı makinelerde (host) bulunan kullanıcıların oturum

açmasına izin verir. Bu olay genellikle kullanıcı adı - parola prosedürü şeklinde cereyan eder. Ayrıca oturum katmanı iletişimin kurulması için kendisine verinin ulaşmasının ardından işaretleşme işlemini başlatır.

Sunum Katmanı: Bu katman karşıdaki bilgisayarla alınıp verilecek verinin formatına karar verir. Burada sentaks ve şematik kontrol mekanizmaları çalışır. Protokollerin birbirleriyle haberleşebilmeleri için birbirlerinin dilinden anlamaları gerekmektedir. Bunun için protokoller birbirlerinin diline bu katmanda çevrilirler. Ayrıca çeşitli sıkıştırma algoritmaları da (Huffman gibi) bu katmanda kullanılır.

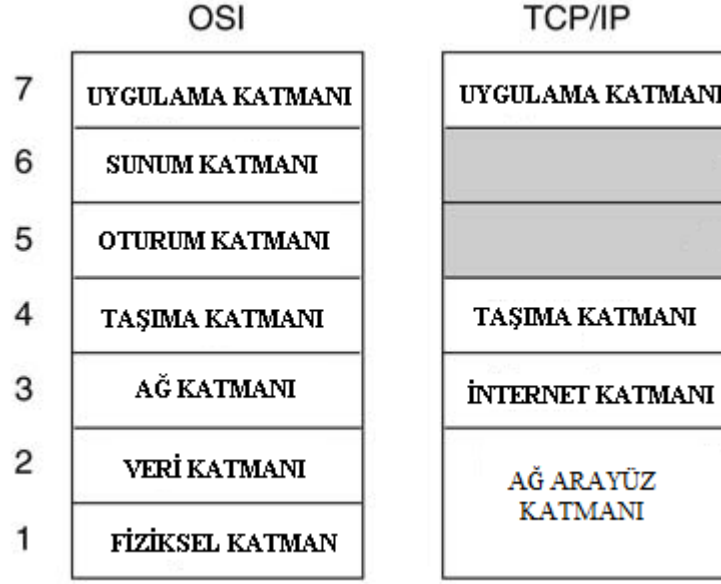
Uygulama Katmanı: Konum olarak en yukarıda bulunan katmandır. Kullanıcıların çalıştırdıkları programlar ve bu programların çalıştırdıkları protokoller bu katmanda bulunurlar. Bu katman, kullanıcı programlarının, üzerinde çalıştığı makineden ve alttaki katmanlardan bağımsız olmasına izin verir.

2.1.2 TCP/IP Referans Modeli

İnternet üzerinden ses aktarımını kavrayabilmek için gerekli olan ilk bilgi TCP/IP referans modelinin nasıl çalıştığıdır. Bu model geliştirilirken üç önemli kavram düşünülmüştür. Bunlar,

- Platformdan bağımsız olmak;
- Sistemin belirli kısımları çökse bile, sistemin çalışan diğer kısımları aracılığıyla aktivitesini sürdürebilmesi;
- Sistemin çok yüksek bir veri trafiğini kaldırabilmesidir.

Günümüzde en çok kullanılan model TCP/IP modelidir. OSI referans modelinde olduğu gibi TCP/IP modeli de katmanlı yapıya sahiptir; fakat katman sayısı OSI referans modelinden farklı olarak dördür. Bu katmanlar, uygulama katmanı, taşıma katmanı, internet katmanı, ağ arayüz katmanı olarak adlandırılırlar.



Şekil 2.2 OSI ve TCP/IP referans modellerinin karşılaştırılması

2.1.2.1 TCP/IP Referans Modeli Katmanları

OSI referans modelinin aksine TCP/IP referans modeli, “birden çok katmanın yaptığı işi yapan tek katman” esasına göre geliştirilmiştir. Aşağıda bu katmanlar açıklanmıştır:

Uygulama Katmanı: Bu katmanda ağ üzerinde iletişim kurabilme yeteneğine sahip tüm yazılımlar bulunur. Uygulama katmanında çalışan protokoller HTTP (HyperText Transfer Protocol), SMTP (Simple Mail Transfer Protocol), FTP (File Transfer Protocol) gibi protokollerdir. Bu protokoller aşağıda kısaca açıklanmıştır.

HTTP: İnternet’ te web sayfaların içeriğini görüntülemek için kullanılır.

FTP: İnternet’ te hızlı dosya transferi yapılması için geliştirilmiştir. Şifreleme yapılmadığından güvenli değildir. Son yıllarda FTP’ nin yerini verileri şifreleyerek yollayan SSH (Güvenli Kabuk) hizmeti almıştır.

SNMP: Belirli bir ağ üzerindeki cihazların birbirleriyle haberleşmelerini ve ortak bir dil kullanmalarını sağlamak için geliştirilmiştir.

SMTP: Bir e-posta yollanacağı zaman bu protokol kullanılır. E-postanın hangi adresten geldiğine ve hangi adrese gideceğine ilişkin işlemler bütünü kontrol eder.

Telnet: Karşı makineye bağlanmak için kullanılan bir protokoldür. Bu protokol kullanılırken her karakter tek tek gönderilir. Herhangi bir şifreleme yapılmadığından güvenli değildir.

POP3: Bir e-posta alınacağı zaman kullanılan bir protokoldür. Özellikle e-postalarını web üzerinden değil de bazı yardımcı programlar (Lotus Notes gibi) aracılığıyla kontrol etmek isteyen kullanıcılar için tasarlanmıştır.

Taşıma Katmanı: İki bilgisayar (host) arasındaki tüm iletişimin kurulma sorumluluğunu üstlenen veya diğer bir deyişle oturumun açılmasını sağlayan katmandır. Bu katmanda iki adet protokolden birisi çalışır. Bunlar TCP (Transmission Control Protocol) ve UDP (User Datagram Protocol) olarak adlandırılan protokollerdir.

TCP: Bu protokol, verilerin doğru adrese ulaşip ulaşmadığını kontrol eder. Bu kontrolün yapılabilmesi için karşı bilgisayardan onay (ACK) mesajının gelip gelmediğine bakılır. Bağlantı yönelimli bir protokoldür. Ayrıca hata sezme mekanizmalarıyla uçtan uca verinin garantili bir şekilde ulaşmasını sağlar.

UDP: Bu protokol TCP' nin yaptığı denetimleri yapmaz. Dolayısıyla veri alışverişi çok daha hızlı gerçekleşir. Garantili veri hizmeti sunmaz. Bağlantı kurma ve çözme süreci olmadığı için bu süreçlerden kaynaklanan gecikmeler de oluşmaz. Dolayısıyla gerçek zaman uygulamalarında tercih edilir.

İnternet Katmanı: İletim katmanından gelen verilerin IP paketleri haline dönüştürülüp yönlendirilmesinden sorumludur. Ayrıca bilgisayarlar arasındaki iletişimin nasıl gerçekleşeceğinden de sorumludur. Bu katmanda çalışan başlıca protokoller dört tanedir. Bunlar IP (Internet Protocol), ARP (Adress Resolution Protocol), ICMP (Internet Control Message Protocol) ve IGMP (Internet Group Management Protocol)' dir.

IP: Ağ üzerindeki herhangi bir cihaza ulaşılacağı zaman sanal bir adres üretilir. Paket anahtarlamalı ağ sistemlerinin birbirlerine bağlanabilmesi için tasarlanmıştır. IP paketlerine datagram denir. Bu protokolde paketlerin bölünmesi ve tekrardan birleştirilmesi de tanımlıdır.

ARP: Bir ağdaki makinelerin sabit MAC adreslerinin belirlenmesinde ve bu makinelerin birbirleriyle konuşurken MAC adreslerini kullanmalarını sağlayan kuralları içeren bir protokoldür.

ICMP: Verilerin yerine ulaşip ulaşmamasıyla ilgili mesajların yollanmasından sorumlu olan protokoldür. İletim esnasında oluşacak sorunlarda ICMP protokolü devreye girerek gelen verinin bozuk olduğunu belirtir. Böylece verinin tekrar

gönderilmesi için gerekli olan mekanizma tetiklenmiş olur.

IGMP: Bu protokol bir veri paketinin birden çok bilgisayara gönderilmesi işleminden (multicasting) sorumludur.

Ağ Arayüz Katmanı (Network Interface Layer): Verilerin elektriksel işaretler yoluyla duyuların algılama sınırlarının ötesinde çeşitli iletim ortamlarından fiziksel olarak aktarılmasını sağlar. ATM ya da Ethernet gibi protokoller bu katmanda çalışırlar. Ayrıca MAC adreslerinin nasıl kullanılacağı da bu katmanda tanımlanmıştır.

ATM (Asenkron Transfer Modu): ATM anahtarlama özellikle ISDN (Integrated Services Digital Network)' de kullanılan bir tekniktir. Bu teknik ile; ses veri ve görüntü aynı ağ üzerinden taşınabilir. IP ağlarında veri paketlere ayrılırken ATM ağlarında veri hücre denilen 53 byte lık bölümlere ayrılır. Bu bölümler 5 byte lık başlık alanına ve 48 byte lık veri alanına sahiptirler. ATM bağlantı yönelimli bir bağlantıdır ve 1.544 Mbit/sn ile 622 Mbit/sn arasındaki hızlarda çalışabilir.

SONET/SDH: SONET optik iletim standartlar ailesine verilen genel isimdir. SONET' de taşınan birim çerçevelerdir. Her çerçeve 810 byte lık (90 byte * 9 satır) enformasyonu taşır. Her çerçeve satır satır 125 µs' de taşınır; yani bir saniyede yaklaşık 8000 çerçeve taşınmış olur. Bu da 51.84 Mbit/sn' lik bir kanal kapasitesine karşılık düşer.

Ethernet: IEEE 802.3 standardıyla belirlenmiştir. Lokal alan ağlarında çok sık kullanılan bir teknolojidir. Ethernet standardı 10 Mbit/sn' lik hıza izin verirken, Fast Ethernet standardı 100 Mbit/sn' lik hızlara olanak verir. Günümüzde "Gigabit Ethernet" standardı yaygınlaşmaya başlamıştır ve bu standart 1 Gbit/sn' lik enformasyon hızlarında veri iletim sağlar.

2.1.3 Ağ Teknolojileri

Sesin aktarılmasında ve telefon haberleşmesinde kullanılan ağ teknolojileri uygulanan anahtarlama tekniği bakımından iki ana gruba ayrılabilirler.

- Devre Anahtarlama Sistemleri
 - o Manuel Anahtarlama Sistemleri
 - o Otomatik Anahtarlama Sistemleri
 - Elektromekanik Sistemler

- Elektronik Sistemler
 - Analog Sistemler
 - Sayısal Sistemler
- Hafızalı Anahtarlama Sistemleri
 - o Mesaj Anahtarlama Sistemleri
 - o Paket Anahtarlama Sistemleri
 - o Hücre Anahtarlama Sistemleri

Bu bölümde, genel olarak telefon haberleşmesinde hakim olan iki ağ teknolojisi örneğinden bahsedilecektir. Bunlardan ilki, devre anahtarlama bir sistem olan PSTN ve diğeri ise paket anahtarlama sistemidir. Bunların dışında bu bölümde söz edilmeyecek olan hücre anahtarlama teknikleri de vardır. Özellikle bu tür hücre anahtarlama teknikleri ATM (Asenkron Transfer Modu) ağlarında kullanılırlar.

2.1.3.1 PSTN(Kamusal Anahtarlama Telefon Ağı)

Günümüzde telefon ağlarında kullanılan ana teknoloji devre anahtarlama sistemidir. Bu sistemde haberleşmeyi yapan uç noktaları arasında bazen elektronik bazen de elektromekanik şekilde kurulan devre yolları kullanılır. Haberleşme boyunca herhangi bir konuşma olmasa dahi devre yolu hizmette kalır. Haberleşme yolu ya da devre üç farklı boyutta kurulabilir:

1. Uzak ullaamalı sistemler
2. Zaman ullaamalı sistemler
3. Frekans ullaamalı sistemler

Uzak ullaamalı sistemler: Bu anahtarlama tekniğinde sistemde elektronik veya elektromekanik devreler üzerinden uzayda ayrılmış fiziksel yollar kurulur. Bu fiziksel yollar seçici denilen döner bir kol ve girişleri tarayan fırçadan oluşmaktadır.

Zaman ullaamalı sistemler: Bu anahtarlama tekniğinde, sistemde farklı zaman dilimlerinde farklı zaman kanalları oluşturulur. Sayısal santrallerde görülen bu teknikte eşlenecek giriş ve çıkışa hükmeden kapılar aynı zaman diliminde kontrol edilirler.

Frekans ullaamalı sistemler: Bu teknikte ise frekans domeninde paylaşılan ortak yollar oluşturulur. Girişlerin herbirindeki işaret farklı bir taşıyıcıya bindirilmekte ve anahtarlama demodülatörün frekansının istenen girişin bindirildiği taşıyıcıya ayarlanmasıyla yapılır.

2.1.3.2 Paket Anahtarlama Ağılar

Bu metodu kullanan ağlarda iletilen veri ya da değişik sayıdaki sembol grupları paket adı verilen parçalara bölünerek iletilirler. Bir mesajı oluşturan her paket birbirinden bağımsız olarak iletilir. Her paket istenilen hedefe farklı yollardan sıraları karışık bir şekilde ulaşabilir. Dolayısıyla her pakette hedef adres bilgisi ve sıra bilgisi bulunmak zorundadır. İletilen bu paketler karşı taraftaki bilgisayarda tekrardan birleştirilmek suretiyle ana mesajı oluştururlar. Paket anahtarlama, veri iletimi açısından çok verimli bir yöntemdir. Bu yöntemle bir iletim kanalı eşzamanlı birden çok haberleşmenin yapılmasına olanak sağlar. İleride de inceleneceği gibi VoIP teknolojisi de paket anahtarlama tekniğini kullanır. Tabii ki, ses veya görüntünün gönderilmesi sırasında yapılması gerekli olan senkronizasyon işlemi ve gerçek zamanlı iletimde oluşacak gecikmeleri azaltmak internet üzerinden yapılan ses ve görüntü yollama işleminin önünde aşılması gereken en önemli engellerdir (Ibe, 2002).

2.2 VoIP Tanımı ve Avantajları

VoIP(Voice over IP), internet protokolü üzerinden ses iletimi için kullanılan bir protokoldür. Diğer bir deyişle, gerçekte bir telefon görüşmesi yapmak için telefon ağının değil bilgisayar ağlarının kullanılması anlamına gelir. Voice over IP internet, intranet, Local Area Network(yerel alan ağları) gibi IP(Internet Protokol) kullanan herhangi bir veri ağı üzerinde başarıyla gerçekleştirilebilir. Her ne kadar teknolojinin ismi "ses" ile bağdaşsa da genelde "ses ve çoklu-ortam" olarak genişletilmekte; sesi olduğu kadar faks, video konferans gibi çoklu ortam uygulamalarını da gerçek zamanlı olarak karşılamaktadır.

Sesin VoIP cihazları sayesinde sıkıştırılarak eş zamanlı olarak hedef noktaya kadar internet üzerinden veri olarak gönderilmesi, hedef noktada sıkıştırılmış (kompres) olarak gelen sesin sağlıklı ve eş zamanlı olarak açılarak (dekompres) görüşmenin yapılmasını sağlayan protokollerin bütünüdür (Ibe, 2002). VoIP teknolojisini cazip kılan unsur ise telefon görüşmesine yakın ses kalitesi ile birlikte ucuz maliyette görüşme yapılabilmesidir. Bu nedenle büyük firmalar yada kuruluşlar tarafından kendi içlerindeki haberleşmeyi hem hızlandırmak hem de ucuza mal etmek amacı ile kullanılmaktadır.

Günümüzde yaygın olarak kullanılan PSTN ağları, kullanıcılara her çağrı için bir uçtan bir uca bi devre bağlantısı sağlarlar. Arayan ve aranan tarafların numarasına göre, arayan tarafın bağlı olduğu santralden başlayarak, aradaki santraller ve diğer uçtaki santrale kadar bir devre kurulmaktadır. Bu santraller arasındaki sinyalleşme temel olarak çağrı kurma, çağrı

yönlendirme ve çağrı sonlandırma işlemlerinden oluşmaktadır. Ancak buna paralel olarak veri trafiği için ayrı ağlar oluşmuştur. Doğal olarak ayrı ses ve veri ağları servis sağlayıcı için ilave yük aboneler için de ilave ücret anlamına gelmektedir. PSTN trafiği her geçen gün daha fazla veri içerikli olmaya yüz tuttukça ses ve veri ağlarının birleşmesi yani tek bir platforma indirgenmesi ihtiyacı daha fazla belirgin hale gelmiştir. Bu nedenle internet servis sağlayıcıları ve ekipman üreticileri IP temelli olarak ses/veri iletimine yönelmişlerdir. Böylece VoIP çözümleri daha esnek ve popüler olmuştur.

İlk olarak, 90' lı yıllarda İnternet kullanıcılarının, özel olarak donatılmış bilgisayarlar ya da normal telefon ile bu PC' ler arasında ücretsiz telefon görüşmesini sağlayan bir uygulama şeklinde olan ilkel VoIP uygulamasını, düşük kalitesine rağmen kullanıcılar, uzak mesafe ve uluslararası görüşmelerde tasarruf sağlamanın en güzel yolu olarak değerlendirdiler. Zamanla, yönetilen IP ağları üzerindeki ses kalitesi, telefon ağı üzerindeki ses kalitesi ile eşit hale geldi. Şu anda, IP ağları, telefon ağlarına göre daha verimli bant genişliği sağlıyor. Yeni ses kodlamaları ile, kabul edilebilir ses kalitesi sağlamak için sadece 8 kb/sn kullanılıyor. Telefon ağlarında aynı kaliteyi sağlamak için ise 64 kb/sn gerekiyor. 1995' lerde modemlerin 14,4 kb/sn hızına erişmesi, aynı anda 8 kb/sn' lik kodeklerin (orijinalinde GSM için geliştirilmiştir) kullanılabilir hale gelmesiyle IP ağları üzerinden ses transferi teknik olarak mümkün hale geldi. 1995 te ilk küçük VoIP uygulaması ortaya çıktı. Uygulamalar yaygın olarak kullanılmamakla birlikte standartlaştırma çalışmaları da aynı yıl başlamıştı. 1996' da ilk VoIP standartları kabul edildi. Düşük kapasiteli H.323 geçitleri(gateway) gibi ilk öncü ürünler aynı yıl geliştirildi. Geçitlerin ortaya çıkması ve kullanımı VoIP tarihinde anahtar bir rol oynamıştır.

Geçitler bilindiği üzere iki farklı tip ağ arasına iletişimi sağlamak için kullanılırlar. Kendi aralarında birbirlerinden oldukça farklı protokollerle konuşan iki ağı birleştirmek, konuşturmak ve birinden diğerine veri akışını sağlamak gibi zor bir görev üstlenmişlerdir. İnternet ve Intranet' lerin gelişerek yaygınlaşmasıyla birlikte ses iletişiminin paketlenerek, analog teknolojilere göre daha avantajlı olan IP ağları üzerinden iletimi günümüzde son derece ekonomik ve cazip görünmektedir.

Genel olarak VoIP teknolojisinin avantajları aşağıdaki gibi sıralanabilir.

- İletişim yatırım maliyetlerinin çok büyük oranda düşmesi

- Bakım onarım giderlerinin düşmesi
- Mobilitayı arttırması ve dünyanın her yerinden erişilebilirlik
- Kurulum maliyetlerinin düşmesi
- Operasyon maliyetlerinin düşmesi
- Gelişme ve yeniliklere açıklık
- Varolan İnternet altyapısı üzerine kurulabilmesi
- Ses ve veri hizmetlerinin bir arada verilebilmesi
- Paket anahtarlama birleşik ağ hizmetlerinin devre anahtarlama ISDN' den çok daha verimli olarak sağlanması

2.3 VoIP Senaryoları

VoIP için genel olarak üç tipik senaryo söz konusudur.

Telefondan-Telefona:

Herhangi bir telefon abonesinin diğer bir telefon abonesini internet üzerinden araması anlamını taşır. Bu tip bağlantı uzun mesafe aramalarında ve milletler arası görüşmelerde oldukça uygundur. Burada arayan PSTN abonesi, PSTN ağı ile internet ağı arasında bulunan geçite yönlendirilmektedir. Geçit tarafından ses ve sinyaller IP protokolüne dönüştürülmekte ve aramanın gideceği PSTN ağı ile internet ağı arasında yer alan diğer bir geçitin IP numarası hedef IP olarak IP paketlerine eklenir. Bu noktadan sonra tekrar aranan taraftaki geçit tarafından, IP paketleri ses ve ses sinyallerine dönüştürülmekte ve arayan taraf uygun numaralandırma sistemi ile hedef telefona ulaştırılmaktadır.

Telefondan-Bilgisayara, Bilgisayardan-Telefona:

Telefondan bilgisayara, bilgisayardan telefona senaryoları genellikle internet servis sağlayıcılarının ilgi alanına girmektedir. Bilgisayardan telefona senaryosu genelde "surf and talk" ya da "click to dial" olarak anılır. Bir son kullanıcının web sayfalarını gezerken bir yandan da internet servis sağlayıcısı ile görüşmesi ve sorularına cevap almasına olanak sağlar. Web' de gezerken telefondan ulaşılabilir olmak bir son kullanıcı için oldukça önemli bir noktadır.

Her iki senaryo da uygulamada bir PSTN telefonunun internet üzerinde bir ses bağlantısı kurması söz konusu. Bu duruma göre aranacak bilgisayar için uygun bir PSTN numarası verilmesi gerekmektedir. Bilgisayar, üzerindeki uygulama programları sayesinde bir nevi IP

telefonu gibi çalışmaktadır. Aynı şekilde bunun tersi; yani bir bilgisayardan bir PSTN telefonunu aramak mümkündür. Yine aynı mantıkla aranan bilgisayardan uygulama programları ile PSTN ağlarının numarasının çevrilmesi gerekmektedir. Bu senaryoda iki farklı tür ağ arasında bağlantı kurulabilmesi bir geçit kullanımını gerektirir. Bu geçit kullanılarak aramalar IP ağdan devre anahtarlama ağına ya da devre anahtarlama ağından IP ağına taşınabilir.

Bilgisayardan-Bilgisayara:

En temel ve gerçekleştirimi en kolay VoIP senaryosudur. Bu senaryoda VoIP bir IP ağı üzerinden birbirine bağlanmış IP terminalleri arasında gerçekleştirilir. Her iki terminal de bir IP adresine sahip olmalı ve kaynak terminal hedef terminalin IP adresini bilmelidir.

2.4 VoIP Çalışma Şekli

VoIP’ te analog işaret, ADC(analog to digital converter – analog’ tan sayısal’ a dönüştürücü) yardımıyla sayısal işarete çevirilir. Bu sayısal işaretlerin hedef kullanıcıya iletilmeden önce uygun bir formatla kodlanmış, sıkıştırılmış(encode) olması gerekmektedir. Ses paketlerini taşıma işlemi, gerçek zamanlı protokoller ile sağlanır. Hedef kullanıcıya ulaşan paketlerin öncelikle çözülmesi(decode) gerekir. Paketler çözüldükten sonra, DAC(digital to analog converter – sayısal’ dan analog’ a dönüştürücü) yardımıyla sayısal ses verisi, analog ses işaretine çevrilir. Son olarak, bu analog ses işareti ses kartına veya telefona yollanarak işlem tamamlanır.

2.5 VoIP Sistemlerde Kullanılan Kodekler

Kodekler analog ses işaretlerini sayısal işaretlere çevirmektedir. Kodekler genelde ses giriş işareti üzerinde öngörülen dizilerin karmaşık işlemler ile analizini ve uygun bir bant genişliğinden iletilebilmesi için sıkıştırılmasını da sağlamaktadır (Ibe, 2002).

Bu kodeklerden en basiti olan PCM (Darbe Kod Modülasyonu - Pulse Code Modulation) çok önceden beridir mevcut telefon ağlarında yaygın olarak kullanılmaktadır. PCM 64 kb/sn bant genişliğinde iki temel varyasyona sahiptir. A-law ve μ -law olarak adlandırılan bu varyasyonlar birbirlerine çok benzer logaritmik bir sıkıştırma ile 8 bit kullanarak 12, 13 bitlik bir ses kalitesi sağlamaktadır. μ -law Amerika ve Kanada tarafından kullanılmakta iken A-law Avrupa tarafından tercih edilmektedir. PCM kodeklerindeki A-law μ -law uyumsuzluğu rahatsız edici olmakla birlikte ses anlaşılabilir niteliktedir.

Sık kullanılan diğ er bir kodek ise ADPCM (Değ işken Farksal Darbe Kod Modulasyonu - Adaptive Differential Pulse Code Modulation) kodlayıcısıdır. En  ok kullanılan y ntem ise her  rnek i in 4 bit' in kullanılmasıdır. PCM' den farklı olarak bu 4 bit, ardışık  rnekler arasındaki farkları dođrusal bir tahmin kullanarak kodlamaktadır. B ylece 32 kb/sn band geniřliđi yeterli olmaktadır (Vaudreuil vd., 2004).

PCM ve ADPCM kodekler dalga řeklini bulmaya y nelik kodlayıcılardır (Vaudreuil vd., 2004). Yeni sıkıştırma teknikleri son 20 senedir ses oluřturma  zerine yapılan  alıřmalardan elde edilen bilgiler sayesinde meydana getirilmiřtir. Bu teknikler iřaret iřleme y ntemlerini kullanarak sesi sıkıřtırmakta ve sadece ses telleri ile ses yoluna bađlı parametreleri kullanmakta, b ylece daha az bant geniřliđi kullanılmaktadır. Bu t r kodekler PCM ve ADPCM' de olduđu gibi sesin dalga řeklini deđil ses kaynađını modelleme y ntemini kullandıkları i in kaynak kodlayıcılar olarak adlandırılırlar. Bu kodlayıcıların LPC (Dođrusal Tahmini ng r m Kod - Linear Predictive Code), CELP (Kod Uyarmalı Dođrusal Tahmin ng r m - Code Excited Linear Prediction) ve MP-MLQ ( oklu Darbe  oklu Seviye Nicemlemesi - Multi Pulse Multilevel Quantization) gibi varyasyonları vardır.

Bu varyasyonların deđiřik tanımlarının olduđu alt kategorileri de bulunmaktadır.  rneđin CELP' in d ř k gecikmeli varyasyonu LD-CELP (D ř k Gecikmeli CELP - Low Delay CELP), daha karmařık ses yolu modellemesini yapan varyasyonu ise CS-ACELP (Konjuke Yapı Cebirsel CELP - Conjugate Structure Algebraic CELP) gibi isimler ile anılmaktadır.

Paket anahtarlamalı ađlarda en  ok kullanılan ses kodlayıcı standartları ITU tarafından tanımlanan ařađıdaki kodlayıcılardır:

- G.711, yukarıda anlatılan PCM kodlayıcısının standartıdır. G.711 ile kodlanmış ses, geleneksel telefon ađlarındaki santrallerin sayısal ses iletimi i in kendi i lerinde veya birbirleri arasında kullandığı y ntemdir.
- G.726, ADPCM kodlayıcısını tanımlamaktadır. G.726 standartında 40, 32, 24, 16 kb/sn ADPCM kodlamaları tanımlanmıştır (Vaudreuil vd., 2004). ADPCM de geleneksel telefon ađlarındaki sayısal santrallerin  ođu tarafından anlaşılabilmektedir.
- G.728, LD-CELP' i tanımlamaktadır. Geleneksel telefon ađları ile haberleřmede G.728 ile kodlanmış sesin G.711 formatına d n řt r lmesi gerekmektedir.

- G.729, CS-ACELP' i tanımlamakta ve ses 8 kb/sn bant genişliğinde ADPCM kalitesinde kodlanabilmektedir (Vaudreuil vd., 2004). G.729 kodlayıcısının daha az karmaşık olan versiyonu G.729 Annex A (G.729a) ile tanımlanmıştır. Konuşma esnasında dinlemede olan insanın sessizliği sırasında daha az bant genişliği harcayabilmesi için ise G.729 Annex B tanımlanmıştır. G. 729B' de VAD (Konuşmanın Başladığının Tespiti - Voice Activation Detection) yöntemi kullanılmaktadır.
- G.723.1 ise sesin çok düşük bit oranlarında sıkıştırılabilmesi için MP-MLQ ve ACELP yöntemlerini tanımlamaktadır. MP-MLQ ile 6.3 kb/sn, ACELP ile de 5.3 kb/sn band genişliğinde sıkıştırma yapılmaktadır.

2.5.1 Kodek Değerlendirme Parametreleri

Paket anahtarlama ağılarda ses iletiminde kodek tecih çeşitli parametrelere bağlıdır. Paket anahtarlama ağılar üzerinden ses iletiminde tercihler hat kapasitesine (ya da maliyetine), iki uç arasındaki mesafeye ve mevcut işlem gücüne göre değişebilmektedir. Kodekler genel olarak aşağıdaki parametrelere göre değerlendirilmektedir.

- Kalite ve kapasite
- Kodlama gecikmesi
- Karmaşıklığı ve maliyeti
- Hata duyarlılığı
- Anahtarlama veya kod değiştirme

2.5.1.1 Kalite ve Kapasite

Genel olarak ses kalitesi ve sıkıştırma kapasitesi birbirleriyle ters orantılıdır. Kapasite nesnel olarak ölçülebilirken (ihtiyaç duyulan band genişliği) kalite için nesnel bir ölçüm çok zordur. Ses kalitesini ölçmek için en çok kullanılan yöntem MOS (Ortalama Fikir Skoru - Mean Opinion Score) ölçümüdür. MOS' da birçok insan sesin kalitesine yönelik kendi öznel değerlendirmelerini 1 ile 5 arasında yapmaktadır. Bu öznel değerlendirmelerin ortalaması ise MOS skorunu vermektedir. Ses kalitesi için diğer öznel yöntemlerden DMOS (Görece MOS - Differential MOS) bir başka kodeğe göre kaliteyi değerlendirmektedir.

Günümüzde telefon ağı oldukça yaygındır ve insanlar bu kaliteye alışmışlardır. G.711 kodeğinin kalitesi genelde telefon ağının kalitesi olarak baz alınır. Çünkü mevcut telefon

ağlarında da ses sayısallaştırılırken genel olarak PCM kullanır. G.711' in kalitesi 16 bitlik bir sesin kalitesine göre 5 üzerinden 4.1 MOS değerindedir. Buna göre diğer konuşma kodeklerinin kapasite ve kaliteleri Çizelge 2.1' de görüldüğü gibidir.

Çizelge 2.1 Konuşma kodeklerinin kapasiteleri ve kaliteleri

Kodek	Bit Hızı	MOS skoru
PCM. G.711	64 kb/sn	4.1
ADPCM. G.726	32 kb/sn	3.85
LD-CELP. G.728	16 kb/sn	3.61
CS-ACELP. G.729	8 kb/sn	3.92
CS-ACELP. G.729.a	8 kb/sn	3.7
MP-MLQ.G.723.1	6.3 kb/sn	3.9
G.723.1 ACELP	5.3 kb/sn	3.65

2.5.1.2 Kodlama Gecikmesi

Kodlama gecikmesi de en az ses kalitesi kadar önemli bir parametredir. Karşılıklı bir konuşmada sesin karşı tarafa ulaşmasındaki gecikmenin yüksek olması rahatsızlık vermektedir. ITU G.114' deki gecikmenin 150 ms' nin altında olmasını normal ve 200 ile 450 ms arasında olması rahatsızlık verici fakat kabul edilebilir (özellikle uluslararası telefon görüşmelerinde 200 ms aşılabilmektedir) olarak belirtmiştir.

Gecikmeler genel olarak üçe ayrılabilir. Bunlar algoritmik gecikme, işlemsel gecikme ve iletişim gecikmesidir. Algoritmik gecikme ve işlemsel gecikme kodeklerin çalışma mantığıyla ilgilidir. Algoritmik gecikme kodeğin işlevini yerine getirebilmesi için tampon belleğe yüklemesi gereken örnek sayısına bağlıdır. VoIP kodeklerinin kodlama gecikmeleri Çizelge 2.2' de verilmektedir.

Çizelge 2.2 Konuşma kodeklerinde algoritmik gecikmeler

Kodek	Kodlama Gecikmesi
PCM. G.711	0.125 ms
ADPCM. G.726	0.125 ms
LD-CELP. G.728	0.625 ms

CS-ACELP. G.729	15 ms
MP-MLQ.G.723.1	37.5 ms
G.723.1 ACELP	37.5 ms

Çizelge 2.2’ de görüldüğü gibi en az gecikme PCM ve ADPCM’ de olmaktadır. Bu da 1/8000 sn değerinde ve ardışık iki örnek arasındaki süredir. G.728 algoritmasında ise herbir ses çerçevesinde 5 örneklik bir tampon kullanılmaktadır. G.729 çerçevesi 10 ms, G.723 çerçeveleri ise 30 ms’ den oluşmaktadır. Fakat bu kodekler sıkıştırma yapmak için de sırasıyla 5 ve 7.5 ms ileriye bakmaktadırlar. Böylece toplam gecikme G.729 için 15 ms G.723.1 için ise 37.5 ms olmaktadır.

PCM ve ADPCM’ in algoritmik gecikmeleri oldukça düşük olsa dahi her örneği paketleyip göndermek paketleme başlıklarının getireceği ek yük nedeniyle bağlantı verimliliği çok düşük olacaktır (Vaudreuil vd., 2004). Bu nedenle bu kodekler kullanılsa dahi genelde en az 20 ms’ lik kısmı paketlenir.

2.5.1.3 Karmaşıklık ve Maliyeti

İşlemsel gecikme kodeğin yaptığı işlemlerin karmaşıklığı ve kodeğin çalıştığı işlemcinin gücü ile bağıntılıdır. Saniyede yapılan işlem MIPS (Milyon İşlem Bölü Saniye – Mega Instruction Per Second) ile ölçülmektedir. Kodek ne kadar karmaşık ise o kadar büyük MIPS değerlerine sahiptir. Dolayısıyla kodeği çalıştıracak donanım o kadar pahalıdır. Buna göre konuşma kodeklerinin MIPS değerleri Çizelge 2.3’ de verilmektedir.

Çizelge 2.3 Konuşma kodeklerinde işlem karmaşıklığı

Kodek	Kodlama Karmaşıklığı
PCM. G.711	0.34 MIPS
ADPCM. G.726	14 MIPS
LD-CELP. G.728	33 MIPS
CS-ACELP. G.729	20 MIPS
CS-ACELP. G.729.a	10.5 MIPS
MP-MLQ.G.723.1	16 MIPS
G.723.1 ACELP	16 MIPS

2.5.1.4 Hata Duyarlılığı

Kodeklerin hataya duyarsız olmaları tercih edilmektedir. G.711 en az hata duyarlılığına sahiptir. G.711' in herhangi bir örneğindeki hata sadece 0.125 ms' lik bir hataya yol açacaktır. G.723.1' in çerçeve büyüklüğü 30 ms' lik sese tekabül ettiği için en fazla hata duyarlılığına sahiptir. G.726 farksal kodlama yaptığı için hata duyarlılığı G.723.1, G.728 ve G.729' dan daha azdır. G.729 ve benzeri kodekler hata duyarlılığını azaltmak için bad frame indication (bfi) bitlerinden faydalanmaktadır.

2.6 VoIP Servis Kalitesi(QoS)

Paket anahtarlamalı ağlar veri iletimi için tasarlandığından sesi PSTN kalitesinde iletmesi çok zor ve karmaşıktır (Ibe, 2002). Bu nedenle birçok standartlar geliştirmiştir. ITU E.800 servis kalitesinin tanımını vermektedir. Bu tanıma göre servis kalitesi; servis kullanıcısının tatminlik derecesini belirleyen servis performansının toplam etkisi olarak ifade edilmiştir.

Servis kalitesi bir performans kriteridir. Servis kalitesini etkileyen her etken, sistem performansını doğrudan etkileyecek ve hatta belirli bir QoS değerinin tutturulabilmesi için de sisteme ek maliyet getirecek yatırımlar yapmak gerekecektir. Bu yüzden servis kalitesini etkileyen unsurları iyi analiz edebilmek büyük önem taşımaktadır.

2.6.1 Gecikme(Delay)

Ses paketinin kaynaktan yollanması ile alıcı tarafından alınması arasında geçen süre olarak tanımlanır. Başlıca gecikmeler beş adettir.

Örnekleme Gecikmesi: Darbe kodlama modülasyonu yapılmadan önce ses işareti örneklenir. İşte bu örnekleme esnasında belirli bir süre geçer. Bu geçen süreye örnekleme gecikmesi denir. Bu değer işaretin örnekleme aralığı ile doğru orantılıdır. Çoğu zaman 125-150 ms kadardır.

Sıkıştırma Gecikmesi: Sesin belirli bir kodek ile kodlanması sırasında geçen süredir. Bu süre kodlama algoritmasının karmaşıklığına ve kodlama yapan işlemcinin işlem yapma hızına bağlıdır.

Kod Çözme Gecikmesi: Belirli bir kodek ile sıkıştırılmış sesin açılması (decode) sırasında geçen süredir. Bu süre de kod çözücü işlemcinin hızına ve sıkıştırma algoritmasının karmaşıklığına bağlı olarak değişir.

Transmisyon Gecikmesi: Paketin iletimi esnasında iletim yollarında geçen süredir. Bu süreyi azaltmanın en iyi yolu, iletim hattının enformasyon hızını arttırmaktır. Örneğin, 16 kb'lık bir veri paketi 4 kb/s hızındaki bir hattan iletildiğinde iletim 4 sn alırken, 48 kb/s'lik bir hattan (ISDN-D kanalı) yollandığında iletim 0,33 sn alır.

Bellekleme Gecikmesi: Tüm paket gecikmelerinin aynı yapılması amacıyla verilerin belirli bir süre belleklerde(buffer) tutulmasından kaynaklanan gecikmedir. Ayrıca bu gecikme sadece karasal transmisyonda görülmez. Bunun yanı sıra uydu haberleşmesinde de uydunun Dünya ile olan konumunun devamlı değiştiği orta (MEO) ve kısa yörünge (LEO) uydularının veri aktarımı esnasında da görülür.

Jitter: Bir kaynaktan çıkan paketlerin her birinin farklı gecikme değerleriyle alıcıya ulaşması sonucu oluşur. Buna gecikme varyasyonu da denir.

2.6.2 Yankı(Echo)

İletim hatlarındaki ek noktalarında, hattın uygun empedans ile sonlandırılmaması nedeniyle oluşur. Empedans uyumsuzluğu olan noktalarda işaretlerin bir bölümü hatta yoluna devam ederken elektriksel işaretin diğer bir bölümü ise yansır. Benzer etki PSTN' deki iletim hatlarında kullanılan 2 tel – 4 tel dönüşümü yapan hibrit trafolar da ortaya çıkar.

2.6.3 Net Çıkış Hızı(Throughput)

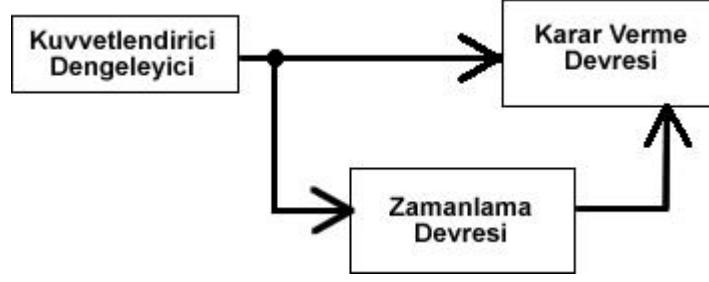
Alıcı tarafından alınan verilerin ortalama hızıdır. Bu ortalama hesaplanırken tıkanma durumu da göz önünde tutulmalıdır.

2.6.4 Paket Kayıp Oranı

Hedefe ulaşmayan veri paketlerinin o hedefe yollanan veri paketlerine oranıdır.

2.6.5 Gürültü

Hattın enformasyon hızını dolayısıyla servis kalitesini etkileyen bir başka unsur da gürültüdür. Gürültü arttıkça kanal kapasitesi düşer; dolayısıyla iletim süresi artar ve servis kalitesi olumsuz etkilenir. Gürültünün azaltılması için c sistemlerinde ayırıcı (splitter) denen bir cihaz kullanılırken, radyo link sistemlerinde eliptik dalga kılavuzları, GSM gibi hücresel kablosuz mobil sistemlerde verici işaret gücünün artırılması gibi yöntemler kullanılır. Ayrıca gürültü bazen o kadar kritik bir noktaya gelir ki; sistemdeki bit – hata oranı (BER) değerlerinin normal sınırlara çekilmesi için ek maliyetli rejeneratif repetörler (repeater, access point) kullanılır. Şekil 2.3' te rejeneratif repetörün blok diyagramı verilmiştir.

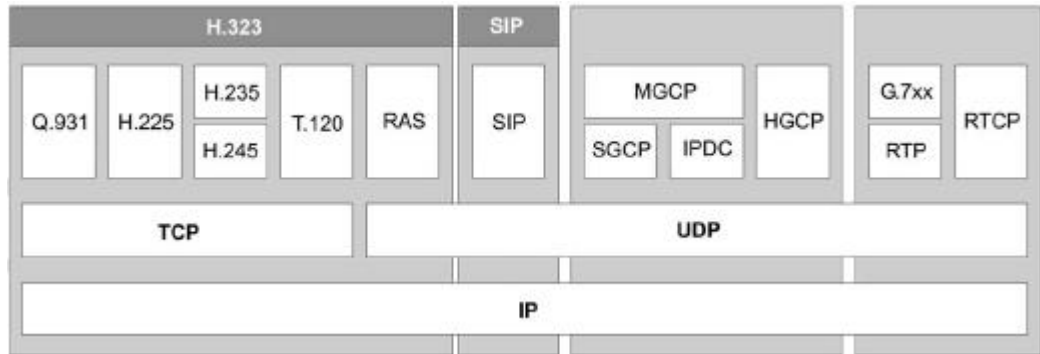


Şekil 2.3 Regeneratif repetörün blok diyagramı

İkinci bir gürültü kaynağı ise; sesin kaydedildiği ortamın gürültüsü ya da arka fon gürültüsüdür. Bu gürültünün engellenmesi, hat gürültüsüne nispeten daha kolaydır. Mikrofon tarafından alınan ses işaretinin genliği, özel bir eşik sezici aracılığıyla belirli bir genlik değeriyle karşılaştırılır ve bu genliğin altında kalan değerler karşı tarafa iletilmezler, çünkü bu bileşenler çoğunlukla gürültüden kaynaklanan işaretlerdir. Buna ek olarak, verici tarafta konuşma olmadığı zaman karşı tarafa sessizlik olduğunun anlatılması için özel bir işaret yollar. Böylece kısıtlı band genişliği daha verimli kullanılmış olur ve hatta gereksiz veri paketleri yollanmaz.

2.7 VoIP Protokolleri

VoIP' in temel problemleri güvenilirlik, ses kalitesi ve IP ile devre anahtarlamalı ağlar arasında kullanılan farklı standartların birbirleriyle uyumudur. Bu problemleri çözmek için birçok protokol standardı geliştirilmiştir. Bu protokollerden bir kısmı H.323, MGCP (Medya Geçidi Kontrol Protokolü - Media Gateway Control Protocol), SIP (Oturum Başlatma Protokolü - Session Initiation Protocol) gibi kontrol ve sinyalleşme ile ilgili standartları tanımlarlarken bir kısmı da RTP (Gerçek Zamanlı Taşıma Protokolü - Realtime Transport Protokol), RSVP gibi sesin kaliteli iletimini sağlamaktadır.



Şekil 2.4 Çoklu ortam ağ protokolleri

2.7.1 H.323

H.323, internet gibi paket-tabanlı ağlar üzerinden görüntü ve bilgi konferansını yapmak için gerekli protokolleri içerir. H.323 hizmet kalitesini dikkate almaksızın paket-tabanlı ağlarda gerçek zamanlı noktadan-noktaya (end-to-end) ses gönderimi protokollerinin özelliklerini belirler (Muller vd., 2001). H.323 içinde bulunduğu ağın iletim tabakasının (transport layer) hemen üstünde yer alır. Örnek olarak, H.323 ethernet, TCP/UDP/IP, ATM ve Frame Relay gibi paket-tabanlı ağların hepsinin üzerinde kullanılabilir. H.323 ağlar arası konferans için IP'yi kullanmaktadır. H.323 ITU' nun hazırladığı video konferans önerilerinden biridir. Fakat H.323' ün diğerlerinden farkı bu standardın H.323 kullanıcılarının diğer video konferans ağ tiplerindeki kullanıcılarla haberleşebilmesini sağlamasıdır. H.310 (B-ISDN için), H.320 (dar bantlı ISDN için), H.321 (ATM için), H.324 (PSTN için) H.323' ün haberleşebildiği ağ tipleridir (Muller vd., 2001). Bir H.323 sistemi oluşturulmasında 4 mantıksal yapı kullanılmaktadır. Bunlar terminaller, geçitler (gateway), gatekeeper' lar ve MCU' lardır. Terminal, gecit ve MCU' lar uç-noktalar (end points) olarak adlandırılırlar.

2.7.2 MGCP

H.323 tam anlamı ile bir çoklu ortam görüşme protokolü iken MGCP ise basit bir aygıt kontrol protokolüdür. MGCP, SIP veya H.323 yerine geliştirilmiş bir protokol olmayıp, ana işlevi ortam geçitleri (analog ses ve paket dönüşümlerini sağlayan ağ cihazları) arasında bağlantıları sağlayarak, farklı ağlar arasında ses iletimini gerçekleştirmektir. SIP ve H.323 çağrı denetimini sağlarken, MGCP ağlar arasında bağlantıyı sağlar. MGCP, bir ağ cihazının (Çağrılar kurmakla sorumlu), asıl İnternet-IP ses akışını gerçekleştiren ortam, ses iletişim cihazlarını denetlemesini sağlayan basit bir denetim protokolüdür. MGCP arama araçları ve ortam geçit denetçileri olarak bilenen yazılımların, harici olarak çoklu servis paket ağlarının uçlarında yerleşik bulunan ortam ses iletim-akış (streaming) cihazlarını veya ortam geçitlerini denetlemelerine olanak sağlar (Andreasen ve Foster, 2003). Ses akış işlemlerinin ve çağrı denetim işlemlerinin bu şekilde ayrılması ortam geçidinin işletimini basitleştirmektedir. Ortam geçitleri, VoIP geçitleri, VoATM geçitleri, modem bankaları, kablo modemler ve santral yazılımları (Soft PBX) olabilir. MGCP, H.323 ve SS7 geçitleri arasında kullanılabilir. H.323 geçitleri ortam geçişini sağlarken, SS7 geçitleri çağrı denetim bilgisinin çevrimini sağlar. MGCP, MEGACO (H.248) gibi tanınmış bir protokol değildir. MGCP, oturum kurulmadan önce internet uç noktaları arasında port tahsisi ve kodek seçimi için SDP' yi (Oturum Tanım Protokolü – Session Description Protocol) kullanır (Andreasen ve Foster, 2003).

2.7.3 SIP

H.323' ten sonra noktadan noktaya mimarilerde SIP de yerini almıştır. H.323' ten farklı olarak SIP, felsefe ve amacı bakımından internet tipi bir protokoldür. SIP, IETF MMUSIC Çalışma Grubu tarafından, Eylül 1999' da RFC 2543' te açıklanmıştır. SIP H.323 için alternatif ve MGCP gibi kullanıcı/sunucu protokolleri için ise tamamlayıcı olarak düşünülmüştür.

SIP' e, sunucu ile çok az iletişim gerektiren veya etkileşim gerektirmeyen akıllı uç noktalarda ihtiyaç duyulur. SIP bir ya da birden fazla katılımcının yer aldığı oturumları kurmak, değiştirmek ve sonlandırmak için tasarlanmış bir kontrol protokolüdür. Her uç nokta kendi işaretlenmesini yönetir, bu yönetim kullanıcı ve diğer uç nokta için de yapılır. En önemli nokta; MGCP aygıt kontrolü sağlarken SIP' in oturum kontrolü sunmasıdır. Bu özellik SIP' e birçok üstünlük sağlar. Öncelikle, basit mesaj yapısı; H.323' e göre arama ayarlarının daha az aşamada gerçekleşmesini sağlar bu özellik donanım olarak benzer işlemi kullanan H.323' e göre performansı yükseltir. SIP, H.323' e göre daha ölçeklenebilir bir yapıdır. Çünkü, SIP yapısal olarak sınıflandırılmış ve durumsuz (stateless) bir arama modelidir. Asıl farklılık ve avantaj sağlayan SIP' in İnternet modeli bir protokol olmasıdır. ASN.1 mesajlaşma yerine basit olan, HTTP/1.1 de yapılandırılmış ASCII kullanır. Böylece, SIP mesajlaşmada şifre çözümünü ve hata denetimi kolaylaşmakta, ve küçük ayarlamalar ile web tipi uygulamalar SIP servisini destekleyebilmektedir. SIP, URL(Evrensel Kaynak Konumlandırıcı – Universal Resource Locator), E.164 (North American Numbering Plan Addressing) standartlarını tam desteklemektedir. Böylece, SIP modelinde kullanıcının e-posta adresi ile telefon adresi aynı olabilir ve karmaşık oturumlarda değişik uç noktalar kendi aralarında iletişim kurabilirler.

SIP ile alakalı daha detaylı bilgi Bölüm 3' te verilmiştir.

2.7.4 RSVP

Birçok veri trafiği bir düzeye kadar gürültülüye ve gecikmelere tolerans gösterebilir. Buna karşılık, görüntü ve ses ile ilgili trafiğin, sürekli ve gürültüsüz olması gerekmektedir. Bu sebepten dolayı ses, video ve radyo sinyal trafiğinin ağ üzerinde, kalitesi yüksek servis gerektirdiğinden (QoS) problemlerin çıkmasına neden olmaktadır. RSVP, IP ağ üzerinde gecikmeye duyarlı olan çoklu medya trafiğinin taşınmasını sağlar. RSVP' de ağ bantgenişliği önceden rezerve edilmektedir. Bu da çoklu yayın ve tekli yayın ağlarda kaliteli servis

garantisi verir. Tekli yayın servislerde tek bir alıcı olmasına karşılık çoklu yayın servislerde daha çok alıcı olabilir.

Eğer band genişliği rezervasyonu için yeteri kadar kapasite yok ise iletişime izin verilmez. Verimli olmayacak iletişimlere izin verilmediği için band genişliğinden tasarruf da yapılmış olur. RSVP, TCP/IP tabanlı ağlarda uçtan-uca haberleşme sağlayan bir protokoldür.

RSVP bir yönlendirme protokolü değildir. IP ağları üzerinde bir kontrol protokolüdür (Braden, 1997). OSI modeline göre taşıma katmanı seviyesindedir. RSVP yönlendirme protokolleri ile birlikte çalışır. Bu anlamda RSVP, ICMP ve IGMP (Internet Grup Yönetimi Protokolü - Internet Group Management Protocol) ile benzerliklere sahiptir.

RSVP ile aynı kaynağa ve servis kalitesine ait mesajlar sıralı bir veri akışı ile taşınırlar (Braden, 1997). Bu veri akışı da oturum olarak bilinir. Bir RSVP işleminde göndericiler, yönlendiriciler ve alıcılar bulunmaktadır.

2.7.5 RTP

Paket anahtarmalı ağlarda sinyalleşme protokolleri sadece birimlerin sinyalleşmesini sağlarlar, veriyi taşımazlar. Verinin gerçek zamanlı olarak, hedef birime taşınması RTP protokolü ile sağlanır.

RTP ile alakalı daha detaylı bilgi Bölüm 4' te verilmiştir.

3. OTURUM BAŞLATMA PROTOKOLÜ (SIP)

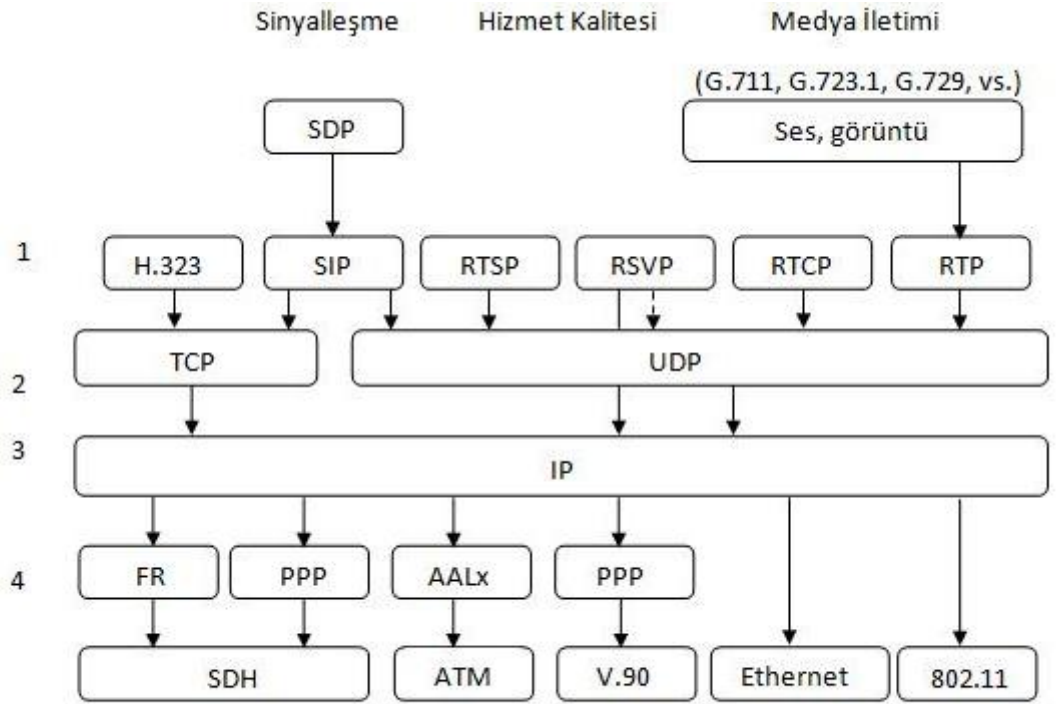
SIP (Oturum Başlatma Protokolü), uygulama katmanında çalışan, iki veya daha fazla kullanıcı arasında bağlantının kurulup oturumun başlatılmasından, parametrelerinin düzenlenmesinden ve de sonlandırılmasından sorumlu olan bir sinyalleşme protokolüdür. SIP protokolü, 1990' lı yılların ortalarında, üniversitelerde düzenlenen büyük ölçekli çoğul ortam konferanslarına, diğer üniversitelerden gelen katılımcıların da katılabilmesine olanak tanımak için Columbia Üniversitesi' nde yapılan araştırmalardan doğmuştur. IETF MMUSIC tarafından tasarlanmış olan SIP protokolü, ilk olarak 1996 yılında RFC 2543 standardı olarak yayınlanmış, daha sonra geliştirilerek 2002 yılında RFC 3261 standardı ile son halini almıştır (Rosenberg vd., 2002).

Bir bilgisayar ağı üzerinden sesli ve görüntülü iletişim sağlayan bir dizi protokolü tanımlayan, ITU-T (Uluslararası Telekomünikasyon Birliği) tarafından tanımlanan H.323 protokolünün, IP uygulamaları için yeteri kadar kullanışlı ve esnek olmaması, SIP protokolünün geliştirilmesine sebep olmuştur. SIP protokolünün basit mesaj yapısı, H.323' e göre arama ayarlarının daha az aşamada gerçekleşmesini sağlar. Bu özellik donanım olarak benzer işlemi kullanan H.323' e göre performansı yükseltir. SIP protokolü, H.323' e göre daha ölçeklenebilir bir yapıdır. Çünkü, SIP yapısal olarak sınıflandırılmış bir arama modelidir. Asıl farklılık ve avantaj sağlayan SIP' in internet modeli bir protokol olmasıdır. SIP protokolü, HTTP ve SMTP gibi metin tabanlı protokoller temel alınarak oluşturulmuş bir protokoldür (Rosenberg vd., 2002). Bu metin tabanlı mimari yaklaşım, SIP protokolüne, ihtiyaçlara göre kural eklenip, değiştirilebilen esnek bir yapı kazandırmıştır. ASN.1 mesajlaşma yerine basit olan, HTTP/1.1 de yapılandırılmış ASCII kullanır. Böylece, SIP sinyalleşmelerinde, şifre çözümü ve hata denetimi kolaylaşmakta, ve küçük ayarlamalar ile web tipi uygulamalar SIP servisini destekleyebilmektedir. SIP, URL(Evrensel Kaynak Konumlandırıcı), E.164 (North American Numbering Plan Addressing) standartlarını tam desteklemektedir. Böylece, SIP modelinde kullanıcının e-posta adresi ile telefon adresi aynı olabilir ve karmaşık oturumlarda değişik uç noktalar kendi aralarında iletişim kurabilirler. Bu özellikleri sayesinde SIP protokolü, internet telefonculuğunda kullanılan en yaygın protokol haline getirmiştir. Bu yüzden, daha eski VoIP sistemleri H.323' ü kullanırken, günümüzdeki mevcut VoIP sistemlerinin çoğu SIP standardını izlemektedir.

SIP protokolü tarafından kurulan çoklu ortam oturumları, ses, görüntü ve veri tipinde olabilir. Ses tipindeki oturumlara, internet üzerinden yapılan telefon çağrıları ve konferanslar; görüntü

tipindeki oturumlara, görüntülü telefon çağrıları ve görüntülü konferanslar; veri tipindeki oturumlara anlık mesajlaşma uygulamaları örnek olarak verilebilir. Ayrıca SIP protokolü; ses ile zenginleştirilmiş e-ticaret, Web sayfası tıkla-çevir(Click-to-Call), arkadaş listeleri ile anlık mesajlaşma ve çok daha fazla özellik ile tümleşik ses ve çoğul ortam servislerini desteklemektedir.

SIP protokolünün temel görevi, çoklu ortam aygıtlarının iletişimini sağlamaktır. Diğer bir deyişle, SIP protokolü kullanıcılar arası sinyalleşmeyi sağlar. SIP protokolünün kendisi bir veri taşıma protokolü değildir, dolayısıyla SIP, tümleşik bir haberleşme protokolü değildir. Bu yüzden SIP protokolünün ses, görüntü ve çoklu ortam servislerini verebilmesi için, uygulama katmanındaki diğer protokollerle işbirliği içinde olması gerekir. Bu protokoller Şekil 3.1’ de gösterilmiştir.



1. Uygulama Katmanı 2. Taşıma Katmanı 3. İnternet Katmanı 4. Ağ Arayüz Katmanı

Şekil 3.1 SIP protokol kümesi

SDP(Oturum Tanımlama Protokolü): SIP protokolü, çoklu ortam uygulamalarında, oturumun başlatılmasını, yürütülmesini ve sonlandırılması işlemlerini yerine getiren bir protokol iken; SDP protokolü, oturumun kendisini(oturumun başlangıç parametrelerini) tanımlayan bir protokoldür (Handley vd., 2006). İlk olarak RFC 2327 standardı olarak

yayımlanan SDP protokolü, daha sonra geliştirilerek RFC 4566 standardı olarak son halini almıştır.

RTP(Gerçek-Zamanlı Aktarım Protokolü): İlk olarak RFC 1889 standardı olarak yayımlanan, daha sonra geliştirilerek RFC 3550 standardı olarak son halini alan RTP protokolü, ses, görüntü veya verilerin, gerçek zamanlı olarak uçtan uca taşınmasını sağlayan bir protokoldür (Schulzrinne, 2003b). Kayıp verinin saptanması, güvenlik ve içerik belirlenmesi gibi konulardan sorumludur. RTP, kaynak rezervasyonu (resource reservation) yapmaz ve hizmet kalitesini (QoS) garanti etmez.

RTCP(RTP Kontrol Protokolü): RFC 3550 standardında tanımlanan RTCP protokolü, RTP protokolü ile birlikte çalışır. RTCP' nin temel fonksiyonu, RTP tarafından sağlanan hizmetin kalitesi hakkında geri bildirimde bulunmaktır. Oturumda yer alan tüm kullanıcılara periyodik olarak kontrol paketleri göndererek alma kalitesi (reception quality) ile ilgili bilgi sağlar (Schulzrinne, 2003a; 2003b). Eğer sıkışıklık oluşmaya başlamışsa, uygulama veri hızını azaltabilir.

RTSP(Gerçek-Zamanlı Akış Protokolü): RFC 2326 standardı olarak yayımlanan RTSP protokolü, akış sistemleri için geliştirilmiş bir protokoldür. Bu protokol istemcinin, akış sunucusunu "oynat", "duraklat" gibi komutlarla uzaktan kontrol edebilmesini sağlar ve istemciye sunucudaki dosyalara süresi sınırlı erişim izni verir. Birçok RTSP sunucusu, RTP standartlarını kullanır.

RSVP(Kaynak Rezervasyon Protokolü): RFC 2205 standardı olarak yayımlanan RSVP protokolü, ses, görüntü ve veri kalitesinin ağ boyunca istenilen kalitede sağlanabilmesi için ağ cihazları arasında sinyalleşmeyi sağlar (Braden, 1997).

Özetle, SIP protokolü aşağıdaki işlevleri desteklemektedir:

- **İsim dönüşümü ve kullanıcı konumu:**

Birbirlerinin aygıt adreslerinin veya fiziksel konumlarının ayrıntılarını bilmeden, kişilerin birbirlerini bulmasına olanak sağlar. Diğer bir deyişle, çağrının aranan kullanıcıya, aranan kullanıcı nerede olursa olsun, bulunduğu konumdan bağımsız olarak ulaştırılmasını sağlar. Bu özellik, SIP kullanıcılarına, tek bir numara ile ister evden, ister işten, isterse de cep telefonlarından ulaşılmasını sağlar. Tek bir numara ile konumdan bağımsız haberleşme(kullanıcı gezginliği), SIP protokolünü, internet telefonculuğunda lider kılan en önemli özelliklerden bir tanesidir.

- **Ortam görüşmesi:**

Bir oturumdaki tüm katılımcıların, oturumda desteklenen özellikler üzerinde uzlaşmalarını sağlayarak ortak bir medya oluşturur. Örneğin, bir oturumdaki tüm katılımcıların, sesli görüşme yapabilmeleri için aynı ses kodekleri üzerinde uzlaşmaları gerekmektedir.

- **Oturum katılımcı yönetimi:**

Katılımcıların eklenmesini, çıkartılmasını ya da aktarılmasını yönetir.

- **Oturum özellik değişiklikleri:**

Oturum yürümekte iken, oturumda kullanılan ortamın değişimine izin verir. Örneğin, çağrı başlangıçta yalnızca ses iletimi için kurulduğu halde daha sonra ihtiyaç duyulduğunda görüntü iletimi de çağrıya eklenebilir.

3.1 SIP Bileşenleri

SIP protokolü temel olarak üç bileşenden oluşur:

- SIP Kullanıcı Arabirimi (SIP User Agent-UA)
- SIP Sunucusu (SIP Server)
- SIP Geçiti (SIP Gateway)

3.1.1 SIP Kullanıcı Arabirimi

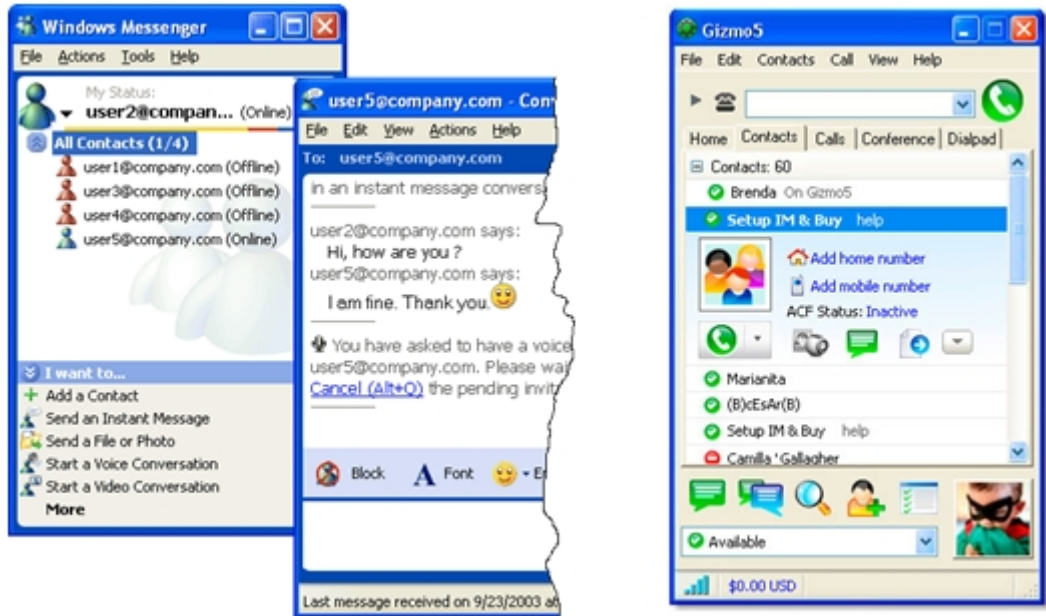
SIP aygıtları olarak bilinen bütün uç birimler, SIP kullanıcı arabirimi olarak tanımlanır. SIP protokolünü destekleyen donanımsal IP telefonlar, yazılımsal IP telefonlar ve cep telefonları, SIP kullanıcı arabirimlerine örnek olarak verilebilir.



Şekil 3.2 Cisco 7970 ve Nortel 1535 donanımsal IP telefon



Şekil 3.3 CounterPath X-Lite yazılımsal IP telefon



Şekil 3.4 Windows Messenger 5.1 ve Gizmo5 yazılımsal IP telefon



Şekil 3.5 Sony Ericsson P990i Wi-Fi üzerinden SIP destekli cep telefonu



Şekil 3.6 Nokia N95 Wi-Fi üzerinden SIP destekli cep telefonu

Kullanıcı arabirimi, istemci ve sunucu kullanıcı arabirimi olmak üzere iki çeşittir.

- **İstemci Kullanıcı Arabirimi (User Agent Client-UAC):** İstemci kullanıcı arabirimi, SIP taleplerini gönderen birimdir.
- **Sunucu Kullanıcı Arabirimi (User Agent Server-UAS):** Sunucu kullanıcı arabirimi, SIP taleplerinin gönderildiği birimdir. UAC' tan gelen SIP talep mesajlarının alınmasından ve bu mesajların cevaplanmasından sorumludur.

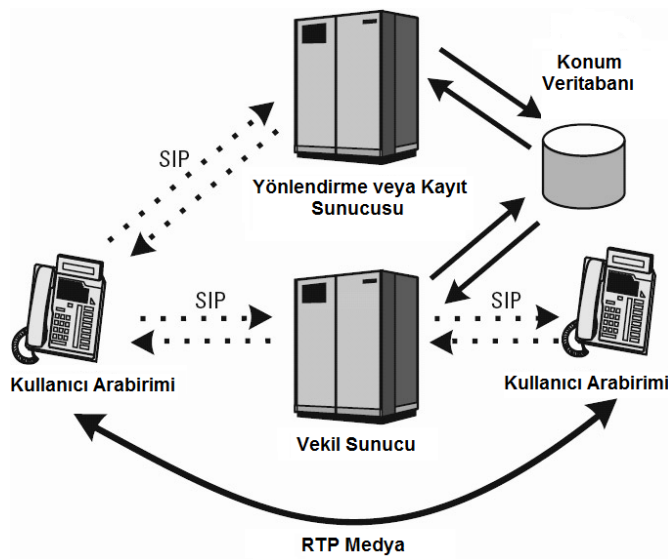
3.1.2 SIP Sunucusu

SIP sunucuları, SIP uç birimlerinin mesaj gönderebilmesi, konum bilgisi kaydı yaptırabilmesi, ağlar arası iletişim kurabilmesi için gerekli SIP elemanlarıdır (Rosenberg vd., 2002). Bu sunucular sayesinde istenilen yönlendirme ve güvenlik işlemleri gerçekleştirilebilir, kullanıcıların kimlik doğrulamaları sağlanabilir. SIP sunucuları çok sayıda bağlantıyı destekleyen, düşük gecikmeli, gerçek zaman performansı yüksek ve ölçeklenebilir sunucular olmalıdır.

SIP protokolünde üç tür sunucu kullanılır:

- Vekil Sunucu (Proxy Server)
- Yönlendirme Sunucusu (Redirect Server)
- Kayıt Sunucusu (Registrar Server)

Sunucuların ve kullanıcı arabirimin ilişkisi Şekil 3.7' de gösterilmiştir.



Şekil 3.7 SIP kullanıcı birimleri ve sunucuları

Vekil Sunucu (Proxy Server): Vekil sunucusu, kendisine gelen talepleri hangi sunucuya ileteceğini kararlaştıran ve gerekli yönlendirmeyi yapan sunucudur. Bu yönlendirmeyi yapacağı birimler başka vekiller yada hedef kullanıcı olabilir. Vekil sunucuları, hem kullanıcı arabirimi adına oturum taleplerini başka birimlere iletir, hem de bu taleplere gelen cevapları kullanıcı arabirime yollar. Böylelikle vekil sunucusu, hem istemci hem de sunucu rolünü üstlenmektedir.

Vekil sunucuların, kullanıcı birimlerden 3 önemli farkı vardır:

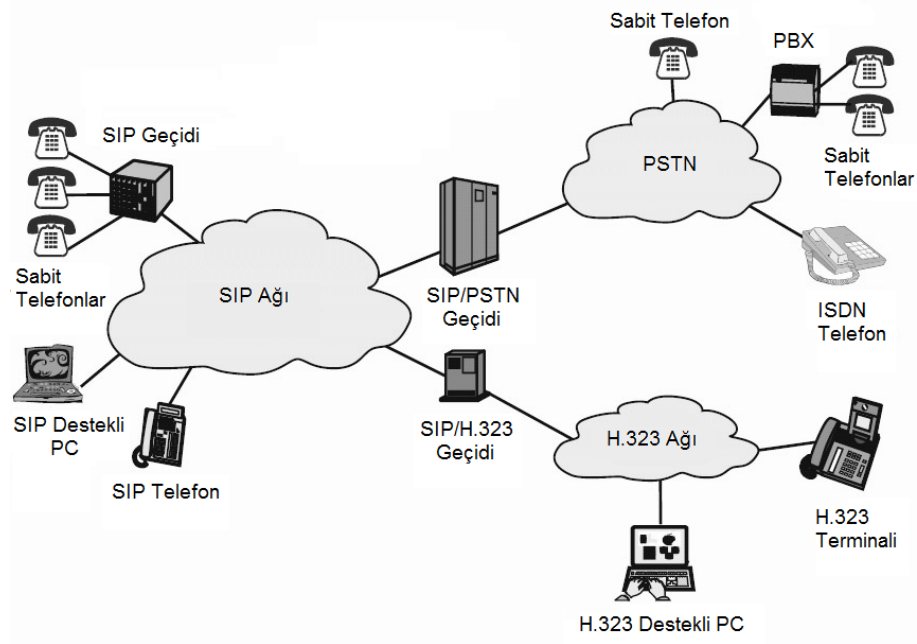
1. Vekil sunucular, talep mesajı gönderemezler. Sadece kullanıcı birimden gelen talep mesajlarını yanıtlarlar (CANCEL talep mesajı bu kuralın haricindedir).
2. Vekil sunucular, medya ortamı oluşturamazlar.
3. Vekil sunucular, mesaj bölümünü ayırtamazlar. Başlık dosyalarına ihtiyaç duyarlar.

Yönlendirme Sunucusu (Redirect Server): Kullanıcı arabiriminden gelen talepleri alır, hedef kullanıcı ile bağlantı kurulabilmesi için çağrı gönderen kullanıcıya hedef kullanıcının adresini gönderir. Kullanıcı adına hedef kullanıcıya herhangi bir talepte bulunamaz.

Kayıt Sunucusu (Registrar): Bütün kullanıcı arabirimlerinin konumlarını kayıt eden bir sunucudur. Buraya kayıt yapan arabirimlerin konum bilgileri (kullanıcının SIP adresi aynı kalabilir fakat kullanıcının konum değişikliğine bağlı olarak bu SIP adresine karşılık gelen IP adresi sürekli değişir) güncellenir. Kayıt sunucusu vekil ile aynı fiziksel ortam içinde bulunabilir.

3.1.3 SIP Geçiti

SIP geçidi, SIP ağlarının ara yüzlerinden yararlanarak protokoller arasında geçişi sağlayan birimlerdir. SIP geçitleri kullanılarak, SIP ağından PSTN, H.323 gibi diğer ağlara iletişim mümkün hale gelir. Şekil 3.8’ de SIP, PSTN ve H.323 ağları gösterilmektedir. SIP ağıyla PSTN ağı arasındaki bağlantı SIP/PSTN geçiti ile sağlanırken, SIP ağıyla H.323 ağı arasındaki bağlantı SIP/H.323 geçiti ile sağlanmıştır.



Şekil 3.8 SIP geçitleriyle kurulmuş hibrid ağ yapısı

3.2 SIP Adreslemesi

Kullanıcılar arasında bağlantı kurulabilmesi için her bir kullanıcıyı tanımlayan belli bir adres olması gereklidir. SIP içindeki bu adresleme sistemi URL - Universal Resource Locator- kullanılarak yapılır (Rosenberg vd., 2002).

SIP URL' si genelde e-posta adresine benzer bir şekilde gösterilir. Örneğin, user@host herhangi bir SIP adresinin yapısını gösterir. Burada user (kullanıcı) bölümünde kullanıcı ismi yada telefon numarası yer alabilir. Host bölümü ise domain ismi yada IP adresinden oluşabilir. SIP adreslemesiyle ilgili örnekler aşağıda listelenmiştir.

nazangonen@biennet.com

nazangonen@138.203.48.64

2165420033@biennet.com

2165420033@138.203.48.64

Yukarıda listenen adreslerin tümü, birbirinin aynıdır ve tek bir kullanıcıya aittir. Kullanıcılar, SIP kayıt sunucularına bu formatta adreslerle kayıt olurlar. SIP kayıt sunucuları akıllı sunuculardır, herhangi bir SIP adresini değişik adreslere çevirebilirler. Örneğin gönderilecek veri ses ise hedef kullanıcının telefonunun adresine, gönderilecek veri video ise hedef kullanıcının bilgisayarının adresine çevrim yapabilir. Bir başka örnek olarak eğer hedef

kullanıcı saat 18:00' den sonra aranıyorsa gelen çağrı kullanıcının iş telefonu yerine ev telefonuna yönlendirilebilir.

3.3 SIP Mesaj Yapısı

SIP karakter tabanlı bir protokoldür ve UTF-8 karakter setini kullanır (Rosenberg vd., 2002). SIP mesajları istemciden sunucuya giden bir talep mesajı formatında ya da sunucudan istemciye giden cevap mesajı şeklindedir. Bütün SIP mesajları bir başlangıç satırı, bir veya daha fazla başlık bölümü ve isteğe bağlı mesaj gövdesinden oluşur.

3.3.1 Başlangıç Satırı

Bütün SIP mesajları bir başlangıç satırı ile başlar. Başlangıç satırı SIP mesajının başlık kısmını tanımlar. Bu alan talep ve cevap mesajları için farklı yapıdadır.

Talep Başlangıç Satırı: Talep mesajları için talep mesajı türü (INVITE, ACK, BYE vs.) başlangıç satırında belirtilir. Talep başlangıç satırına ait bir örnek aşağıda verilmiştir.

INVITE sip:unalkucuk@biennet.com SIP/2.0

Bu talep başlangıç satırı örneğinde ilk bölümde talep metodu(INVITE), ikinci bölümde çağrıyı başlatan kişinin ulaşmak istediği kullanıcının URL adresi(sip:unalkucuk@biennet.com), son olarak ta SIP protokol versiyonu belirtilmiştir.

Cevap Başlangıç Satırı: Cevap mesajı için cevap mesajının kodu (100, 180, 200 vs.) başlangıç satırında belirtilir. Cevap başlangıç satırına ait bir örnek aşağıda verilmiştir.

SIP/2.0 100 Trying

Bu cevap başlangıç satırı örneğinde SIP protokol versiyonu SIP/2.0, mesajın kodu(100) ve bu kodun anlamı(Trying) gösterilmiştir.

3.3.2 Başlık Bölümü

SIP başlık bölümleri mesaj niteliklerini gösteren bölümdür. RFC 3261 standartına göre, başlık bölümleri Çizelge 3.1' de de gösterildiği gibi, 4 gruba ayrılır:

- Hem talep hem cevap mesajlarında bulunabilen genel başlık kısımları
- Sadece talep mesajında bulunan başlık kısımları
- Sadece cevap mesajında bulunan başlık kısımları
- Mesaj gövdesi başlık kısımları

Çizelge 3.1 Başlık bölümü içeriği

Genel Başlık	Mesaj Gövdesi Başlığı	Talep Başlığı (Özgü)	Cevap Başlığı (Özgü)
Accept	Content-Encoding	Authorization	Allow
Accept-Encoding	Content-Length	Hide	Proxy-Authenticate
Accept-Language	Content-Type	Max-Forwards	Retry-After
Call-ID		Organization	Server
Contact		Priority	Unsupported
CSeq		Proxy-Authorization	Warning
Date		Proxy-Require	WWW- Authenticate
Encryption		Route	
Expires		Require	
From		Response-Key	
Record-Route		Subject	
Timestamp		User Agent	
To			
Via			

3.3.2.1 Genel Başlık Kısımları

Accept

Kabul edilen medya tipini belirtir. Örnek:

- Accept: text/*
- Accept: application/sdp
- Accept: application/cpim-pidf+xml

İlk örnekte, text ile şifrelenmiş bütün düzenlerin kabul edileceği belirtiliyor.

Accept-Encoding

Kabul edilen mesaj gövdesi, şifreleme şeması bilgisini içerir. Örnek: Accept-Encoding: text/plain

Accept-Language

Dil seçeneklerini belirtir. Örnek: Accept-Language: fr (Sadece Fransızca kullanılır.)

Call-ID

Call-ID başlık kısmı bütün SIP talep ve SIP cevap mesajlarında kullanılması gereken başlık kısmıdır. İki kullanıcı arasında kimlik tanımlama için kullanılır. Kullanıcı arabirimler için verinin kayıt bilgileri de aynı Call-ID kullanılarak tutulur. Kompakt formu “i” dir. Örnek:

- Call-ID: 34a5d553192cc35@15.34.3.1
- i: 35866383092031257@biennet.com

Contact

Contact başlık kısmı URI kısmını iletmek için kullanılır. Böylece talep ve cevap kaynakları belirlenir. Kompakt Formu “m” dir. Contact başlık kısmı örnekleri:

- Contact: sip:bell@telephone.com (Kullanıcı ismi yazılmadan sadece SIP URI yazılmıştır.)
- Contact: Lentz <h.lentz@petersburg.edu> (İsimle beraber “< >” işaretleri arasında olan URI belirtilmiştir.)
- Contact: M. Faraday <faraday@effect.org>, "Faraday" <mailto:faraday@pop.effect.org> (2 URI bilgisi listelenmiştir.)
- m: <morse@telegraph.org;transport=tcp>; expires= "Fri, 13, Oct 1998 12:00:00 GMT" (Başlık kısmı kompakt form olarak belirtilmiş bir tane URI kullanılmıştır. URI port numarası ve URI parametresi <> işaretleri içerisinde belirtilmiştir.)

CSeq

CSeq başlık kısmı her talepte kullanılır. Her talepte bir artan (CANCEL ve ACK hariç) sayı bilgisi içerir. Örnek: CSeq: 1 INVITE

Date

Talebin veya cevabın gönderildiği tarihi gösterir. Gün, ay, yıl, saat ve zaman dilimi olarak belirtir. Örnek: Date: Fri, 13 Oct 1998 23:29:00 GMT

Encryption

Şifreleme var ise, iletilecek verinin şifreleme metodunu bildirir.

Expires

Talep mesajının geçerli olduğu süre aralığını belirtir. Örnek:

- Expires: 60
- Expires: Fri, 15 Apr 2000 00:00:00 GMT

From

From başlık kısmı talebi gönderen kişinin adres bilgilerini içerir. Her talepte olması gerekir.

From, etiket bilgisi(tag) içerebilir. Etiket, özel aramaların kimliğini belirlemek için kullanılır.

Örnek: From: <sip:armstrong@hetrodyne.com> ;tag=3342436

Record-Route

Record Route, iki kullanıcı arabirim arasındaki taleplerde izlenen yolun güzergâhının kaydedilmesidir. Örnek:

- Record-Route: <sip:proxy1.carrier.com;lr>, <sip:firewall33.corporation.com;lr>
- Record-Route:<sip:139.23.1.44;lr>

Timestamp

UAC' nin gönderdiği talebin oluşturulduğu zamanı, sayı formatında gösterir. Örnek:

Timestamp: 235.15 (Talep için başlangıç zamanı bilgisi bildirildi.)

To

To başlık kısmının, bütün SIP mesajlarında olması zorunludur. Mesajı alacak kişinin adres bilgisini belirtir. Örnek: To: sip:babage@engine.org;tag=2443a8f7

Via

Taleplerin iletilirken kullandığı SIP güzergâhını kaydeder. Bu bilgi cevap gönderilirken izlenecek yolu da belirler. Örnekler:

- Via: SIP/2.0/UDP 100.101.102.103;branch=z9hG4bK776a
(Mesajın, UDP protokolü kullanılarak iletileceği bildirilmiştir. Ayrıca IP adresi belirtilmiştir.)
- Via: SIP/2.0/TCP cube451.office.com:60202;branch=z9hG4bK776a

(Mesajın, TCP protokolü kullanılarak iletileceği bildirilmiştir. Ayrıca alan adı ve port numarası belirtilmiştir.)

- Via: SIP/2.0/UDP 120.121.122.123;branch= z9hG4bK56a234f3.1
(Vekil sunucu tarafından kullanılan güzergâh bilgileri de eklenmiştir.)
- Via: SIP/2.0/TCP 192.168.1.2;received=12.4.5.50;branch=z9hG4bK334
(Bu adres multicast adres tipidir.)

3.3.2.2 Talep Mesajına Özgü Başlık Kısımları

Authorization

Kullanıcı arabirimin niteliklerini taşımak için sunucudan talep edilen izindir.

Hide

Via başlık kısmındaki yol bilgilerinin gizlenmesini sağlar.

Max-Forwards

SIP talebinin yapabileceği maksimum gönderme sayısını gösterir. Örnek: Max-Forwards: 10

Organization

Mesajların ait olduğu kaynağın organizasyonunu belirtmek için kullanılır. Örnek: Organization: MCI

Priority

UAC tarafından talebin aciliyetini bildirmek için kullanılır. Örnek: Priority: acil

Proxy-Authorization

Kullanıcı arabirimin niteliklerini taşımak için sunucudan talep edilen izindir.

Proxy-Require

Sunucunun desteklediği taleplerin özelliklerini ve eklentilerini belirtmek için kullanılır. Örnek: Proxy-Require: timer

Route

Route, talep bilgilerinin yol güzergâhının bildirilmesini sağlar. Örnek: Route: <sip:proxy@example.com;lr>

Require

UAC' ın desteklediği özelliklerin veya eklentilerin UAS' e bildirilmesini sağlar. Örnek: Require: rel100

Response-Key

Şifreleme bilgilerini içerir.

Subject

Talep mesajı ile birlikte, yapılan çağrının amacını diğer bir deyişle konusunu belirtir. Kompakt formu “s” ile gösterilir. Örnek:

- Subject: Hafta sonu ne yapıyoruz?
- s: Final sınavları tarihleri ilan edildi.

User-Agent

Talebi oluşturan kullanıcı arabirimin, kullandığı araç bilgisini belirtir. Örnek:

- User-Agent: Acme SIP Phone v2.2
- User-Agent: MobileSIPClient

3.3.2.3 Cevap Mesajına Özgü Başlık Kısımları

Allow

Allow başlık kısmı, kullanıcı arabirim veya vekil sunucusunun, gönderdiği cevap mesajlarıyla desteklediği metotları bildirmesinde kullanılır. Örnek: Allow: INVITE, ACK, BYE, INFO, OPTIONS, CANCEL

Proxy-Authenticate

Sunucu tarafından UAC' a değişimin doğrulanmasını tasdik etme bilgisini gönderir.

Retry-After

Kaynağın ya da hizmetin tekrar ulaşılacağı zamanı gösterir. Örnekler:

- Retry-After: 3600 (Talebin 1 saat sonra tekrarlanabileceğini gösterir.)
- Retry-After: Sat, 21 May 2000 08:00:00 GMT (Talebin tekrarlanabileceği günü gösterir.)
- Retry-After: Mon, 29 Feb 2000 13:30:00 GMT ;duration=1800 (Talebin tekrarı, ilan edilen gün ve saat içinde 30 dakikalık bir zaman içinde olabileceğini gösterir.)

Server

Cevabın geldiği sunucu bilgisini belirtir. Örnek: Server: Dotcom Announcement Server B3

Unsupported

Sunucu tarafından desteklenmeyen özellikleri bildirir. Örnek: Unsupported: rel100

Warning

Alınan talep mesajında desteklenmeyen özellikler için uyarı belirtir. Cevaplar kod olarak iletilir. Örnek: Warning: 302 proxy "Incompatible transport protocol"

WWW-Authenticate

Kullanıcı arabirim tarafından ya da sunucu tarafından UAC' ye gönderilir. İzinsiz girişleri bildirir.

3.3.2.4 Mesaj Gövdesi Başlık Kısımları

Content-Encoding

İletilecek verinin, kodlama tekniği bilgisini içerir. Kompakt formu “e” dir.

- Content-Encoding: text/plain
- e: gzip

Content-Length

Mesaj gövdesinin uzunluğunu belirtir. Kompakt gösterimi “l” dir. Örnek:

- Content-Length: 0
- l: 287

Content-Type

İletilecek verinin, medya tipini gösterir. Kompakt formu “c” dir. Örnek:

- Content-Type: application/sdp
- c: text/html

SIP başlık kısımlarının kompakt formları, Çizelge 3.2’ de gösterilmiştir.

Çizelge 3.2 SIP başlık kısımlarının kompakt formları

Başlık Kısım	Kompakt Form
Accept - Contact	a
Allow - Event	u
Call - ID	i
Contact	m
Content - Encoding	e
Content - Length	l
Content - Type	c

Başlık Kısmı	Kompakt Form
Event	o
From	f
Refer-To	r
Referred-By	b
Reject-Contact	j
Subject	s
To	t
Via	v

3.3.3 Mesaj Gövdesi

Mesaj gövdesinde kurulacak oturumun özellikleri belirtilir. Oturumda kullanılacak ses ve video kodekleri, örnekleme oranları gibi parametreler bu kısımda gösterilir. SDP (Session Description Protocol – Oturum Tanımlama Protokolü) mesaj gövdesinde yer alabilecek protokollerden biridir. Her talep ve cevap mesajı, mesaj gövdesi içerebilir. Fakat zorunlu değildir. Örnek olarak ACK, INVITE ve OPTIONS belli bir mesaj gövdesi (SDP) içerirken, BYE mesaj gövdesi gerektirmez.

3.3.3.1 Oturum Tanımlama Protokolü(SDP)

SIP çoklu ortam uygulamalarında çağrının başlatılmasını, yürütülmesini ve sonlandırılmasını tanımlayan bir protokol iken, SDP oturumun kendisini tanımlayan (bant genişliği bilgisi, ortam bilgisi vs.) bir protokoldür (Handley vd., 2006). SDP ile tanımlanan oturum parametreleri, Çizelge 3.3’ te gösterilmiştir.

Çizelge 3.3 SDP içeriği

Alan	İsim	Zorunlu/Opsiyonel
v =	Protokol versiyon numarası	Z
o =	Sahibi/Yaratıcısı ve oturum kimliği	Z
s =	Oturum ismi	Z
c =	Bağlantı bilgisi	Z
t =	Oturum başlangıç ve bitiş zamanı	Z

Alan	İsim	Zorunlu/Opsiyonel
e =	Email adresi	O
p =	Telefon numarası	O
i =	Oturum bilgileri	O
b =	Band genişliği bilgisi	O
u =	Değişmeyen kaynak kimliği	O
r =	Tekrar zamanı	O
z =	Zaman dilimi doğrulaması	O
k =	Şifre çözme anahtarı	O
a =	Medya Nitelik tipi	O
m =	Medya bilgisi	O

Aşağıda SDP yapısına ait bir örnek verilmiştir.

v=0

o=johnston 2890844526 2890844526 IN IP4 43.32.1.5

s=SIP Tutorial

i=This broadcast will cover this new IETF protocol

u=http://www.digitalari.com/sip

e=Alan Johnston alan@mci.com

p=+1-314-555-3333 (Daytime Only)

c=IN IP4 225.45.3.56/236

b=CT:144

t=2877631875 2879633673

m=audio 49172 RTP/AVP 0

a=rtpmap:0 PCMU/8000

m=video 23422 RTP/AVP 31

a=rtpmap:31 H261/90000

Protokol Versiyonu

“v” alanı, SDP versiyon numarasını içerir. Geçerli versiyon numarası 0’ dır. Geçerli SDP mesajları daima v=0 ile başlayacaktır.

Orijin (Köken, kaynak)

“o” alanı, oturum ve oturum kimliği kaynağı hakkında bilgi verir. Oturumu başlatan kullanıcının IP adresi bilgisini içerir. Gösterimi aşağıdaki gibidir.

o=(kullanıcı ismi)(oturum-ID)(versiyon)(ağ tipi)(adres tipi)(adres)

Oturum İsmi ve Bilgileri

“s” alanı, oturumun ismini(konusunu) belirler. Bu alan, sayı karakteri içermez.

URI

Opsiyonel olan “u” alanı, sabit kaynak göstergesi (URI-uniform resource indicator) ile oturum hakkında daha fazla bilgi içerir.

Email Adresi ve Telefon Numarası

Opsiyonel olan “e” alanı oturum, mesajı gönderen kişinin adres bilgisini içerir.

Opsiyonel olan “p” alanı telefon numarası bilgisini içerir. Telefon numarası uluslararası formatta verilir. Yorum satırı " () " içine yazılabilir.

Bağlantı Verisi

“c” alanı, medya bağlantıları hakkında bilgi içerir. Gösterimi aşağıdaki gibidir.

c=(ağ tipi)(adres tipi)(bağlantı adresi)

Adres tipi IPv4 için IP4, IPv6 için IP6 olarak tanımlanmıştır. Bağlantı adresi medya paketlerinin gönderildiği IP adres bilgisini içerir.

Band Genişliği

Opsiyonel olan “b” alanı, gerekli band genişliği hakkında bilgi içerir. Gösterimi aşağıdaki gibidir.

b= modifier(anlam değiştirici): band genişliği değeri

Band genişliği değeri kb/sn cinsinden yazılır.

Zaman, Tekrar Zamanı ve Zaman Dilimi

“t” alanı, oturumun başlangıç zamanı ve bitiş zamanı bilgilerini içerir. Gösterimi aşağıdaki gibidir.

$t = (\text{başlangıç zamanı})(\text{bitiş zamanı})$

“r” alanı, opsiyonel olmakla beraber oturum tekrar zamanı bilgisini içerir. Gün, saat veya dakika cinsinden olabilir. Opsiyonel olan “r” alanı ise zaman dilimi bilgilerini içerir.

Şifre çözme anahtarı

Opsiyonel olan “k” alanı, medya oturumları için şifre çözücü olarak kullanılır. Gösterimi aşağıdaki gibidir.

$k = \text{metod}; \text{şifre çözme anahtarı}$

Medya bilgisi

Opsiyonel olan “m” alanı, medya oturum türü hakkında bilgi içerir. Gösterimi aşağıdaki gibidir.

$m = (\text{medya}) (\text{port}) (\text{transport}) (\text{payload tipi})$

Medya türü,

- Ses(Audio)
- Görüntü(Video)
- Uygulama(Application)
- Veri(Data)
- Telefon olayı(Telephony Event)
- Kontrol(Control)

olabilir. Port parametresi, gelen paketleri almak için sürekli dinlenilecek port numarası bilgisini içerir. Transport parametresi, taşıma bilgileri parametrelerini içerir. Payload tipi, oturumda kullanılacak kodlama bilgisini içerir.

Nitelikler

Opsiyonel olan “a” alanı, oturumda kullanılacak kodlama bilgisini içerir. Yukarıdaki örnekte, payload tipi “0” dır. “0” payload tipi, G.711 PCMU kodeğine karşılık gelir. Bu, oturumda G.711 PCMU kodeğinin kullanılacağını belirtir.

3.4 SIP Mesajları

SIP haberleşmesinde iki tip mesaj tanımlanmıştır. Bu mesajlar kullanıcının sunucuya gönderdiği talep (request) mesajları ve sunucunun kullanıcıya gönderdiği cevap (response) mesajlarıdır.

3.4.1 SIP Talep Mesajları

RFC 3261 standardında 6 SIP metodu yer almaktadır (Rosenberg vd., 2002). SIP protokolü, zamanla genişletilerek 7 SIP metodu daha geliştirilmiştir.

INVITE, REGISTER, BYE, ACK, CANCEL ve OPTIONS metodları SIP’ te bulunan orijinal 6 metoddur. REFER, SUBSCRIBE, NOTIFY, MESSAGE, UPDATE, INFO ve PRACK metodları diğer RFC dokümanlarında bulunan sonradan geliştirilmiş metodlardır.

3.4.1.1 INVITE

Çağrıyı başlatmak için kullanılan mesaj tipidir. INVITE içindeki mesaj bölümünde oturumun ne tür bir yapıda olduğu, çağrı yapılan kullanıcıya belirtilir. İkili görüşmeler için çağrıyı yapan hangi tür ortam üzerinden haberleşmek istediğini bu bölümde gösterir. INVITE talebinin zorunlu başlık kısımları; Call-ID, Cseq, From, To, Via, Contact, Max-Forwards başlık kısımlarıdır.

3.4.1.2 REGISTER

Kullanıcıların, Contact URI, IP adresi ve bu Contact için izlenecek yol talebi gibi kayıt bilgilerinin kayıt sunucusuna bildirilmesinde kullanılır. REGISTER talebinin zorunlu başlık kısımları; Call-ID, Cseq, From, To, Via, Max-Forwards başlık kısımlarıdır.

3.4.1.3 BYE

Oturumu sonlandırmak için kullanılan mesajdır. İki taraftan biri çağrıyı sonlandırmak için BYE talebinde bulunmalıdır. BYE talebinin zorunlu başlık kısımları; Call-ID, Cseq, From, To, Via, Contact, Max-Forwards başlık kısımlarıdır.

3.4.1.4 ACK

Çağrı başlatma talebinin sonucunu bildirir. Çağrıyı başlatanın davet talebine karşılık son cevabı aldığını belirtir. ACK talebinin içinde mesaj bölümü olması opsiyoneldir. Bu bölüm boş ise çağrıyı başlatan INVITE talebindeki ortam parametrelerini kullanıyor demektir. ACK talebinin zorunlu başlık kısımları; Call-ID, Cseq, From, To, Via, Max-Forwards başlık kısımlarıdır.

3.4.1.5 CANCEL

Devam eden çağrı talebinin iptal edilmesi için kullanılır. CANCEL talebinin zorunlu başlık kısımları; Call-ID, Cseq, From, To, Via, Max-Forwards başlık kısımlarıdır.

3.4.1.6 OPTIONS

Bu mesaj ulaşılmak istenen kullanıcıya ilgili sunucunun bağlantı kurup kuramayacağını anlamak için çağrıyı başlatan kullanıcı tarafından yollanır. Eğer kullanıcı sunucunun üzerindeyse bu mesaja cevap verebilir. Örneğin kullanıcının bulunamadığını yada meşgul olduğunu bildirir. OPTIONS talebinin zorunlu başlık kısımları; Call-ID, Cseq, From, To, Via, Max-Forwards başlık kısımlarıdır.

3.4.1.7 REFER

REFER metodu kullanıcı arabirim tarafından, başka bir arabirimin URI ve URL kaynak bilgilerine ulaşmak için kullanılır. REFER talebinin zorunlu başlık kısımları; Call-ID, Cseq, From, To, Via, Contact, Max-Forwards, Refer-To başlık kısımlarıdır.

3.4.1.8 SUBSCRIBE

SUBSCRIBE metodu, kullanıcı arabirim tarafından durum bilgisi gibi özel olaylar(event) için bildirimlerin iletildiği abonelik gerektiren oturumlar kurmak için kullanılır. SIP protokolünde UNSUBSCRIBE metodu yoktur. Onun yerine SUBSCRIBE mesajı EXPIRES: 0 başlığıyla gönderilir. Böylece abonelik iptal edilir. Abonelik kaldırıldığında, NOTIFY mesajı ile üyeliğin iptal edildiği bildirilir. SUBSCRIBE talebinin zorunlu başlık kısımları; Call-ID, Cseq, From, To, Via, Max-Forwards, Contact, Expires başlık kısımlarıdır.

3.4.1.9 NOTIFY

NOTIFY metodu, kullanıcı arabirim tarafından durum bilgisi gibi özel olaylar(event) ile ilgili bilgileri iletmek için kullanılır. NOTIFY talebinin zorunlu başlık kısımları; Call-ID, Cseq, From, To, Via, Contact, Max-Forwards başlık kısımlarıdır.

3.4.1.10 MESSAGE

MESSAGE metodu, kullanıcılar arası anlık mesajlaşmayı sağlar. Mesaj içeriği mesaj gövdesinde taşınır. MESSAGE talebinin zorunlu başlık kısımları; Call-ID, Cseq, From, To, Via, Max-Forwards başlık kısımlarıdır.

3.4.1.11 INFO

Kullanıcı arabirim oturum durumunu değiştirmeden oturum bilgisini diğer kullanıcıya gönderebilmesini sağlayan metottur. INFO talebinin zorunlu başlık kısımları; Call-ID, Cseq, From, To, Via, Max-Forwards başlık kısımlarıdır.

3.4.1.12 PRACK

Geçici cevap mesajlarının (1xx sınıfı) doğru olarak alındığını doğrulayan mesajdır. 2xx, 3xx, 4xx, 5xx ve 6xx mesajlarının doğruluğu ise ACK metodu uygulanarak belirlenir. PRACK metodu 100 Trying cevabı haricinde tüm geçici cevap uygulamalarında kullanılabilir. PRACK talebinin zorunlu başlık kısımları; Call-ID, Cseq, From, To, Via, Max-Forwards, RACK başlık kısımlarıdır.

3.4.1.13 UPDATE

Bu metot, dialog durumunu değiştirmeden oturum bilgilerini değiştirmek için kullanılır. Kurulmuş bir oturumda re-INVITE mesajı gönderilerek oturum bilgileri yenilenir. Ancak askıda olan oturumda (INVITE mesajı gönderilmiş fakat nihai cevap alınamamış), re-INVITE yerine UPDATE metodu kullanılır. UPDATE talebinin zorunlu başlık kısımları; Call-ID, Cseq, From, To, Via, Contact, Max-Forwards başlık kısımlarıdır.

3.4.2 SIP Cevap Mesajları

SIP cevap mesajları, SIP talep mesajlarına cevap olarak gönderilen mesaj tipleridir. Mesajlar HTTP cevap kodlarına benzer sayısal kodlar içerir. Bu mesaj tipleri,

- Geçici Mesajlar (1xx)
- Sonuç Mesajları (2xx, 3xx, 4xx, 5xx, 6xx)

olarak iki kısımda incelenebilir. Geçici mesajlar talep mesajının sonuçlanması sürecine kadar gönderilen bilgilendirme mesajlarıdır. Sonuç mesajları ise çağrı talebinin sonucunu bildirir.

3.4.2.1 1xx Cevap Mesajları

Bilgilendirme amaçlı cevap mesajlarıdır.

- 100 Deniyor
- 180 Çalışıyor
- 181 Arama Yönlendiriliyor
- 182 Sıraya Alındı
- 183 Oturum Devam Ediyor

3.4.2.2 2xx Cevap Mesajları

Talebin başarıyla alındığına dair cevap mesajlarıdır.

- 200 TAMAM
- 202 Kabul Edildi: Göndermeler için kullanılır

3.4.2.3 3xx Cevap Mesajları

Talebin yönlendirildiğine dair cevap mesajlarıdır.

- 300 Çoklu Seçenekler
- 301 Kalıcı Olarak Yeri Değişti
- 302 Geçici Olarak Yeri Değişti
- 305 Proxy Kullan
- 380 Alternatif Servis

3.4.2.4 4xx Cevap Mesajları

Talebin kullanıcı hatasından dolayı kabul edilmediğine dair hata cevap mesajlarıdır. Kullanıcı talep mesajını yinelemelidir.

- 400 Geçersiz Talep
- 401 Yetki Dışı Kullanım: Sadece kayıt hizmeti verenler tarafından kullanılır Proxy'ler, proxy yetkilendirmesi için 407' yi kullanmalıdır
- 402 Ödeme Gerekli (gelecekte kullanılmak üzere ayrılmıştır)
- 403 Yasak
- 404 Bulunamadı: Kullanıcı bulunamadı
- 405 İzin Verilmeyen Yöntem
- 406 Kabul Edilemez

- 407 Proxy Kimlik Doğrulanması Gerekıyor
- 408 Talep Zaman Aşımı: Kullanıcı gerekli süre içerisinde bulunamadı
- 410 Gitmiş: Bir zamanlar var olan bu kullanıcı artık burada yok
- 413 Talep Çok Büyük
- 414 Talep-URI Çok Uzun
- 415 Desteklenmeyen Medya Tipi
- 416 Desteklenmeyen URI Şeması
- 420 Geçersiz Uzantı: Geçersiz SIP Protokol Uzantısı kullanıldı, sunucu tarafından anlaşılmadı
- 421 Uzantı Gerekıyor
- 423 Ara Çok Kısa
- 480 Geçici Olarak Ulaşılamıyor
- 481 Arama/İşlem Mevcut Değil
- 482 Döngü Tespit Edildi
- 483 Çok Fazla Sayıda Sekme
- 484 Eksik Adres
- 485 Belirsiz
- 486 Burası Meşgul
- 487 Talep Sona Erdirildi
- 488 Burada Kabul Edilemez
- 491 Talep Beklemede
- 493 Deşifre Edilemiyor: S/MIME metni deşifre edilemiyor

3.4.2.5 5xx Cevap Mesajları

Talebin sunucu hatasından dolayı kabul edilmediğine dair hata cevap mesajlarıdır. Kullanıcı talep mesajını başka bir sunucuda yinelemelidir.

- 500 Dahili Sunucu Hatası
- 501 Geçerli Değil: SIP talep metodu burada geçerli değildir
- 502 Geçersiz Ağ Geçidi
- 503 Hizmet Sunulamamaktadır
- 504 Sunucu Zaman Aşımı
- 505 Sürüm Desteklenmiyor: Sunucu, SIP protokolünün bu sürümünü desteklemiyor
- 513 Mesaj Çok Büyük

3.4.2.6 6xx Cevap Mesajları

Talebin genel bir hatadan dolayı kabul edilmediğine dair hata cevap mesajlarıdır. Kullanıcı talep mesajını yinelememelidir.

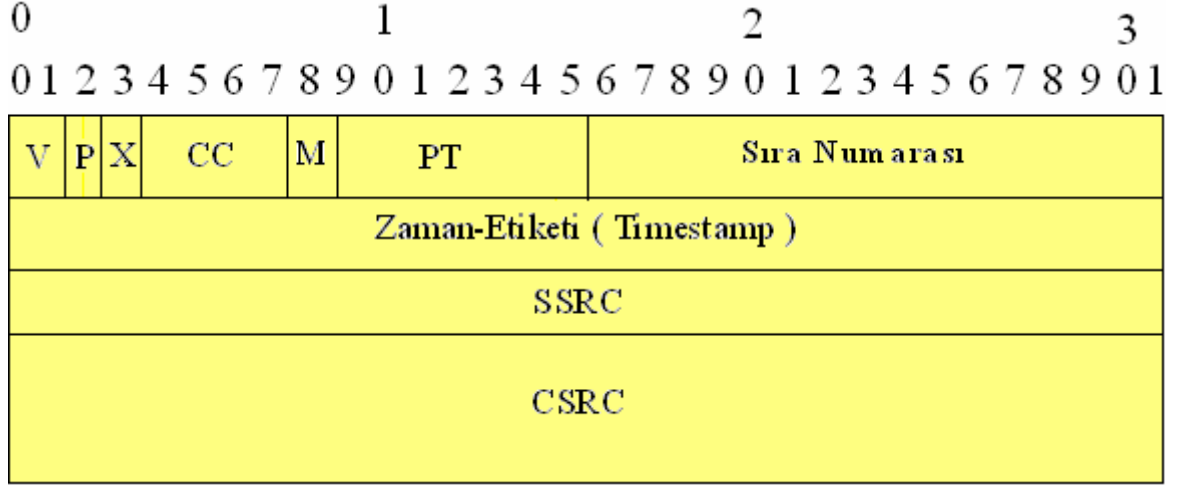
- 600 Her Yer Meşgul
- 603 Reddedilme
- 604 Hiçbir Yerde Mevcut Değil
- 606 Kabul Edilemez

4. GERÇEK ZAMANLI VERİ TAŞIMA PROTOKOLÜ(RTP)

RTP kavramı birbiriyle çok yakından ilişkili iki kısımdan oluşur; bunlardan biri, gerçek zamanlı veriyi taşımakla ilgili olan RTP' dir; diğeri ise, bir görüşme esnasında servis kalitesini kontrol etmek ve katılımcılar hakkında bilgi aktarımı yapmak için kullanılan RTCP (RTP Control Protocol – RTP Kontrol Protokolü)' dir. RTP uygulaması genelde ayrı bir katman olarak ele alınmaz, uygulama işlemlerine bütünleştirilerek gerçekleştirilir. Değişiklik ve düzenlemelere açık olması için, RTP çatısı (framework) kasten gevşek tanımlanmıştır. RTP ek olarak, belirli bir uygulamanın eksiksiz bir tanımı, yük (payload) yapısı ve yük profili tanımlamasını da içerir. Yük yapısı, belirli bir yükün (örn: ses, görüntü) RTP' de nasıl taşınacağını tanımlar. Profil tanımlaması ise, ilgili yük tipleri kodları kümesinin yük yapılarına karşılık gelen kümesini belirtir.

RTP oturum kurulumu, RTP ve RTCP için birer tane taşıma katmanı adresi tanımlanmasını içerir. Bu adreslerin her biri birer adet IP adresi ve UDP port numarasından oluşur. Çoklu-yayın görüşmesi durumunda IP adresi D sınıfı bir çoklu-yayın adresidir. Bir çoklu-ortam oturumunda her ortam, oturum kalitesini raporlayan kendi RTCP paketlerine sahip farklı RTP oturumlarıyla taşınır (Schulzrinne, 2003a; 2003b). İlave ortamlara genellikle ilave port çiftleri tahsis edilir ve görüşme için yalnızca tek bir çoklu-yayın adresi kullanılır.

Bir kodek ile kodlanan gerçek zaman sesi RTP başlığı ile paketlenir. RTP, eş zamanlama gibi yöntemler kullanarak ağ gecikme değişkenliğinin etkilerini azaltmaktadır. Geç iletilen paketler alıcı tarafından düşürüleceği için eş zamanlama ile ağdaki gecikme değişkenliği büyüdükçe ses parçalarının alıcı tarafında işlenmesi için gereken bekleme süresi (playout time) artırılır. RTP başlık yapısı Şekil 4.1' de gösterilmiştir.



Şekil 4.1 RTP başlık yapısı

Şekil 4.1’ de gösterilen alanlar ve açıklamaları aşağıdaki gibidir:

- V : Version Information (Versiyon Bilgisi), RTP’ nin değişik versiyonları arasında ayırım yapabilmek için kullanılır. (2 bit)
- P : Padding (Dolgu), eğer değeri 1 olarak atanmışsa; paketin sonunda bir ya da daha fazla, yükün parçası olmayan dolgu baytları mevcut demektir. Dolgu bitleri sabit boyutlu başlıklarda bazı kriptolama algoritmaları için kullanılabilir. (1 bit)
- X : Extension (Uzantı), eğer değeri 1 olarak atanmışsa; başlığın sonunda kesinlikle bir adet başlık uzantısı mevcut demektir. (1 bit)
- CC : CSRC count (Katılımcı Kaynağı Tanımlayıcısı - Contributing Source Identifier Count), sabit başlığı takip eden CSRC tanımlayıcı sayısını gösterir. (4 bit)
- M : Marker (İşaretleyici), yük profili tarafından tanımlanır. Çerçeve sınırlarının işaretlenmesi benzeri önemli işlevler için kullanılır. (1 bit)
- PT : RTP yükünün yapısını tanımlar ve uygulama tarafından anlamlandırılacağını belirtir. (7 bit)

- Sıra Numarası: Gönderilen her RTP veri paketiyle birlikte, değeri bir artırılır. Alıcı tarafından paket kayıpları tesbiti ve paket sıralamasının yeniden oluşturulması için kullanılabilir. (16 bit)
- Zaman-Etiketi: RTP veri paketindeki ilk bayta ait örnekleme zaman kesitini içeren, ortama özgü bir zaman etiketidir. Örnekleme kesiti, eş zamanlılığa imkan tanımak ve gecikme değişkenliği hesaplarını yapabilmek için, zamanda doğrusal ve tekdüze artan bir saatten türetilmiş olmalıdır. (32 bit)
- SSRC : (Eş Zamanlama Kaynağı Tanımlayıcısı - Synchronisation Source Identifier), eş zamanlayıcı kaynağı tanımlar. Tanımlayıcı, bir oturum içinde iki ayrı kaynağın aynı tanımlayıcıya sahip olmaması için, rastgele seçilir. Kaynak tanımlamanın amacı, tek bir zamanlama ve sıra numaralama uzayına sahip olmaktır. Yerel ağ adresini (örn: IP adresi) kullanmak yeterli değildir, çünkü bu adres yegane olmayabilir. RTP ayrıca taşıma katmanından bağımsız olmak üzere tasarlanmıştır. RTP çeviricileri, değişik adres uzaylarına sahip farklı ağlar arasındaki uyumluluğu sağlamak üzere tasarlanmıştır. Rastgele seçilmiş bir SSRC kullanmak, iki farklı adres uzayında tahsis dizileri kullanmaktan hem daha basittir hem de daha düşük adres çakışması ihtimaline sahiptir. Bir oturumda aynı SSRC tanımlayıcısına sahip iki kaynağın bulunduğu az rastlanır durumlarda çarpışmaya dair problem çözme mekanizmaları devreye girer (Schulzrinne, 2003a; 2003b). (32 bit)
- CSRC, RTP paketindeki yüke katılım yapan kaynakları listeler (Schulzrinne, 2003a; 2003b). CSRC tanımlayıcıları RTP karıştırıcılarında (mixer) değişik kaynaklardan gelen RTP paketleri tek bir RTP paketi içine yerleştirildiğinde eklenir. Yani değişik kaynaklardan gelen ses paketlerinin SSRC tanımlayıcıları CSRC içine yerleştirilir ve yeni oluşturulan bu paket için karıştırıcı SSRC' si kullanılır. Bu işlem, alıcı tarafındaki konuşma tesbitinin doğruluğu için gereklidir. ((0-15) * 32 bit)

Bir tekli-yayın (unicast) oturumunda, eğer alıcı ve verici ağ protokolünün kaynak alanını kullanarak birbirlerini tanımlayabiliyorsa (internet üzerinden iki kişinin konuşmasında, kaynak adresinin IP adresiyle tanımlanabiliyor olması gibi), SSRC ve CSRC doğal olarak gereksizdir.

RTP kontrol protokolü, belli bir oturumdaki kontrol paketlerinin, oturumdaki tüm katılımcılara periyodik olarak gönderilmesine dayanır (Schulzrinne, 2003a; 2003b). Kontrol paketlerinin dağıtılışı veri paketlerinininkiyle aynıdır. Her RTCP paketi, uygulamaya yönelik, alıcı ve/veya verici istatistik raporlarını içerir. Bu istatistikler, gönderilen paket sayısı, kayıp paket sayısı, gecikme değişkenliği süreleri, son SR (Gönderici Raporu - Sender Report)' dan bu yana olan gecikme, son SR zamanı gibi değerleri içerir.

RTCP dört farklı işleve sahiptir:

1. Temel işlevi veri dağıtımının kalitesi hakkında geri besleme yapmaktır. Bu geri besleme adaptif kodlama kontrolü için kullanılabilir. IP çoklu-yayın deneyleri sonucunda görülmüştür ki geri besleme, dağıtım hatalarını teşhis etmek için de kritik bir role sahiptir. Geri besleme işlevi alıcı ve verici raporlarıyla elde edilir.
2. RTCP bir oturumdaki bütün katılımcıların kaydını tutar. Herbir kaynağa ait, CNAME (Takma Ad - Canonical Name) denilen taşıma katmanı tanımlayıcılarını ve de eş zamanlayıcı kaynak tanımlayıcısını taşıyarak bu işlevi yerine getirir. Eş zamanlayıcı kaynak tanımlayıcısı bir oturum içinde değişebilir. CNAME ayrıca ilintili çoklu akışların (ses ve görüntü) eş zamanlaması için de gereklidir. Bir katılımcı oturumu terkettiğinde, RTCP çıkış mesajı gönderilir.
3. RTCP paketleri, bahsi geçen 1 ve 2 no' lu işlevleri yerine getirmesi için sırayla gönderilir, dolayısıyla bu sıranın da kontrol edilmesi gerekmektedir. Bu kontrol de RTCP tarafından gerçekleştirilir. Gözlemlenen katılımcı sayısı, paket gönderme oranını belirleme için kullanılır. Bir oturumdaki katılımcı sayısı ile herbir katılımcının paket gönderme oranı ters orantılıdır.
4. Bahsi geçen ve gerçekleştirilmesi zorunlu olan ilk üç işlevin aksine, isteğe bağlı olarak gerçekleştirilebilen dördüncü işlev ise asgari oturum kontrol bilgisi (örn: katılımcı tanımlanmasının kullanıcı arabiriminde görüntülenmesi) taşımayı amaçlar.

5. WAV SES DOSYASI FORMAT YAPISI

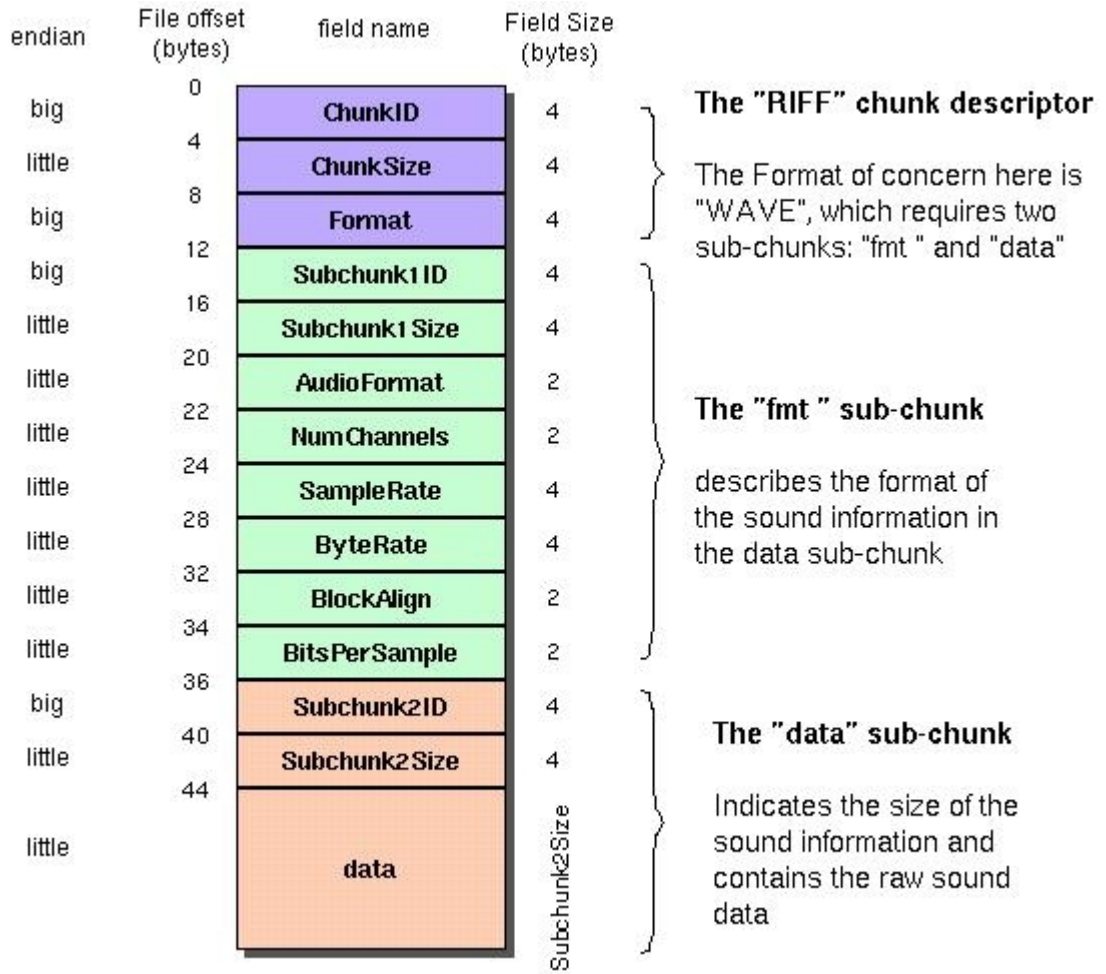
WAV, İngilizce dalga anlamına gelen Wave kelimesinin ilk üç harfinin alınmasıyla oluşturulmuş, sayısal ortamda hiçbir sıkıştırma yöntemi uygulamadan ses saklama biçimidir.

IBM ve Microsoft' un küçük ses kayıtlarını herhangi bir bilgisayarda çalmak için geliştirdiği ses dosyası formatıdır. Yaygın olan formatlar arasında en basitlerinden biridir. Windows 3.1' den bu yana Windows ortamında ses dosyalarının ana formatı olarak kullanılmaktadır. WAV, (.wav) uzantısıyla tanımlanır. Öncelikle kişisel bilgisayarlarda kullanılmasına rağmen, internetteki küçük müzik dosyaları için de yaygın olarak kullanılmaktadır. Macintosh tarafından da geçerli ve kullanılabilir bir araç olarak kabul edilmektedir. Yapısı çok basittir. Mp3 ve diğer sıkıştırılmış formatların aksine WAV' lar sadece sayısallaştırılmış seslerdir. Çok yer kaplarlar, ancak basittirler ve herhangi bir bilgisayarda bile kaliteli ses verebilirler.

WAV format yapısı, Şekil 5.1' de gösterildiği gibi üç kısımdan oluşur (Craig, 2005).

- Dosyayı WAV olarak adlandıran birinci kısım
- Verinin
 - o Medya format
 - o Kanal sayısı
 - o Örnekleme frekansı
 - o Örnek başına düşen bit sayısı
 değişkenlerin tanımlandığı ikinci kısım
- Ham gerçek ses verisinin depolandığı üçüncü kısım

İlk iki kısım birlikte, 44 byte lık WAV başlığını oluştururlar.



Şekil 5.1 WAV dosya formatı (Craig, 2005)

Bu projenin VoIP uygulaması kısmında, bu formata uyularak 44 byte lık WAV başlığı oluşturulmuş ve gerçek ses verisinin başına eklenmiştir.

6. PROJEDE KULLANILAN PROGRAMLAMA DİLLERİ VE ARAÇLAR

“Taşınabilir Sistemlerde SIP Sinyalleşmesi Aracılığıyla VoIP İstemci Uygulaması” projesi, Eclipse geliştirme ortamında, Java programlama dilinin J2ME platformunda, Sun WTK 2.5.2 emülatörü kullanılarak geliştirilmiştir ve uygulamanın sonucuna ait istatistiki veriler Ethereal aracı ile analiz edilmiştir.

6.1 Java

Java, Sun Microsystems tarafından geliştirilmeye başlanmış açık kodlu, nesneye yönelik, platformdan bağımsız, yüksek performanslı, çok işlevli, yüksek seviye, adım adım işletilen (interpreted) bir programlama dildir (Schildt, 2003).

Java ilk çıktığında daha çok küçük cihazlarda kullanılmak için tasarlanmış ortak bir platform dili olarak düşünülmüştür. Ancak platform bağımsızlığı özelliği ve standart kütüphane desteği C ve C++’ tan çok daha üstün ve güvenli bir yazılım geliştirme ve işletme ortamı sunduğundan, hemen her yerde kullanılmaya başlanmıştır (Schildt, 2003).

Bir Java yazılımı şu şekilde geliştirilir:

- Programcı Java kodunu yazar.
- Bu kod bir Java derleyicisi ile derlenir. Sonuçta bytecode adı verilen bir tür makine kodu ortaya çıkar. Platform bağımsızlığını sağlayan şey bytecode’ dur. Çünkü bir kere bytecode oluştuktan sonra yazılım sanal makine içeren tüm işletim sistemlerinde çalışabilir.
- Bu bytecode Java Virtual Machine (Java Sanal Makinesi) tarafından adım adım işletilir.

Bu adımlar aşağıda detaylı bir şekilde anlatılmaktadır.

1. Java Kodunun Yazılması

Java nesneye yönelik bir dil olduğundan tüm yazılım sınıflar ve nesneler üzerinden yürütülür (Schildt, 2003). Sınıflar uygulamadaki nesnelerin tanımlandığı kod parçalarıdır. Java’ da her bir sınıf bir dosya içerisinde yer alır. Dosyaların uzantıları .java şeklindedir. Dosya adı ise içinde tanımlanan sınıf ile aynıdır.

2. JVM - Java Virtual Machine (Java Sanal Makinesi)

Sanal makine donanımdan bağımsız yazılım geliştirme ihtiyacına cevap verme amacıyla geliştirilen bir teknolojidir. Java ile yazılan kaynak kodlar sanal bir işlemcinin anlayacağı makine koduna çevrilir, fakat bu kod bilinen gerçek makine kodu olmadığından bu koda bytecode denir (Schildt, 2003). Yazılan kaynak kod derlenerek bytecode' a dönüştürülür. Bu bytecode' lar programın çalışması esnasında Java Sanal Makinesi tarafından gerçek makine koduna dönüştürülür.

Sonuç olarak Java dili ile yazılan kaynak kodlar .java uzantılı dosyalarda saklanır. javac derleyicisi bu kaynak kodları .class uzantılı dosyalarda saklanan bytecode' a çevirir. Ve bu bytecode' lar java yorumlayıcısı tarafından çalıştırılır.

3. JRE(Java Runtime Engine) ve SDK(Software Development Kit)

JRE java kodlarını çalıştıran sistemdir (Schildt, 2003). Bir makinede bir Java programının çalıştırılabilmesi için JRE yüklenmelidir. Fakat JRE bir derleyici barındırmaz sadece derlenmiş programları çalıştırır. Dolayısıyla Java kodlarının derlenebilmesi için sisteme SDK yüklenmelidir. SDK içerisinde JRE' yi de barındırır dolayısıyla SDK yüklendikten sonra tekrar bir JRE' ye ihtiyaç yoktur.

4. Java Uygulamalarının Çalıştırılması

Java' nın temel felsefesi olan "bir kere yaz, her yerde çalıştır" sanal makine sayesinde varolmuştur. Sanal makine, bir tür hayali bir mikroişlemci gibi düşünülebilir. Gerçek tüm mikroişlemciler (Intel Pentium, AMD Athlon, Sun Sparc vs) belirli bir grup komutu işlemek üzere tasarlanmıştır. Bu komutlara işlemcinin komut kümesi adı verilir. Örneğin x86 komut kümesi gibi. Tüm yazılımlar çalışabilmek için önce bu komut kümesine dönüştürülür, daha sonra işlemci bu komutları sıra ile gerçek işlemci komutlarına dönüştürüp işletir. Java Sanal makinesi de bytecode komut kümesini tıpkı işlemci gibi adım adım işletir. Java' nın interpreted (adım adım işletilen) bir dil olarak adlandırılmasının nedeni budur.

6.2 Java2 Micro Edition (J2ME)

Java 2 Platform, Micro Edition (J2ME platformu), tüketici aygıtları ve gömülü aygıtlar (cep telefonları, PDA' lar, TV dekodeerleri ve diğeer türden tüm taşınabilir ve kablosuz aygıtlar) için tasarlanmış Java platformudur. J2ME, Sun Java Runtime Environment, Standard Edition (JRE) gibi Java platformunun standart yönergelerine uygun olarak üretilmiş ve sektörün önde gelen mobil aygıt üreticileri ile kablosuz hizmet sağlayıcılarının katkılarıyla geliştirilmiştir.

J2ME, cep telefonlarına ya da PDA aygıtlarına yeni uygulamaların gücünü ve en ileri kablosuz bağlantı teknolojisinin avantajlarını kazandıran bir platformdur. J2ME şu anda milyonlarca aygıtta kullanılmaktadır ve bugün dünyanın her yanındaki cep telefonları ile PDA' larda bulunan tüketici araçlarının ve işlevlerin geliştirilmesinde tercih edilen platformdur.

J2ME küçük elektronik cihazlar piyasasının değişik parçaları için tasarlanan teknoloji ve spesifikasyonların bir toplamıdır. J2ME platformunun temel kısmı 2 değişik yapılandırmadan oluşur:

- Bağlantılı cihaz yapılandırıcısı (CDC Connected Device Configuration)
- Bağlantılı sınırlı cihaz yapılandırıcısı (CLDC Connected Limited Device Configuration)

Bir yapılandırıcı (configuration) merkezi Java™ teknolojileri kitaplığını ve sanal makine yeteneklerini tanımlar. CDC yüksek teknoloji taşıyabilir aletler içindir ve CLDC ise genellikle cep telefonları gibi düşük maliyetli taşıyabilir cihazlar içindir.

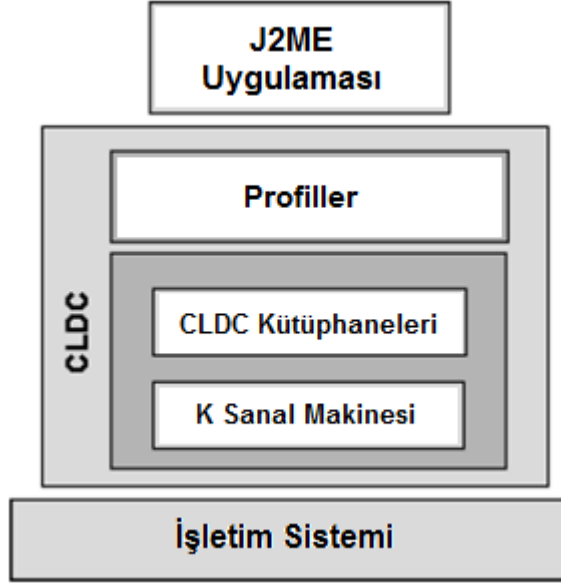
Yapılandırıcılar (configuration) üzerinde özellikli cihaz kategorisinde işlevselliği tanımlayan görüntüleyiciler/profiller(profile) bulunmaktadır. Mobil bilgi cihaz görüntüsü (MIDP Mobile Information Device Profile) cep telefonları gibi iletişim kapasitesine sahip CLDC temelli taşıyabilir aletler için hazırlanan bir görüntüleyicidir (Inc. Sun Microsystems, 2004a; 2004b; Knudsen ve Li, 2005). Bu yapılandırıcılar ve görüntüleyiciler ilerleyen bölümlerde ayrıntılı olarak ele alınacaktır.

6.2.1 J2ME Yapılandırıcıları (J2ME Configurations)

J2ME için iki tane yapılandırıcı tanımlanmıştır.

6.2.1.1 CLDC (Connected Limited Device Configuration)

Connected Limited Device Configuration (CLDC), kısıtlı kaynaklı cihazlar için tasarlanmıştır (cep telefonları, PDA gibi) (Inc. Sun Microsystems, 2004a; 2004b; Knudsen ve Li, 2005). Bundan dolayı herşey optimize edilmiştir, örneğin Sanal Makine (Virtual Machine) küçültülmüş, Java diline ait bazı özellikler desteklenmemiştir. Kütüphane olarak çok küçüktür, fakat mevcut kütüphanesi oldukça işlevseldir.



Şekil 6.1 CLDC mimari yapısı

CLDC' nin temel özellikleri:

- Long ve float veri tipleri desteklenmemektedir. J2SE' nin kütüphaneleri içinde bu veri tiplerini destekleyen sınıflar CLDC' den çıkarılmıştır. (CLDC1.1 bu veri tiplerini desteklemektedir.)
- Çalıştırma zamanı hataları (runtime exception) mevcut fakat sayıca azaltılmıştır. Bunlar
 - o Java.lang.Error
 - o Java.lang.OutOfMemoryError
 - o Java.lang.VirtualMachineError
- Garbage Collection basitleştirilmiştir, java.lang.Object sınıfının içinde finalize metodu yoktur.
- Java Native Interface (JNI) desteklenmemektedir.

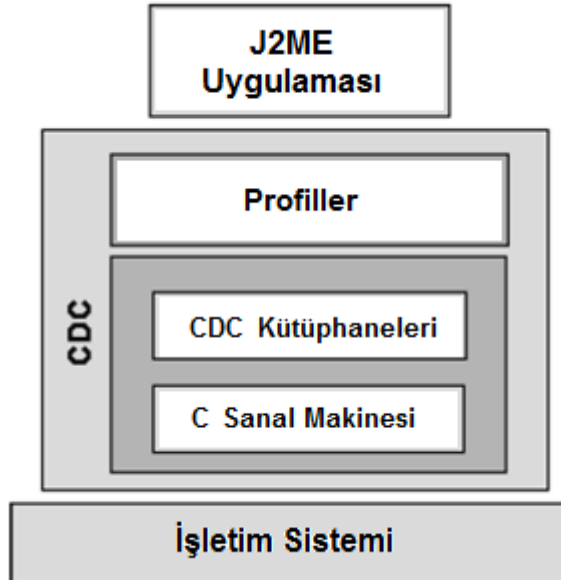
- Grup threadler haricindeki threadler kullanılabilir.
- Farklı güvenlik modeli CLDC' lerde uygulanmaktadır. Standart Edition' daki güvenlik modelinin uygulanmamasındaki en önemli sebep küçük kapasiteli cihazlar için ağır olmasıdır.

CLDC' de 4 paket (package) mevcuttur (Inc. Sun Microsystems, 2004a; 2004b; Knudsen ve Li, 2005). Bunlar:

- java.io
- java.lang
- java.util
- javax.microedition.io

6.2.1.2 CDC (Connected Device Configuration)

Connected Device Configuration(CDC) daha gelişmiş cihazlar için tasarlanmıştır (Inc. Sun Microsystems, 2004a; 2004b; Knudsen ve Li, 2005). Bu cihazlarla kastedilen, CLDC' nin hedef cihazları kadar kısıtlı olmayan cihazlardır. CDC hedef cihazlar 32-bit işlemci ve en az 2 MB belleklere sahiptirler. Java Standart Edition' nın CLDC' ye göre daha fazla özelliğini sunmaktadır. CLDC' ye göre CDC daha fazla API' ye sahiptir.



Şekil 6.2 CDC mimari yapısı

CDC, ağ(network) uygulamaları için daha fazla destek sunmaktadır ve güvenlik mekanizması daha esnektir. CDC, Foundation Profil ile birlikte kullanılmaktadır.

CDC konfigürasyonu aşağıdaki paketleri içermektedir (Inc. Sun Microsystems, 2004a; 2004b; Knudsen ve Li, 2005).

- java.io
- java.lang
- java.lang.ref
- java.lang.reflect
- java.math
- java.net
- java.security
- java.text
- java.text.resources
- java.util
- java.util.jar
- java.util.zip
- javax.microedition.io

6.2.2 J2ME Profilleri (J2ME Profiles)

J2ME’ de 7 tane profil tanımlanmıştır. Bunlar aşağıda açıklanmıştır.

Foundation Profile: CDC yapılandırıcılarında kullanılır.

Game Profile: CDC yapılandırıcılarında kullanılır. Küçük cihazlarda oyun geliştirmek için geliştirilmiştir.

Mobile Information Device Profile: CLDC yapılandırıcısı için geliştirilmiştir. Bu profil,

- Yerel bellek
- Kullanıcı Arayüzü
- Network uygulamaları

sunmaktadır.

Personal Digital Assistant Profile: CLDC yapılandırıcısı için geliştirilmiştir. Bu profil MIDP görüntüsüne göre daha

- Geniş Yerel bellek
- İyi Kullanıcı Arayüzü

sunmaktadır.

Personal Profile: CDC yapılandırıcısı için geliştirilmiştir.

- Kompleks Kullanıcı Arayüzü
- Aynı anda çoklu ekran

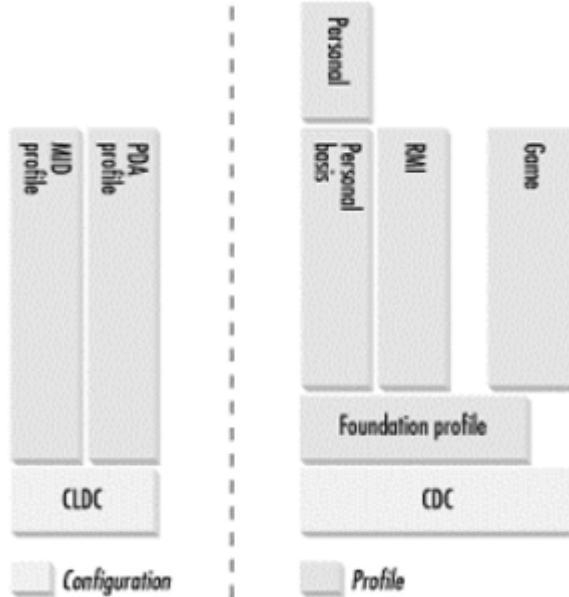
sunmaktadır.

Personal Basis Profile: CDC yapılandırıcısı için geliştirilmiştir.

- Basit Kullanıcı Arayüzü
- Tek ekran

sunmaktadır.

RMI görüntüsü: CDC yapılandırıcısı ve Foundation profili ile kullanılır.



Şekil 6.3 J2ME yapılandırıcı ve profil mimari yapısı

Küçük kapasiteli cihazlar için en yaygın kullanılan profil MIDP profilidir.

6.2.2.1 MIDP (Mobile Information Device Profile)

CLDC yapılandırıcısı (configuration), küçük kapasitedeki cihazlara uygulama yazmak için kullanılmaktadır, fakat bu yapılandırıcı sınırlı işlevsellik sunmaktadır. Örneğin, Kullanıcı Grafik Arayüz (Graphical User Interface) desteği sunmamaktadır. Bu ve diğer nedenlerden dolayı, etkili uygulama geliştirmek için bir tür profile ihtiyaç duyulmaktadır. İhtiyaç duyulan bu profil MIDP (Mobile Information Device Profile) profilidir.

CLDC' nin üzerinde çalışan en önemli profil MIDP (Mobile Information Device Profile)' dir. Bu profile 96x54 ekranı 1 bit görüntüleme derinliği, bir veya iki el için klavyesi olan, dokunmalı ekrana sahip, çift taraflı telsiz bağlantı sağlayabilen aygıtlar için tanımlanmıştır. Bu profile şu paketleri içerir (Inc. Sun Microsystems, 2004a; 2004b; Knudsen ve Li, 2005):

- java.lang
- java.util
- java.io
- javax.microedition.io
- javax.microedition.lcdui
- javax.microedition.midlet
- javax.microedition.rms

MIDlet, CLDC üzerinde MIDP ile çalışan Java uygulamasıdır. Yaygın bir şekilde desteklenen bir profil olduğundan dolayı, J2ME uygulaması ve bir taşınabilir aygıtta Java denildiğinde kasıt muhtemelen bir MIDlet' tir. MIDlet, aygıtta değişik yollarla indirilir ve çalıştırılır. LCD UI (Limited Connected Device User) arayüz için kullanılan sistemdir. RMS (Record Management System) veri saklama ve erişim için kullanılır.

6.2.3 JSR (J2ME Spesifikasyonları)

Bütün J2ME yapılandırıcıları ve görüntüleri(profile) Java Community Process bünyesi altında geliştirilmiştir. JCP, konu ile ilgili büyük endüstri kuruluşları kendi üretimlerinde kullanacakları ortak bir spesifikasyon oluşturdular. Bütün yapılandırıcılar ve görüntüler(profile) Java Specification Request (JSR) altında geliştirilmiştir.

Number	Scope
JSR 30	J2ME Connected Limited Device Configuration (CLDC)
JSR 37	Mobile Information Device Profile for the J2ME Platform (MIDP)
JSR 75	PDA Profile for the J2ME Platform
JSR 36	J2ME Connected Device Configuration (CDC)
JSR 46	J2ME Foundation Profile
JSR 129	Personal Basis Profile Specification
JSR 62	Personal Profile Specification
JSR 66	J2ME RMI Profile
JSR 134	Java Game Profile

Şekil 6.4 JSR örnek listesi

6.3 Sun WTK Yazılım Geliştirme Kiti

Sun Microsystems tarafından geliştirilmiş, JSR spesifikasyonlarına uyumlu çalışabilen taşınabilir aygıtlar için, Java tabanlı uygulamalar geliştirmek için kullanılan bir platformdur. Java uygulamalarının, ekran çıktısını görmek için kullanılan bir kit, emülatör görevi görür. Bu projede kullanılan Sun WTK 2.5.2 ekran arayüzü Şekil 6.5’ te gösterilmiştir.

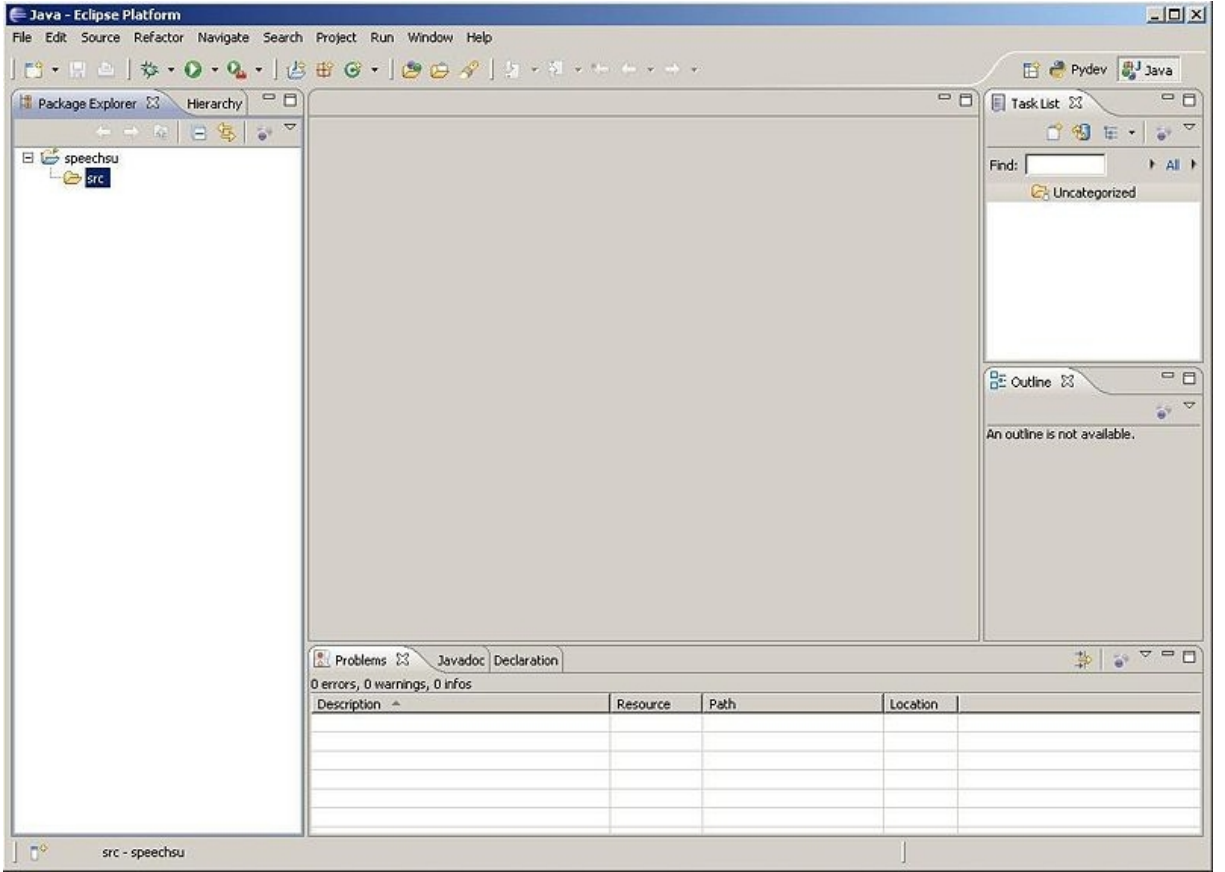


Şekil 6.5 Sun WTK emülatör arayüzü

6.4 Eclipse Geliştirme Ortamı

Eclipse, açık kaynak kodlu bir tümleşik geliştirme ortamıdır (IDE). Ana odak noktası Java ve Java ile ilişkili teknolojiler olsa da esnek yapısı sayesinde C ve Python gibi farklı diller için de kullanılmaktadır.

2001 yılında IBM tarafından başlatılan proje, Java' nın ana grafik sistemi olan Swing yerine bulunduğu platformda bulunan özellikleri doğrudan kullanan SWT' yi kullanarak Java dünyasında tartışmalara yol açmıştır. Hızlı arayüzü, şık görünümü ve çok kuvvetli özellikleriyle kısa zamanda Java geliştiricileri arasında en popüler geliştirme ortamı olmuştur.



Şekil 6.6 Eclipse geliştirme ortamı arayüzü

6.5 Ethereum Ağ Analiz Aracı

Ethereum, Windows, Linux, MacOS veya Solaris gibi bir çok işletim sisteminde çalışabilen ve bilgisayara bağlı olan her türlü ağ kartlarındaki (Ethernet kartı veya modem) tüm TCP/IP mesajlarını analiz edebilen bir programdır. Özellikleri aşağıda listelenmiştir.

- 750' nin üzerinde protokolü analiz edebilir.
- Paketleri yakalayıp bir dosyaya kaydedebilir.
- Daha önceden kaydedilmiş bir dosyayı açabilir.
- Gerçek zamanlı analiz yapabilir.
- Bir analizi filtre edebilir. (örneğin "sadece SIP mesajlarını göster" gibi)
- Terminal veya kullanıcı arabirimi ile kullanılabilir.
- Servis kalitesi istatistiklerini oluşturabilir.
- Sistem istatistiklerini oluşturabilir.

7. TAŞINABİLİR SİSTEMLERDE SIP SİNYALLEŞMESİ ARACILIĞI İLE VOIP İSTEMCİ UYGULAMASI

Kablosuz iletişim sistemlerin, kablolu iletişim sistemlerinin yerini aldığı günümüz haberleşme sektöründe, taşınabilirlik her geçen gün daha çok önem kazanmaktadır. Bu proje ile taşınabilir sistemlerde SIP sinyalleşmesi aracılığı ile bir VoIP istemcisi geliştirmek amaçlanmıştır.

Proje, IBM' in geliştirmiş olduğu Eclipse geliştirme ortamında, J2ME platformunda, Sun WTK 2.5.2 emülatörü kullanılarak geliştirilmiştir. Sunucu – istemci mimari yapısı temel alınarak tasarlanmıştır. Tasarlanan SIP istemcinin haberleşeceği istemci olarak CounterPath X-Lite, Windows Messenger 5.1 ve Gizmo5 gibi yazılımsal Multimedia istemci kullanılmıştır. Tasarlanan SIP istemciden gelen istekleri almak ve diğer istemciden gelen cevapları tasarlanan SIP istemciye iletmek amacıyla bir vekil SIP sunucusu kullanılmıştır. İki istemci arasında gelen, giden SIP mesajlarını, iki istemci arasındaki görüşme sırasında gelen, giden RTP paketlerini analiz etmek ve sistem performanslarını incelemek amacıyla Ethereal aracından faydalanılmıştır.

Tasarlanan SIP istemcinin genel özellikleri aşağıdaki şekilde maddeler halinde özetlenebilir:

- Hesap oluşturma
- Oturum açma
- Oturum kapatma
- Kişi listesini görüntüleme
- Anında kişi durum bilgisi değişikliğini alma
- Kişi ekleme
- Kişi silme
- Anlık ileti gönderme
- Arama başlatma
- Hesap bilgilerini düzenleme

7.1 Tasarlanan SIP İstemcisinin Çalışma Yapısı

Eclipse geliştirme ortamında, J2ME platformuyla yazılan SIP istemciye ait kod derlediğinde ve çalıştırıldığında, Şekil 7.1’ de olduğu gibi Sun WTK 2.5.2 emülatörü devreye girer. Bu istemciye ait telefon numarası 2165420444 olarak atanmıştır.

SIP istemcisinin gerçekleyebileceği 8 temel işlevi vardır. “Ana Menü” altında toplanan bu işlevler

- Oturum aç
- Oturum kapat
- Kişi listesi
- Kişi ekle
- Kişi sil
- Anlık ileti gönder
- Arama başlat
- Hesap bilgilerini düzenle

adlarıyla sıralanır.



Şekil 7.1 Taşınabilir SIP istemcisi

Proje derlenip, çalıştırıldığında konsol çıktısı aşağıdaki gibidir.

MMAPI Versiyon: 1.1

MMAPI tarafından desteklenen SES KODEKLERİ:
 encoding=pcm
 encoding=pcm&rate=8000&bits=8&channels=1
 encoding=pcm&rate=22050&bits=16&channels=2

MMAPI tarafından desteklenen VIDEO KODEKLERİ:
 encoding=rgb565&width=160&height=120
 encoding=rgb565&width=320&height=240
 encoding=rgb565&width=120&height=160

MMAPI tarafından desteklenen GÖRÜNTÜ KODEKLERİ:
encoding=jpeg encoding=png

MMAPI tarafından desteklenen SES DOSYA FORMATI: audio/x-wav
MMAPI Ses Yakalama Özelliği: true
MMAPI Video Yakalama Özelliği: true
MMAPI Kayıt Özelliği: true

AYARLAR YÜKLENİYOR...

Kayıt 5: 47.168.94.32
Kayıt 4: 5060
Kayıt 3: 47.168.16.129
Kayıt 2: Nazan
Kayıt 1: netas444@co10ss.com

Gelen SIP mesajlarını almak için MobileSIPClient' in dinlemekte olduğu
lokal UDP PORTU: 53685

Bu konsol çıktısının ilk kısmında, bu projede kullanılan, Sun WTK 2.5.2 emülatörü ile birlikte gelen; ses, görüntü, video ve çoklu ortam desteği sunarak J2ME platformunu genişleten MMAPI paketi ilgili bilgiler verilmektedir. İlerki işlevlerden “Arama başlat” menüsünde, iki istemci arasındaki görüşme esnasında MMAPI tarafından desteklenen ses kodeklerinden encoding=pcm&rate=8000&bits=8&channels=1, ses dosya formatı olarak audio/x-wav, ve MMAPI Kayıt Özelliği kullanılacaktır.

Konsol çıktısının ikinci kısmında,

netas444	: İstemcinin Kullanıcı Adı
co10ss.com	: İstemcinin Alan Adı
netas444@co10ss.com	: İstemcinin SIP Adresi
Nazan	: İstemcinin Takma Adı
47.168.94.32	: İstemcinin IP Adresi
47.168.16.129	: Sunucunun IP Adresi
5060	: Sunucunun SIP Portu

SIP istemcisi, SIP sunucusunda co10ss.com alanında netas444 kullanıcı adıyla kayıtlıdır. netas444 kullanıcısına ait telefon numarası 2165420444 olarak veritabanına daha önceden kayıt edilmiştir. Dolayısıyla, netas444@co10ss.com SIP adresi yerine 2165420444@co10ss.com SIP adresi de kullanılabilir. İki SIP adresi de tek bir kullanıcıya aittir. Bu projenin tüm uygulamalarında, tasarlanan SIP istemcinin adresi,

netas444@co10ss.com olacaktır. 5060 standart SIP portudur. Proje derlenip çalıştırıldığında, Şekil 7.1’deki ilk ekran görüntüsünde olduğu gibi, “Uygulama, ağı kullanarak veri alıp göndermek istiyor. Bu ücretlendirilebilir.” şeklindeki uyarı mesajı ekrana gelir. İstemcinin her türlü SIP mesajını alıp gönderebilmesi için sunucunun IP adresine 5060 portuyla bir bağlantı açması gerekmektedir. Bu uyarı mesajının gelmesinin sebebi, istemcinin sunucunun IP adresine bir bağlantı açmak istemesinden kaynaklanmaktadır. Ayrıca gelen SIP mesajlarını alabilmek için istemcinin sürekli dinlemekte olduğu bir porta ihtiyaç vardır. Bu da, genellikle istemcinin lokal portudur (53685).

SIP istemcisi, çevrimdışı ve çevrimiçi olmak üzere iki durumda olabilir. SIP sunucusuna bağlı ise çevrimiçi, değil ise çevrimdışıdır. Bu yapı, günümüzde var olan ve yaygın olarak kullanılan Messenger programlarındaki gibi çalışır. SIP istemcisinin çevrimiçi veya çevrimdışı olmasına bağlı olarak gerçekleştirebileceği işlevler, işleyiş açısından farklılık göstermektedir.

7.1.1 Çevrimdışı Durumunda İken

7.1.1.1 Oturum Açma

Oturum açmak demek, istemcinin sunucuya, IP adresi ile kayıt olması diğer bir deyişle giriş yapması demektir. Giriş yaptıktan sonra, istemci çevrimiçi olur. Sunucuya giriş yapmak için istemci, SIP adres bilgisi ile (netas444@co10ss.com), sunucuya SIP talep mesajlarından REGISTER mesajını gönderir. Sunucudan, istemciyi bilgilendirmek amacıyla talebin alınıp, değerlendirildiğine dair 100 Trying cevap mesajı gelir. Ardından sunucu, istemciye 200 OK cevap mesajı göndererek, kaydın başarılı bir şekilde gerçekleştirildiğini bildirir.

İstemcinin arkadaş kişi listesinde, arkadaşları mevcut ise sunucuya, SUBSCRIBE mesajı göndererek arkadaş bilgilerinin güncellenmesini talep eder. Bu projede istemcinin (netas444@co10ss.com), arkadaş listesinde halihazırda 4 tane arkadaşı vardır:

- netas555@co10ss.com
- netas666@co10ss.com
- netas777@co10ss.com
- netas888@co10ss.com

İstemci, her bir arkadaşının durum bilgisini öğrenmek ve güncellemek için, sunucuya mesaj başlık kısmında Event başlığı, “presence” olarak seçilmiş 4 ayrı SUBSCRIBE mesajı gönderir. Sunucudan istemciye SUBSCRIBE mesajının alınıp, değerlendirildiğine ve başarıyla işlendiğine dair 100 Tying ve 200 OK mesajları gelir. Ardından, istemciye arkadaşlarının durum bilgisini içeren NOTIFY mesajları gönderilir. Oturum açma uygulamasına ait istemci ile sunucu arasında gelen giden SIP mesajları aşağıda listelenmiştir. Bu NOTIFY mesajları incelendiğinde, istemcinin arkadaşlarının durumu aşağıdaki gibidir.

- netas555@co10ss.com (ÇEVİRİMİÇİ)
- netas666@co10ss.com (ÇEVİRİMDIŞI)
- netas777@co10ss.com (YEMEKTE)
- netas888@co10ss.com (MEŞGUL)

Oturum açma uygulamasına ait emülatör çıktısı Şekil 7.2’ de gösterilmiştir. “Ana Menü” den “Oturum aç” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Oturum aç” menüsünü ekrana getirmektir. “Oturum aç” menüsünde tanımlanan “Geri” tuşu, ana menüye geri dönüşü sağlarken, “Oturum aç” tuşu yukarıda anlatılan gerekli SIP mesajlarını sunucuya göndererek oturum açma işlemini gerçekleştirir. Bu işlemler sırasında, istemciyi bilgilendirmek için “Oturum açılıyor...” bilgisi ekrana basılır. Oturum açma işlemi gerçekleştirildikten sonra da, “Oturum başarıyla açıldı.” bilgisi ekrana basılır ve işlem başarıyla sonlanır. “Geri” tuşuna basılarak “Ana Menü” ye geri dönülebilir.



Şekil 7.2 Çevrimdışı durumunda oturum açma emülatör çıktısı

Şekil 7.3' te oturum açma uygulamasına ait istemci ile sunucu arasındaki SIP mesajları akış şeması görülmektedir. Bu şemadaki 47.168.94.20 istemcinin IP adresi, 47.168.16.129 ise

sunucunun IP adresidir. 53685, istemcinin gelen SIP mesajlarını alabilmek için dinlediği lokal UDP portudur. 5060 ise, sunucunun gelen SIP mesajlarını alabilmesi için dinlediği standart SIP portudur.

Time	47.168.94.20	47.168.16.129	Comment
0,000	(53685)	Request: REGISTER sip:47.168.16.129:5060	SIP: Request: REGISTER sip:47.168.16.129:5060
0,033	(53685)	Status: 100 Trying (0 bindings)	SIP: Status: 100 Trying (0 bindings)
0,059	(53685)	Status: 200 Registr	SIP: Status: 200 Registration Successful (8 bindings)
1,008	(53685)	Request: SUBSCRIBE sip:netas555@co10ss.com	SIP: Request: SUBSCRIBE sip:netas555@co10ss.com
1,045	(53685)	Status: 100 Trying	SIP: Status: 100 Trying
1,052	(53685)	Status: 200 OK	SIP: Status: 200 OK
1,061	(53685)	Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20
1,566	(53685)	Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20
2,016	(53685)	Request: SUBSCRIBE sip:netas666@co10ss.com	SIP: Request: SUBSCRIBE sip:netas666@co10ss.com
2,052	(53685)	Status: 100 Trying	SIP: Status: 100 Trying
2,060	(53685)	Status: 200 OK	SIP: Status: 200 OK
2,070	(53685)	Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20
2,576	(53685)	Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20
2,586	(53685)	Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20
3,021	(53685)	Request: SUBSCRIBE sip:netas777@co10ss.com	SIP: Request: SUBSCRIBE sip:netas777@co10ss.com
3,057	(53685)	Status: 100 Trying	SIP: Status: 100 Trying
3,066	(53685)	Status: 200 OK	SIP: Status: 200 OK
3,076	(53685)	Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20
3,577	(53685)	Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20
3,587	(53685)	Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20
4,025	(53685)	Request: SUBSCRIBE sip:netas888@co10ss.com	SIP: Request: SUBSCRIBE sip:netas888@co10ss.com
4,062	(53685)	Status: 100 Trying	SIP: Status: 100 Trying
4,069	(53685)	Status: 200 OK	SIP: Status: 200 OK
4,079	(53685)	Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20

Şekil 7.3 Oturum açma SIP mesajları akış şeması

Bu akış şemasındaki SIP mesaj içerikleri ve konsol çıktısı aşağıdaki gibidir. GÖNDERİLEN MESAJ lar, istemciden sunucuya giden mesajlar, ALINAN MESAJ lar ise, sunucudan istemciye gelen mesajlardır.

GÖNDERİLEN MESAJ - Sun Aug 24 12:21:38 UTC 2008

```
REGISTER sip:47.168.16.129:5060 SIP/2.0
from: <sip:netas444@co10ss.com>;tag=1497588170
to: <sip:netas444@co10ss.com>
cseq: 1 REGISTER
call-id: f8d4e881acef618ab74c1ca6604a309c@47.168.94.32
via: SIP/2.0/UDP
47.168.94.32:53685;branch=z9hG4bKb4561f049d257b45ee958e0fe741cc46
contact: <sip:netas444@47.168.94.32:53685>
```

max-forwards: 70
content-length: 0

Listedeki Kişi Sayısı: 4

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SIP/2.0 100 Trying
From: <sip:netas444@co10ss.com>;tag=1497588170
To: <sip:netas444@co10ss.com>
Call-ID: f8d4e881acef618ab74c1ca6604a309c@47.168.94.32
CSeq: 1 REGISTER
Via: SIP/2.0/UDP
47.168.94.32:53685;received=47.168.94.20;branch=z9hG4bKb4561f049d257b45ee958e0fe741cc46
Content-Length: 0

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SIP/2.0 200 Registration Successful
From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=1497588170
To: <sip:netas444@co10ss.com>;tag=837647886
Call-ID: f8d4e881acef618ab74c1ca6604a309c@47.168.94.32
CSeq: 1 REGISTER
Via: SIP/2.0/UDP
47.168.94.32:53685;received=47.168.94.20;branch=z9hG4bKb4561f049d257b45ee958e0fe741cc46
Contact:
<sip:netas444@47.168.94.32:50010>;expires=86400,<sip:netas444@47.168.94.32:57516>;expires=86400,<sip:netas444@47.168.94.32:59296>;expires=86400,<sip:netas444@47.168.94.32:54706>;expires=86400,<sip:netas444@47.168.94.32:64159>;expires=86400,<sip:netas444@47.168.94.32:64163>;expires=86400,<sip:netas444@47.168.94.32:49961>;expires=3573,<sip:netas444@47.168.94.32:53685>;expires=3261
Content-Length: 0

GÖNDERİLEN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SUBSCRIBE sip:netas555@co10ss.com SIP/2.0
from: <sip:netas444@co10ss.com>;tag=5a20-acda
to: <sip:netas555@co10ss.com>
cseq: 1 SUBSCRIBE
call-id: 1db5-5a20-acda-b64438e3@47.168.94.32
via: SIP/2.0/UDP 47.168.94.32:53685;branch=b64438e3
Expires: 86400
contact: <sip:netas444@47.168.94.32:53685>
max-forwards: 70
Supported: https
Event: presence
Accept: application/cpim-pidf+xml
1: 0

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SIP/2.0 100 Trying
From: <sip:netas444@co10ss.com>;tag=5a20-acda
To: <sip:netas555@co10ss.com>

Call-ID: 1db5-5a20-acda-b64438e3@47.168.94.32
 CSeq: 1 SUBSCRIBE
 Via: SIP/2.0/UDP 47.168.94.32:53685;received=47.168.94.20;branch=b64438e3
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SIP/2.0 200 OK
 From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=5a20-acda
 To: <sip:netas555@co10ss.com>;tag=1219575763451
 Call-ID: 1db5-5a20-acda-b64438e3@47.168.94.32
 CSeq: 1 SUBSCRIBE
 Via: SIP/2.0/UDP 47.168.94.32:53685;received=47.168.94.20;branch=b64438e3
 Contact: <sip:netas555@co10ss.com:5060;maddr=47.168.16.129>
 Expires: 83575
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
 From: "2165420555"<sip:2165420555@co10ss.com>;tag=1219575763451
 To: "netas444"<sip:netas444@co10ss.com>;tag=5a20-acda
 Call-ID: 1db5-5a20-acda-b64438e3@47.168.94.32
 CSeq: 44 NOTIFY
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b24a2-6a071978-1dc52360
 Content-Type: application/cpim-pidf+xml
 Contact: <sip:2165420555@co10ss.com:5060;maddr=47.168.16.129>
 Max-Forwards: 20
 Subscription-Expires: 83576
 Event: presence
 Subscription-State: active;expires=83576
 Content-Length: 194

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420555@co10ss.com"/>
<t>s2:</t>
<t>p2:</t>
<t>w2:</t>
<t>i2:</t>
<t>d2:</t>
<t>u2:</t>
</presence>
```

2165420555 --> (ÇEVİRİMİÇİ)

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
 From: "2165420555"<sip:2165420555@co10ss.com>;tag=1219575763451
 To: "netas444"<sip:netas444@co10ss.com>;tag=5a20-acda
 Call-ID: 1db5-5a20-acda-b64438e3@47.168.94.32
 CSeq: 44 NOTIFY
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b24a2-6a071978-1dc52360
 Content-Type: application/cpim-pidf+xml
 Contact: <sip:2165420555@co10ss.com:5060;maddr=47.168.16.129>
 Max-Forwards: 20
 Subscription-Expires: 83576
 Event: presence
 Subscription-State: active;expires=83576

Content-Length: 194

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420555@co10ss.com"/>
<t>s2:</t>
<t>p2:</t>
<t>w2:</t>
<t>i2:</t>
<t>d2:</t>
<t>u2:</t>
</presence>
```

2165420555 --> (ÇEVİRİMİÇİ)

GÖNDERİLEN MESAJ - Sun Aug 24 12:21:38 UTC 2008

```
SUBSCRIBE sip:netas666@co10ss.com SIP/2.0
from: <sip:netas444@co10ss.com>;tag=9aa8-9e43
to: <sip:netas666@co10ss.com>
cseq: 1 SUBSCRIBE
call-id: 3990-9aa8-9e43-361f6d00@47.168.94.32
via: SIP/2.0/UDP 47.168.94.32:53685;branch=361f6d00
Expires: 86400
contact: <sip:netas444@47.168.94.32:53685>
max-forwards: 70
Supported: https
Event: presence
Accept: application/cpim-pidf+xml
l: 0
```

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

```
SIP/2.0 100 Trying
From: <sip:netas444@co10ss.com>;tag=9aa8-9e43
To: <sip:netas666@co10ss.com>
Call-ID: 3990-9aa8-9e43-361f6d00@47.168.94.32
CSeq: 1 SUBSCRIBE
Via: SIP/2.0/UDP 47.168.94.32:53685;received=47.168.94.20;branch=361f6d00
Content-Length: 0
```

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

```
SIP/2.0 200 OK
From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=9aa8-9e43
To: <sip:netas666@co10ss.com>;tag=1219575763452
Call-ID: 3990-9aa8-9e43-361f6d00@47.168.94.32
CSeq: 1 SUBSCRIBE
Via: SIP/2.0/UDP 47.168.94.32:53685;received=47.168.94.20;branch=361f6d00
Contact: <sip:netas666@co10ss.com:5060;maddr=47.168.16.129>
Expires: 83018
Content-Length: 0
```

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

```
NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
From: "2165420666"<sip:2165420666@co10ss.com>;tag=1219575763452
To: "netas444"<sip:netas444@co10ss.com>;tag=9aa8-9e43
Call-ID: 3990-9aa8-9e43-361f6d00@47.168.94.32
```

CSeq: 45 NOTIFY
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b24a3-6a071d68-65f25d78
 Content-Type: application/cpim-pidf+xml
 Contact: <sip:2165420666@co10ss.com:5060;maddr=47.168.16.129>
 Max-Forwards: 20
 Subscription-Expires: 83019
 Event: presence
 Subscription-State: active;expires=83019
 Content-Length: 236

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420666@co10ss.com"/>
<t>s7:Offline</t>
<t>p7:Offline</t>
<t>w7:Offline</t>
<t>i7:Offline</t>
<t>d7:Offline</t>
<t>u7:Offline</t>
</presence>
```

2165420666 --> (ÇEVİRİMDİŞİ)

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
 From: "2165420555"<sip:2165420555@co10ss.com>;tag=1219575763451
 To: "netas444"<sip:netas444@co10ss.com>;tag=5a20-acda
 Call-ID: 1db5-5a20-acda-b64438e3@47.168.94.32
 CSeq: 44 NOTIFY
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b24a2-6a071978-1dc52360
 Content-Type: application/cpim-pidf+xml
 Contact: <sip:2165420555@co10ss.com:5060;maddr=47.168.16.129>
 Max-Forwards: 20
 Subscription-Expires: 83576
 Event: presence
 Subscription-State: active;expires=83576
 Content-Length: 194

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420555@co10ss.com"/>
<t>s2:</t>
<t>p2:</t>
<t>w2:</t>
<t>i2:</t>
<t>d2:</t>
<t>u2:</t>
</presence>
```

2165420555 --> (ÇEVİRİMİÇİ)

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
 From: "2165420666"<sip:2165420666@co10ss.com>;tag=1219575763452
 To: "netas444"<sip:netas444@co10ss.com>;tag=9aa8-9e43
 Call-ID: 3990-9aa8-9e43-361f6d00@47.168.94.32
 CSeq: 45 NOTIFY
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b24a3-6a071d68-65f25d78
 Content-Type: application/cpim-pidf+xml

Contact: <sip:2165420666@co10ss.com:5060;maddr=47.168.16.129>
 Max-Forwards: 20
 Subscription-Expires: 83019
 Event: presence
 Subscription-State: active;expires=83019
 Content-Length: 236

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420666@co10ss.com"/>
<t>s7:Offline</t>
<t>p7:Offline</t>
<t>w7:Offline</t>
<t>i7:Offline</t>
<t>d7:Offline</t>
<t>u7:Offline</t>
</presence>
```

2165420666 --> (ÇEVİRİMDİŞİ)

GÖNDERİLEN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SUBSCRIBE sip:netas777@co10ss.com SIP/2.0
 from: <sip:netas444@co10ss.com>;tag=a41f-cf32
 to: <sip:netas777@co10ss.com>
 cseq: 1 SUBSCRIBE
 call-id: 26f3-a41f-cf32-f63de748@47.168.94.32
 via: SIP/2.0/UDP 47.168.94.32:53685;branch=f63de748
 Expires: 86400
 contact: <sip:netas444@47.168.94.32:53685>
 max-forwards: 70
 Supported: https
 Event: presence
 Accept: application/cpim-pidf+xml
 l: 0

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SIP/2.0 100 Trying
 From: <sip:netas444@co10ss.com>;tag=a41f-cf32
 To: <sip:netas777@co10ss.com>
 Call-ID: 26f3-a41f-cf32-f63de748@47.168.94.32
 CSeq: 1 SUBSCRIBE
 Via: SIP/2.0/UDP 47.168.94.32:53685;received=47.168.94.20;branch=f63de748
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SIP/2.0 200 OK
 From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=a41f-cf32
 To: <sip:netas777@co10ss.com>;tag=1219575763453
 Call-ID: 26f3-a41f-cf32-f63de748@47.168.94.32
 CSeq: 1 SUBSCRIBE
 Via: SIP/2.0/UDP 47.168.94.32:53685;received=47.168.94.20;branch=f63de748
 Contact: <sip:netas777@co10ss.com:5060;maddr=47.168.16.129>
 Expires: 78394
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
 From: "2165420777"<sip:2165420777@co10ss.com>;tag=1219575763453
 To: "netas444"<sip:netas444@co10ss.com>;tag=a41f-cf32
 Call-ID: 26f3-a41f-cf32-f63de748@47.168.94.32
 CSeq: 46 NOTIFY
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b24a4-6a072156-406bc841
 Content-Type: application/cpim-pidf+xml
 Contact: <sip:2165420777@co10ss.com:5060;maddr=47.168.16.129>
 Max-Forwards: 20
 Subscription-Expires: 78394
 Event: presence
 Subscription-State: active;expires=78394
 Content-Length: 266

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420777@co10ss.com"/>
<t>s3:Out+to+Lunch</t>
<t>p3:Out+to+Lunch</t>
<t>w3:Out+to+Lunch</t>
<t>i3:Out+to+Lunch</t>
<t>d3:Out+to+Lunch</t>
<t>u3:Out+to+Lunch</t>
</presence>
```

2165420777 --> (YEMEKTE)

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
 From: "2165420777"<sip:2165420777@co10ss.com>;tag=1219575763453
 To: "netas444"<sip:netas444@co10ss.com>;tag=a41f-cf32
 Call-ID: 26f3-a41f-cf32-f63de748@47.168.94.32
 CSeq: 46 NOTIFY
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b24a4-6a072156-406bc841
 Content-Type: application/cpim-pidf+xml
 Contact: <sip:2165420777@co10ss.com:5060;maddr=47.168.16.129>
 Max-Forwards: 20
 Subscription-Expires: 78394
 Event: presence
 Subscription-State: active;expires=78394
 Content-Length: 266

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420777@co10ss.com"/>
<t>s3:Out+to+Lunch</t>
<t>p3:Out+to+Lunch</t>
<t>w3:Out+to+Lunch</t>
<t>i3:Out+to+Lunch</t>
<t>d3:Out+to+Lunch</t>
<t>u3:Out+to+Lunch</t>
</presence>
```

2165420777 --> (YEMEKTE)

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
 From: "2165420666"<sip:2165420666@co10ss.com>;tag=1219575763452

To: "netas444"<sip:netas444@co10ss.com>;tag=9aa8-9e43
 Call-ID: 3990-9aa8-9e43-361f6d00@47.168.94.32
 CSeq: 45 NOTIFY
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b24a3-6a071d68-65f25d78
 Content-Type: application/cpim-pidf+xml
 Contact: <sip:2165420666@co10ss.com:5060;maddr=47.168.16.129>
 Max-Forwards: 20
 Subscription-Expires: 83019
 Event: presence
 Subscription-State: active;expires=83019
 Content-Length: 236

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420666@co10ss.com"/>
<t>s7:Offline</t>
<t>p7:Offline</t>
<t>w7:Offline</t>
<t>i7:Offline</t>
<t>d7:Offline</t>
<t>u7:Offline</t>
</presence>
```

2165420666 --> (ÇEVİRİMDİŞİ)

GÖNDERİLEN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SUBSCRIBE sip:netas888@co10ss.com SIP/2.0
 from: <sip:netas444@co10ss.com>;tag=673c-e770
 to: <sip:netas888@co10ss.com>
 cseq: 1 SUBSCRIBE
 call-id: ba9b-673c-e770-40931995@47.168.94.32
 via: SIP/2.0/UDP 47.168.94.32:53685;branch=40931995
 Expires: 86400
 contact: <sip:netas444@47.168.94.32:53685>
 max-forwards: 70
 Supported: https
 Event: presence
 Accept: application/cpim-pidf+xml
 l: 0

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SIP/2.0 100 Trying
 From: <sip:netas444@co10ss.com>;tag=673c-e770
 To: <sip:netas888@co10ss.com>
 Call-ID: ba9b-673c-e770-40931995@47.168.94.32
 CSeq: 1 SUBSCRIBE
 Via: SIP/2.0/UDP 47.168.94.32:53685;received=47.168.94.20;branch=40931995
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SIP/2.0 200 OK
 From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=673c-e770
 To: <sip:netas888@co10ss.com>;tag=1219575763454
 Call-ID: ba9b-673c-e770-40931995@47.168.94.32
 CSeq: 1 SUBSCRIBE
 Via: SIP/2.0/UDP 47.168.94.32:53685;received=47.168.94.20;branch=40931995
 Contact: <sip:netas888@co10ss.com:5060;maddr=47.168.16.129>

Expires: 77906
Content-Length: 0

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
From: "2165420888"<sip:2165420888@co10ss.com>;tag=1219575763454
To: "netas444"<sip:netas444@co10ss.com>;tag=673c-e770
Call-ID: ba9b-673c-e770-40931995@47.168.94.32
CSeq: 47 NOTIFY
Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b24a5-6a072542-38589334
Content-Type: application/cpim-pidf+xml
Contact: <sip:2165420888@co10ss.com:5060;maddr=47.168.16.129>
Max-Forwards: 20
Subscription-Expires: 77907
Event: presence
Subscription-State: active;expires=77907
Content-Length: 218

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420888@co10ss.com"/>
<t>s7:Busy</t>
<t>p7:Busy</t>
<t>w7:Busy</t>
<t>i7:Busy</t>
<t>d7:Busy</t>
<t>u7:Busy</t>
</presence>
```

2165420888 --> (MEŞGUL)

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
From: "2165420888"<sip:2165420888@co10ss.com>;tag=1219575763454
To: "netas444"<sip:netas444@co10ss.com>;tag=673c-e770
Call-ID: ba9b-673c-e770-40931995@47.168.94.32
CSeq: 47 NOTIFY
Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b24a5-6a072542-38589334
Content-Type: application/cpim-pidf+xml
Contact: <sip:2165420888@co10ss.com:5060;maddr=47.168.16.129>
Max-Forwards: 20
Subscription-Expires: 77907
Event: presence
Subscription-State: active;expires=77907
Content-Length: 218

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420888@co10ss.com"/>
<t>s7:Busy</t>
<t>p7:Busy</t>
<t>w7:Busy</t>
<t>i7:Busy</t>
<t>d7:Busy</t>
<t>u7:Busy</t>
</presence>
```

2165420888 --> (MEŞGUL)

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
 From: "2165420777"<sip:2165420777@co10ss.com>;tag=1219575763453
 To: "netas444"<sip:netas444@co10ss.com>;tag=a41f-cf32
 Call-ID: 26f3-a41f-cf32-f63de748@47.168.94.32
 CSeq: 46 NOTIFY
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b24a4-6a072156-406bc841
 Content-Type: application/cpim-pidf+xml
 Contact: <sip:2165420777@co10ss.com:5060;maddr=47.168.16.129>
 Max-Forwards: 20
 Subscription-Expires: 78394
 Event: presence
 Subscription-State: active;expires=78394
 Content-Length: 266

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420777@co10ss.com"/>
<t>s3:Out+to+Lunch</t>
<t>p3:Out+to+Lunch</t>
<t>w3:Out+to+Lunch</t>
<t>i3:Out+to+Lunch</t>
<t>d3:Out+to+Lunch</t>
<t>u3:Out+to+Lunch</t>
</presence>
```

2165420777 --> (YEMEKTE)

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
 From: "2165420888"<sip:2165420888@co10ss.com>;tag=1219575763454
 To: "netas444"<sip:netas444@co10ss.com>;tag=673c-e770
 Call-ID: ba9b-673c-e770-40931995@47.168.94.32
 CSeq: 47 NOTIFY
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b24a5-6a072542-38589334
 Content-Type: application/cpim-pidf+xml
 Contact: <sip:2165420888@co10ss.com:5060;maddr=47.168.16.129>
 Max-Forwards: 20
 Subscription-Expires: 77907
 Event: presence
 Subscription-State: active;expires=77907
 Content-Length: 218

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420888@co10ss.com"/>
<t>s7:Busy</t>
<t>p7:Busy</t>
<t>w7:Busy</t>
<t>i7:Busy</t>
<t>d7:Busy</t>
<t>u7:Busy</t>
</presence>
```

2165420888 --> (MEŞGUL)

7.1.1.2 Oturum Kapatma

İstemci çevrimdışı durumdaki iken, oturum zaten kapalıdır, diğer bir deyişle, sunucuya kayıt yapılmamıştır. Ancak açılmış bir oturum kapatılabilir. Dolayısıyla, istemcinin çevrimdışı durumunda iken, oturum kapatma işlemi engellenmiştir. İstemci çevrimdışı durumunda iken oturum kapatma uygulamasına ait emülatör çıktısı Şekil 7.4’ de gösterilmiştir.



Şekil 7.4 Çevrimdışı durumda oturum kapatma emülatör çıktısı

“Ana Menü” den “Oturum kapat” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Oturum kapat” menüsünü ekrana getirmektir. Ancak istemci çevrimdışı olduğu için, “Oturum zaten kapalı.” uyarısı ekrana basılarak “Ana Menü” ye geri dönülür.

7.1.1.3 Kişi Listesini Görüntüleme

İstemci çevrimdışı durumdaki iken, oturum kapalıdır. Dolayısıyla istemcinin, arkadaşlarının olduğu kişi listesini görmesi engellenmiştir. İstemci çevrimdışı durumunda iken kişi listesi uygulamasına ait emülatör çıktısı Şekil 7.5’ de gösterilmiştir.



Şekil 7.5 Çevrimdışı durumda kişi listesi emülatör çıktısı

“Ana Menü” den “Kişi listesi” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Kişi listesi” menüsünü ekrana getirmektir. Ancak istemci çevrimdışı olduğu için, “Kişi listesini görebilmek için oturum açmanız gerekmektedir.” uyarısı ekrana basılarak “Ana Menü” ye geri dönülür.

7.1.1.4 Kişi Ekleme

İstemci çevrimdışı durumdaki iken, oturum kapalıdır. Dolayısıyla istemcinin, kişi eklemesi engellenmiştir. İstemci çevrimdışı durumunda iken kişi ekle uygulamasına ait emülatör çıktısı Şekil 7.6’ da gösterilmiştir.



Şekil 7.6 Çevrimdışı durumda kişi ekle emülatör çıktısı

“Ana Menü” den “Kişi ekle” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Kişi ekle” menüsünü ekrana getirmektir. Ancak istemci çevrimdışı olduğu için, “Kişi ekleyebilmek için oturum açmanız gerekmektedir.” uyarısı ekrana basılarak “Ana Menü” ye geri dönülür.

7.1.1.5 Kişi Silme

İstemci çevrimdışı durumdaki iken, oturum kapalıdır. Dolayısıyla istemcinin, kişi silmesi engellenmiştir. İstemci çevrimdışı durumunda iken kişi sil uygulamasına ait emülatör çıktısı Şekil 7.7’ de gösterilmiştir.



Şekil 7.7 Çevrimdışı durumda kişi sil emülatör çıktısı

“Ana Menü” den “Kişi sil” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Kişi sil” menüsünü ekrana getirmektir. Ancak istemci çevrimdışı olduğu için, “Kişi silmek için oturum açmanız gerekmektedir.” uyarısı ekrana basılarak “Ana Menü” ye geri dönülür.

7.1.1.6 Anlık İleti Gönderme

İstemci çevrimdışı durumdaki iken, oturum kapalıdır. Dolayısıyla istemcinin, anlık ileti göndermesi engellenmiştir. İstemci çevrimdışı durumunda iken anlık ileti gönder uygulamasına ait emülatör çıktısı Şekil 7.8’ de gösterilmiştir.



Şekil 7.8 Çevrimdışı durumda anlık ileti gönder emülatör çıktısı

“Ana Menü” den “Anlık ileti gönder” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Anlık ileti gönder” menüsünü ekrana getirmektir. Ancak istemci çevrimdışı olduğu için, “Anlık ileti gönderebilmek için oturum açmanız gerekmektedir.” uyarısı ekrana basılarak “Ana Menü” ye geri dönülür.

7.1.1.7 Arama Başlatma

İstemci çevrimdışı durumdaki iken, oturum kapalıdır. Dolayısıyla istemcinin, arama başlatması engellenmiştir. İstemci çevrimdışı durumunda iken arama başlat uygulamasına ait emülatör çıktısı Şekil 7.9’ da gösterilmiştir.



Şekil 7.9 Çevrimdışı durumunda arama başlat emülatör çıktısı

“Ana Menü” den “Arama başlat” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Arama başlat” menüsünü ekrana getirmektir. Ancak istemci çevrimdışı olduğu için, “Arama başlatılabilmek için oturum açmanız gerekmektedir.” uyarısı ekrana basılarak “Ana Menü” ye geri dönülür.

7.1.1.8 Hesap Bilgilerini Düzenleme

İstemci çevrimdışı durumdaki iken, oturum kapalıdır. Dolayısıyla istemci, hesap bilgilerini düzenleyebilir. İstemci çevrimdışı durumunda iken hesap bilgilerini düzenle uygulamasına ait emülatör çıktısı Şekil 7.10’ da gösterilmiştir.



Şekil 7.10 Çevrimdışı durumunda hesap bilgilerini düzenle emülatör çıktısı

“Ana Menü” den “Hesap bilgilerini düzenle” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Hesap bilgilerini düzenle” menüsünü ekrana getirmektir. İstemciye ait hesap bilgileri aşağıdaki gibidir.

İstemcinin SIP Adresi	: netas444@co10ss.com
İstemcinin Takma Adı	: Nazan GONEN
Sunucunun IP Adresi	: 47.168.16.129
Sunucunun Port Numarası	: 5060
İstemcinin IP Adresi	: 47.168.94.32

Bu bilgiler için, “Hesap bilgilerini düzenle” menüsüne 5 adet metin kutusu eklenmiştir. Her bir metin kutusu değerlerini, Record Store adı verilen kayıt alanlarından almaktadır. Bu kayıt alanları ise, J2ME platformunun sunmuş olduğu RMS(Record Management System) paketi yardımıyla oluşturulmuşlardır. 5 adet metin kutusu için, metin kutusu değerlerini oluşturmak için 5 adet kayıt alanı yaratılmıştır. RMS sistemi kayıt bilgilerini, kullanılan aracın, bu projede Sun WTK 2.5.2 emülatörünün kalıcı hafızasına kaydetmektedir, yani kayıt oluşturmaktadır. Böylelikle program kapatılıp, tekrar açıldığında diğer bir deyişle, proje tekrar

derlenip çalıştırıldığında, hesap bilgileri silinmez, program kapatılmadan önceki en son kaydedilen hesap bilgileri korunmuş olur.

“Hesap bilgilerini düzenle” menüsünde, “Geri” ve “Kaydet” tuşları oluşturulmuş ve proramlanmıştır. “Geri” tuşunun görevi, “Ana Menü” ye geri dönmektir. “Kaydet” tuşunun görevi ise, hesap bilgilerinde yapılan değişiklikleri, kayıt alanlarında güncellemektir.

“Takma Adım” başlıklı metin kutusundaki “Nazan GONEN” kaydı, “Nazan” olarak değiştirilip, programlanmış “Kaydet” tuşuna basıldığında, takma adına ait kayıt güncellenecektir ve programın bir sonraki açılışında, “Takma Adım” “Nazan” olarak gelecektir.

Değişiklik yapılp, “Kaydet” tuşuna basıldığında, “Hesap değişiklikleriniz başarıyla kaydedildi.” uyarısı ekrana basılarak “Ana Menü” ye geri dönülür. Yapılan değişiklik sonucu, güncellenen hesap bilgileri konsola aşağıdaki gibi yazdırılır.

GÜNCELLENEN HESAP BİLGİLERİ

```
Kayıt 5 :47.168.94.32
Kayıt 4 :5060
Kayıt 3 :47.168.16.129
Kayıt 2 :Nazan
Kayıt 1 :netas444@co10ss.com
```

Hesap değişiklikleriniz başarıyla kaydedildi.

7.1.2 Çevrimiçi Durumunda İken

7.1.2.1 Oturum Açma

İstemci çevrimiçi durumdaki iken, oturum açıktır, diğer bir deyişle, istemci sunucuya daha önceden kayıt olmuştur. Dolayısıyla, istemcinin çevrimiçi durumunda iken, oturum açma işlemi engellenmiştir. İstemci çevrimiçi durumunda iken oturum açma uygulamasına ait emülatör çıktısı Şekil 7.11’ de gösterilmiştir.



Şekil 7.11 Çevrimiçi durumunda oturum açma emülatör çıktısı

“Ana Menü” den “Oturum aç” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Oturum aç” menüsünü ekrana getirmektir. Ancak istemci çevrimiçi olduğu için, “Oturum zaten açık.” uyarısı ekrana basılarak “Ana Menü” ye geri dönülür.

7.1.2.2 Oturum Kapatma

Oturum kapatmak demek, istemcinin sunucudan çıkış yapması demektir. Çıkış yaptıktan sonra, istemci çevrimdışı olur. SIP protokolünde, sunucuya kayıt olmak için kullanılan REGISTER mesajı gibi sunucudan çıkış yapmak için kullanılan bir UNREGISTER mesajı yoktur. Onun yerine REGISTER mesajı Expires: 0 başlık bilgisi ile gönderilir. Sunucudan çıkış yapmak için istemci, SIP adres bilgisi ile (netas444@co10ss.com), sunucuya Expires: 0 başlıklı bir REGISTER mesajını gönderir. Sunucudan, istemciyi bilgilendirmek amacıyla talebin alınıp, değerlendirildiğine dair 100 Trying cevap mesajı gelir. Ardından sunucu, istemciye 200 OK cevap mesajı göndererek, sunucudan çıkış işleminin başarılı bir şekilde gerçekleştirildiğini bildirir.

Oturum kapatma uygulamasına ait emülatör çıktısı Şekil 7.12’ de gösterilmiştir. “Ana Menü” den “Oturum kapat” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Oturum kapat” menüsünü ekrana getirmektir. “Oturum kapat” menüsünde tanımlanan “Geri” tuşu, ana menüye geri dönüşü sağlarken, “Oturum kapat” tuşu yukarıda anlatılan gerekli SIP mesajlarını sunucuya göndererek oturum kapatma işlemini gerçekleştirir. Bu işlemler sırasında, istemciyi bilgilendirmek için “Oturum kapatılıyor...” bilgisi ekrana basılır. Oturum kapatma işlemi gerçekleştirildikten sonra da, “Oturum başarıyla kapatıldı.” bilgisi ekrana basılır ve işlem başarıyla sonlanır. “Geri” tuşuna basılarak “Ana Menü” ye geri dönülebilir.



Şekil 7.12 Çevrimiçi durumda oturum kapatma emülatör çıktısı

Şekil 7.13' te oturum kapatma uygulamasına ait istemci ile sunucu arasındaki SIP mesajları akış şeması görülmektedir. Bu şemadaki 47.168.94.124 istemcinin IP adresi, 47.168.16.129

ise sunucunun IP adresidir. 61082, istemcinin gelen SIP mesajlarını alabilmek için dinlediği lokal UDP portudur. 5060 ise, sunucunun gelen SIP mesajlarını alabilmesi için dinlediği standart SIP portudur.

Time	47.168.94.124	47.168.16.129	Comment
0,000	Request: REGISTER s (61082)	(5060)	SIP: Request: REGISTER sip:47.168.16.129:5060
0,032	Status: 100 Trying (61082)	(5060)	SIP: Status: 100 Trying (0 bindings)
0,154	Status: 200 Registr (61082)	(5060)	SIP: Status: 200 Registration Successful (8 bindings)

Şekil 7.13 Çevrimiçi durumda kişi listesi emülatör çıktısı

Bu akış şemasındaki SIP mesaj içerikleri ve konsol çıktısı aşağıdaki gibidir. GÖNDERİLEN MESAJ lar, istemciden sunucuya giden mesajlar, ALINAN MESAJ lar ise, sunucudan istemciye gelen mesajlardır. Dikkat edilecek olursa, istemciden sunucuya gönderilen REGISTER mesajındaki Expires başlığı 0 olarak set edilmiştir.

GÖNDERİLEN MESAJ - Tue Aug 26 15:46:09 UTC 2008

```
REGISTER sip:47.168.16.129:5060 SIP/2.0
From: <sip:netas444@co10ss.com>;tag=1497588170
To: <sip:netas444@co10ss.com>
CSeq: 2 REGISTER
Call-ID: f8d4e881acef618ab74c1ca6604a309c@47.168.94.124
Via: SIP/2.0/UDP 47.168.94.124:61082;branch=31f3a068
Content-Length: 0
Contact: <sip:netas444@47.168.94.124:61082>;expires=0;description=LOGOUT
Expires: 0
Max-Forwards: 70
```

ALINAN MESAJ - Tue Aug 26 15:46:09 UTC 2008

```
SIP/2.0 100 Trying
From: <sip:netas444@co10ss.com>;tag=1497588170
To: <sip:netas444@co10ss.com>
Call-ID: f8d4e881acef618ab74c1ca6604a309c@47.168.94.124
CSeq: 2 REGISTER
Via: SIP/2.0/UDP 47.168.94.124:61082;branch=31f3a068
Content-Length: 0
```

ALINAN MESAJ - Tue Aug 26 15:46:09 UTC 2008

```
SIP/2.0 200 Registration Successful
From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=1497588170
To: <sip:netas444@co10ss.com>;tag=1066527344
Call-ID: f8d4e881acef618ab74c1ca6604a309c@47.168.94.124
CSeq: 2 REGISTER
Via: SIP/2.0/UDP 47.168.94.124:61082;branch=31f3a068
Contact:
<sip:netas444@47.168.94.14:61879>;expires=86400,<sip:netas444@47.168.94.14:
57422>;expires=86400,<sip:netas444@47.168.94.14:61866>;expires=86400,<sip:n
```

```

etas444@47.168.94.14:61871>;expires=86400,<sip:netas444@47.168.94.14:61875>
;expires=86400,<sip:netas444@47.168.94.14:59325>;expires=86400,<sip:netas44
4@47.168.94.14:51323>;expires=86400,<sip:netas444@47.168.94.14:57427>;expir
es=86400
Content-Length: 0

```

7.1.2.3 Kişi Listesini Görüntüleme

İstemci çevrimiçi olduğu sürece, arkadaşlarının bulunduğu kişi listesini görebilecektir. İstemci çevrimiçi durumunda iken kişi listesi uygulamasına ait emülatör çıktısı Şekil 7.14’ de gösterilmiştir.



Şekil 7.14 Çevrimiçi durumda kişi listesi emülatör çıktısı

“Ana Menü” den “Kişi listesi” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Kişi listesi” menüsünü ekrana getirmektir. Burada istemcinin (netas444@co10ss.com), halihazırda daha önceden eklenmiş arkadaşları, durum bilgileri ile birlikte listelenir. Daha önceden bahsedildiği gibi oturum açma uygulamasında, istemci sunucuya REGISTER olduktan sonra, arkadaş listesindeki kişilerin durum bilgilerini öğrenmek için her bir kişi için sunucuya SUBSCRIBE mesajı göndermişti. Sunucudan bu SUBSCRIBE taleplerine karşılık, kişilerin durum bilgilerini içeren NOTIFY mesajları

gelmişti. İşte kişi listesindeki bu durumlar, gelen NOTIFY mesajlarının içerisindeki durum bilgileri işlenerek oluşturulmuştur. Ekranda programlanan “Geri” tuşuna basılarak “Ana Menü” ye geri dönülebilir.

İstemcinin kişi listesindeki kullanıcılar ve durumları Şekil 7.15’ te gösterilmiştir.



Şekil 7.15 İstemcinin kişi listesindeki kullanıcılar

İstemci çevrimiçi olduğu sürece, listesindeki kullanıcılar, durumlarını değiştirebilirler. Kullanıcılar, durumlarını değiştirdiklerinde sunucuya PUBLISH mesajı göndererek, yeni durumlarını bildirirler. Sunucu, kullanıcıların yeni durumlarından haberdar olur ve kullanıcıların arkadaş olduğu tüm istemcilere, kullanıcıların yeni durumlarını NOTIFY mesajı ile bildirir. Bu projede de, istemcinin listesindeki kullanıcılar, durumlarını değiştirdiklerinde sunucudan istemciye, kullanıcının durumunu değiştirdiğine dair bir NOTIFY gönderilir.

Şekil 7.16’ da görüldüğü gibi netas555 kişisi, durumunu ÇEVİRİMİÇİ’nden DIŞARIDA olarak değiştirmiştir.



Şekil 7.16 Kullanıcı durum değişikliği (Çevrimiçi --> Dışarıda)

Şekil 7.17' de ise netas555 kişinin yaptığı durum değişikliğine ait istemci ile sunucu arasındaki SIP mesajları akış şeması görülmektedir. Bu şemadaki 47.168.94.20 istemcinin IP adresi, 47.168.16.129 ise sunucunun IP adresidir. 53685, istemcinin gelen SIP mesajlarını alabilmek için dinlediği lokal UDP portudur. 5060 ise, sunucunun gelen SIP mesajlarını alabilmesi için dinlediği standart SIP portudur.

Time	47.168.16.129	47.168.94.20	Comment
0,000	(5060)	(53685)	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20

Şekil 7.17 Kullanıcı durum değişikliği (Çevrimiçi --> Dışarıda) SIP mesajları akış şeması

Bu akış şemasındaki SIP mesaj içerikleri ve konsol çıktısı aşağıdaki gibidir. ALINAN MESAJ, sunucudan istemciye gelen mesajdır.

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

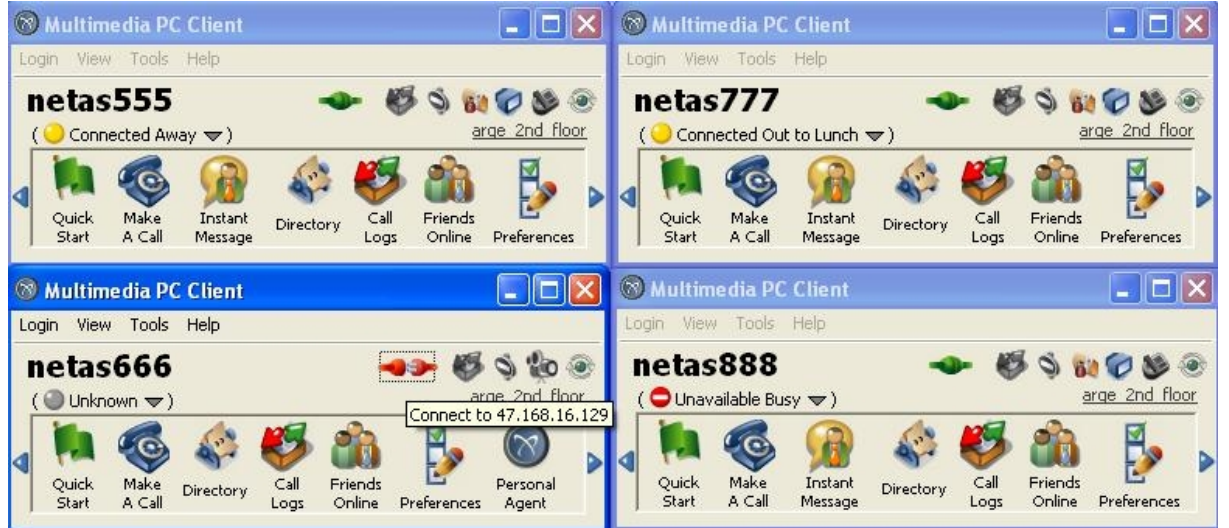
```
NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
From: "2165420555"<sip:2165420555@co10ss.com>;tag=1219575763451
To: "netas444"<sip:netas444@co10ss.com>;tag=5a20-acda
Call-ID: 1db5-5a20-acda-b64438e3@47.168.94.32
CSeq: 48 NOTIFY
Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b2b87-6a220763-7602b027
Content-Type: application/cpim-pidf+xml
Contact: <sip:2165420555@co10ss.com:5060;maddr=47.168.16.129>
Max-Forwards: 20
Subscription-Expires: 81811
Event: presence
Subscription-State: active;expires=81811
```

Content-Length: 218

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420555@co10ss.com"/>
<t>s3:Away</t>
<t>p3:Away</t>
<t>w3:Away</t>
<t>i3:Away</t>
<t>d3:Away</t>
<t>u3:Away</t>
</presence>
```

2165420555 --> (DIŞARIDA)

Şekil 7.18' de görüldüğü gibi netas666 kişisi, durumunu ÇEVİRİMDIŞI ndan ÇEVİRİMİÇİ olarak değiştirmiştir.



Şekil 7.18 Kullanıcı durum değişikliği (Çevrimdışı --> Çevrimiçi)

Şekil 7.19' da ise netas666 kişinin yaptığı durum değişikliğine ait istemci ile sunucu arasındaki SIP mesajları akış şeması görülmektedir. Bu şemadaki 47.168.94.20 istemcinin IP adresi, 47.168.16.129 ise sunucunun IP adresidir. 53685, istemcinin gelen SIP mesajlarını alabilmek için dinlediği lokal UDP portudur. 5060 ise, sunucunun gelen SIP mesajlarını alabilmesi için dinlediği standart SIP portudur.

Time	47.168.16.129	47.168.94.20	Comment
0,000	(5060)	(53685)	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20

Şekil 7.19 Kullanıcı durum değişikliği(Çevrimdışı-->Çevrimiçi) SIP mesajları akış şeması

Bu akış şemasındaki SIP mesaj içerikleri ve konsol çıktısı aşağıdaki gibidir. ALINAN MESAJ, sunucudan istemciye gelen mesajdır.

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

```
NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
From: "2165420666"<sip:2165420666@co10ss.com>;tag=1219575763452
To: "netas444"<sip:netas444@co10ss.com>;tag=9aa8-9e43
Call-ID: 3990-9aa8-9e43-361f6d00@47.168.94.32
CSeq: 49 NOTIFY
Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b2d43-6a28cf04-3557ef8b
Content-Type: application/cpim-pidf+xml
Contact: <sip:2165420666@co10ss.com:5060;maddr=47.168.16.129>
Max-Forwards: 20
Supported: com.nortelnetworks.firewall,p-3rdpartycontrol,nosec,join,x-
nortel-sipvc,com.nortelnetworks.im.encrypted
Subscription-Expires: 80810
Event: presence
Subscription-State: active;expires=80810
Content-Encoding: x-nortel-short
Content-Length: 194
```

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420666@co10ss.com"/>
<t>s2:</t>
<t>p2:</t>
<t>w2:</t>
<t>i2:</t>
<t>d2:</t>
<t>u2:</t>
</presence>
```

2165420666 --> (ÇEVİRİMİÇİ)

Şekil 7.20’ de görüldüğü gibi netas777 kişisi, durumunu YEMEKTE den DÖNECEK olarak değiştirmiştir.



Şekil 7.20 Kullanıcı durum değişikliği (Yemekte --> Dönecek)

Şekil 7.21' de ise netas777 kişinin yaptığı durum değişikliğine ait istemci ile sunucu arasındaki SIP mesajları akış şeması görülmektedir. Bu şemadaki 47.168.94.20 istemcinin IP adresi, 47.168.16.129 ise sunucunun IP adresidir. 53685, istemcinin gelen SIP mesajlarını alabilmek için dinlediği lokal UDP portudur. 5060 ise, sunucunun gelen SIP mesajlarını alabilmesi için dinlediği standart SIP portudur.

Time	47.168.16.129	47.168.94.20	Comment
0,000	(5060)	(53685)	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20

Şekil 7.21 Kullanıcı durum değişikliği (Yemekte --> Dönecek) SIP mesajları akış şeması

Bu akış şemasındaki SIP mesaj içerikleri ve konsol çıktısı aşağıdaki gibidir. ALINAN MESAJ, sunucudan istemciye gelen mesajdır.

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

```
NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
From: "2165420777"<sip:2165420777@co10ss.com>;tag=1219575763453
To: "netas444"<sip:netas444@co10ss.com>;tag=a41f-cf32
Call-ID: 26f3-a41f-cf32-f63de748@47.168.94.32
CSeq: 64 NOTIFY
Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b2edc-6a2f0bb3-4848e055
Content-Type: application/cpim-pidf+xml
Contact: <sip:2165420777@co10ss.com:5060;maddr=47.168.16.129>
Max-Forwards: 20
Supported: com.nortelnetworks.firewall,p-3rdpartycontrol,nosec,join,x-
nortel-sipvc,com.nortelnetworks.im.encryption
```

Subscription-Expires: 75778
 Event: presence
 Subscription-State: active;expires=75778
 Content-Encoding: x-nortel-short
 Content-Length: 272

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420777@co10ss.com"/>
<t>s3:Be+Right+Back</t>
<t>p3:Be+Right+Back</t>
<t>w3:Be+Right+Back</t>
<t>i3:Be+Right+Back</t>
<t>d3:Be+Right+Back</t>
<t>u3:Be+Right+Back</t>
</presence>
```

2165420777 --> (DÖNECEK)

Şekil 7.22' de görüldüğü gibi netas888 kişisi, durumunu MEŞGUL den ÇEVİRİMDİŞİ olarak değiştirmiştir.



Şekil 7.22 Kullanıcı durum değişikliği (Meşgul --> Çevrimdışı)

Şekil 7.23' te ise netas888 kişinin yaptığı durum değişikliğine ait istemci ile sunucu arasındaki SIP mesajları akış şeması görülmektedir. Bu şemadaki 47.168.94.20 istemcinin IP adresi, 47.168.16.129 ise sunucunun IP adresidir. 53685, istemcinin gelen SIP mesajlarını alabilmek için dinlediği lokal UDP portudur. 5060 ise, sunucunun gelen SIP mesajlarını alabilmesi için dinlediği standart SIP portudur.

Time	47.168.16.129	47.168.94.20	Comment
0,000	Request: NOTIFY sip: (5060)	(53685)	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20

Şekil 7.23 Kullanıcı durum değişikliği (Meşgul --> Çevrimdışı) SIP mesajları akış şeması

Bu akış şemasındaki SIP mesaj içerikleri ve konsol çıktısı aşağıdaki gibidir. ALINAN MESAJ, sunucudan istemciye gelen mesajdır.

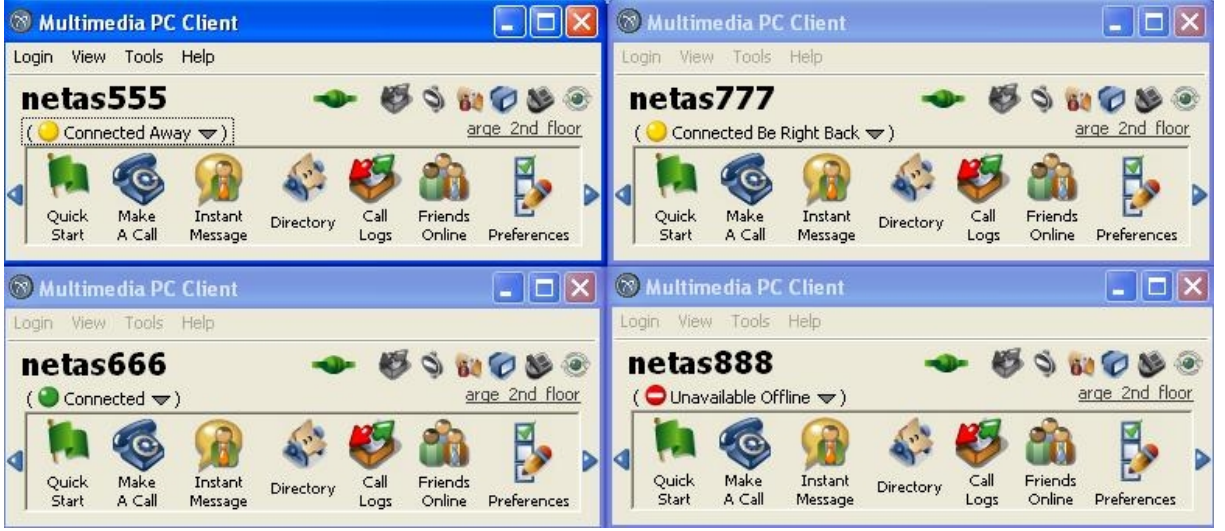
ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

```
NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
From: "2165420888"<sip:2165420888@co10ss.com>;tag=1219575763454
To: "netas444"<sip:netas444@co10ss.com>;tag=673c-e770
Call-ID: ba9b-673c-e770-40931995@47.168.94.32
CSeq: 79 NOTIFY
Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b3087-6a3591d8-5f87e7b1
Content-Type: application/cpim-pidf+xml
Contact: <sip:2165420888@co10ss.com:5060;maddr=47.168.16.129>
Max-Forwards: 20
Supported: com.nortelnetworks.firewall,p-3rdpartycontrol,nosec,join,x-
nortel-sipvc,com.nortelnetworks.im.encryption
Subscription-Expires: 74864
Event: presence
Subscription-State: active;expires=74864
Content-Encoding: x-nortel-short
Content-Length: 236
```

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420888@co10ss.com"/>
<t>s7:Offline</t>
<t>p7:Offline</t>
<t>w7:Offline</t>
<t>i7:Offline</t>
<t>d7:Offline</t>
<t>u7:Offline</t>
</presence>
```

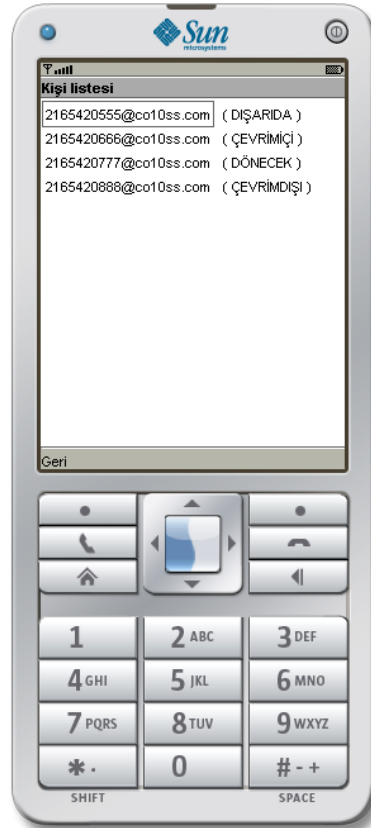
2165420888 --> (ÇEVİRİMDİŞİ)

İstemcinin kişi listesindeki kullanıcıların, durumlarını değiştirdikten sonraki hali Şekil 7.24'te gösterilmiştir.



Şekil 7.24 İstemcinin kişi listesindeki kullanıcıların durum bilgileri

İstemcinin kişi listesindeki kullanıcılar, durumlarını değiştirdiklerinde sunucudan istemciye, kullanıcıların durumlarını değiştirdiklerine dair gönderilen NOTIFY mesajları işlenerek, istemcinin kişi listesi, kullanıcıların yeni durumları ile Şekil 7.25’ teki gibi anında güncellenir.



Şekil 7.25 Kullanıcıların durum bilgileri güncellenmiş istemci kişi listesi

7.1.2.4 Kişi Ekleme

İstemci çevrimiçi olduğu sürece, kişi ekleyebilecektir. İstemci çevrimiçi durumunda iken kişi ekle uygulamasına ait emülatör çıktısı Şekil 7.26’ da gösterilmiştir.



Şekil 7.26 Çevrimiçi durumda kişi ekle emülatör çıktısı

“Ana Menü” den “Kişi ekle” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Kişi ekle” menüsünü ekrana getirmektir. “Kişi ekle” menüsünde, eklenecek kişinin SIP adresini girebilmek için “Eklenecek Kişinin SIP Adresi” başlıklı bir metin kutusu, kişi eklemekten vazgeçip “Ana Menü” ye dönmek için “Geri” tuşu ve metin kutusuna girilen kişiyi kişi listesine eklemek için “Kaydet” tuşu tanımlanmıştır.

Yukarıdaki örnek gösterimde, “Eklenecek Kişinin SIP Adresi” olarak netas999@co10ss.com girilerek “Kaydet” tuşuna basılmıştır. “Kaydet” tuşuna basıldığında, eklenen kişinin SIP adres bilgisi kullanılarak sunucuya bir SUBSCRIBE mesajı gönderilir. Sunucudan istemciye SUBSCRIBE mesajının alınıp, değerlendirildiğine ve başarıyla işlendiğine dair 100 Trying ve 200 OK mesajları gelir. Ardından, istemciye netas999@co10ss.com kişinin durum bilgisini içeren NOTIFY mesajı gönderilir. Kişi ekleme işlemi başarıyla gerçekleştirildikten sonra, “Kişi, listenize başarıyla eklendi.” bilgisi ekrana basılarak işlem sonlandırılır ve “Ana Menü” ye dönülür.

Şekil 7.27’ de ise kişi ekle uygulamasına ait istemci ile sunucu arasındaki SIP mesajları akış şeması görülmektedir. Bu şemadaki 47.168.94.20 istemcinin IP adresi, 47.168.16.129 ise sunucunun IP adresidir. 49607, istemcinin gelen SIP mesajlarını alabilmek için dinlediği lokal UDP portudur. 5060 ise, sunucunun gelen SIP mesajlarını alabilmesi için dinlediği standart SIP portudur.

Time	47.168.94.20	47.168.16.129	Comment
0,000	(49607) Request: SUBSCRIBE	(5060)	SIP: Request: SUBSCRIBE sip:netas999@co10ss.com
0,037	(49607) Status: 100 Trying	(5060)	SIP: Status: 100 Trying
0,050	(49607) Request: NOTIFY sip:	(5060)	SIP: Request: NOTIFY sip:netas444@47.168.94.32:49607;maddr=47.168.94.20
0,055	(49607) Status: 200 OK	(5060)	SIP: Status: 200 OK

Şekil 7.27 Kişi ekle SIP Mesajları akış şeması

Bu akış şemasındaki SIP mesaj içerikleri ve konsol çıktısı aşağıdaki gibidir. GÖNDERİLEN MESAJ lar, istemciden sunucuya giden mesajlar, ALINAN MESAJ lar ise, sunucudan istemciye gelen mesajlardır.

GÖNDERİLEN MESAJ - Sun Aug 24 14:03:12 UTC 2008

SUBSCRIBE sip:netas999@co10ss.com SIP/2.0

From: <sip:netas444@co10ss.com>;tag=0605-c159
 To: <sip:netas999@co10ss.com>
 Cseq: 1 SUBSCRIBE
 Call-ID: 3047-0605-c159-daad05a2@47.168.94.32
 Via: SIP/2.0/UDP 47.168.94.32:49607;branch=daad05a2
 Expires: 86400
 Contact: <sip:netas444@47.168.94.32:49607>
 Max-Forwards: 70
 Supported: https
 Event: presence
 Accept: application/cpim-pidf+xml
 l:0

Listenizdeki Kişi Sayısı: 5

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

SIP/2.0 100 Trying
 From: <sip:netas444@co10ss.com>;tag=0605-c159
 To: <sip:netas999@co10ss.com>
 Call-ID: 3047-0605-c159-daad05a2@47.168.94.32
 CSeq: 1 SUBSCRIBE
 Via: SIP/2.0/UDP 47.168.94.32:49607;received=47.168.94.20;branch=daad05a2
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

NOTIFY sip:netas444@47.168.94.32:49607;maddr=47.168.94.20 SIP/2.0
 From: "2165420999"<sip:2165420999@co10ss.com>;tag=1219575763491
 To: "netas444"<sip:netas444@co10ss.com>;tag=0605-c159
 Call-ID: 3047-0605-c159-daad05a2@47.168.94.32
 CSeq: 125 NOTIFY
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b3c5a-6a63c363-30107276
 Content-Type: application/cpim-pidf+xml
 Contact: <sip:2165420999@co10ss.com:5060;maddr=47.168.16.129>
 Max-Forwards: 20
 Subscription-Expires: 78279
 Event: presence
 Subscription-State: active;expires=78279
 Content-Length: 236

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420999@co10ss.com"/>
<t>s7:Offline</t>
<t>p7:Offline</t>
<t>w7:Offline</t>
<t>i7:Offline</t>
<t>d7:Offline</t>
<t>u7:Offline</t>
</presence>
```

2165420999 --> (ÇEVİRİMDİŞİ)

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

SIP/2.0 200 OK
 From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=0605-c159
 To: <sip:netas999@co10ss.com>;tag=1219575763491

Call-ID: 3047-0605-c159-daad05a2@47.168.94.32
CSeq: 1 SUBSCRIBE
Via: SIP/2.0/UDP 47.168.94.32:49607;received=47.168.94.20;branch=daad05a2
Contact: <sip:netas999@co10ss.com:5060;maddr=47.168.16.129>
Expires: 78278
Content-Length: 0

Kişi ekle uygulaması ile bir kişi eklendiğinde,

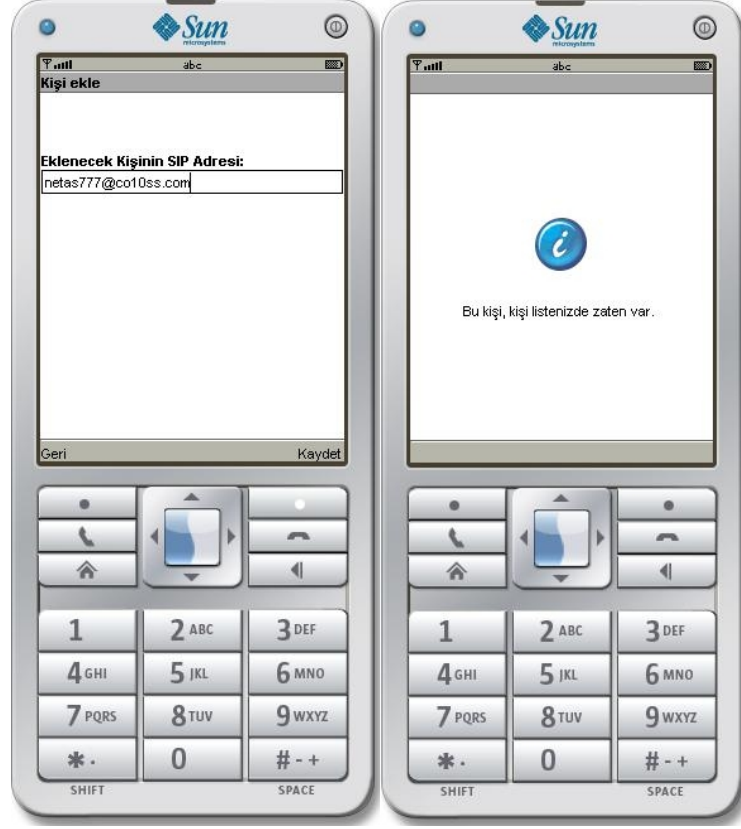
- Kişi listesi
- Kişi sil
- Anlık ileti gönder
- Arama başlat

menülerindeki kişi listeleri de Şekil 7.28' de olduğu gibi güncellenir.



Şekil 7.28 Kişi eklendikten sonra güncellenen kişi listeleri emülatör çıktısı

Kişi listesinde var olan bir kullanıcıyı tekrar eklemek engellenmiştir. Aşağıdaki şekilde, istemcinin kişi listesinde var olan netas777@co10ss.com kullanıcısı eklenmeye çalışıldığında ekrana “Bu kişi, kişi listenizde zaten var.” uyarı mesajı basılır, “Ana Menü” ye geri dönülür.



Şekil 7.29 Kişi listesinde mevcut bir kullanıcıyı tekrar ekleme emülatör uyarı çıktısı

7.1.2.5 Kişi Silme

İstemci çevrimiçi olduğu sürece, kişi sililebilecektir. İstemci çevrimiçi durumunda iken kişi sil uygulamasına ait emülatör çıktısı Şekil 7.30’ da gösterilmiştir.



Şekil 7.30 Çevrimiçi durumda kişi sil emülatör çıktısı

“Ana Menü” den “Kişi sil” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Kişi sil” menüsünü ekrana getirmektir. “Kişi sil” menüsünde, silinecek kişiyi

seçmek için elemanları kişiler olan bir seçenek listesi, kişi silmekten vazgeçip “Ana Menü” ye dönmek için “Geri” tuşu ve seçilen kişinin silme işlemini gerçekleştirmek için “Sil” tuşu tanımlanmıştır.

SIP protokolünde, kullanıcı arabirim tarafından özel olaylar, etkinlikler için bildirimlerin iletildiği abonelik gerektiren oturumlar kurmak için kullanılan SUBSCRIBE mesajı gibi abonelik sonlandırmak için kullanılan bir UNSUBSCRIBE mesajı yoktur. Onun yerine SUBSCRIBE mesajı Expires: 0 başlık bilgisi ile gönderilir. Yukarıdaki örnek gösterimde, netas777@co10ss.com seçilerek “Sil” tuşuna basılmıştır. Silme işlemini onaylamak için “netas777 kişisini, kişi listenizden silmek istediğinizden emin misiniz?” uyarı mesajı içeren yeni bir form oluşturulmuştur. “Hayır” tuşuna basarak “Kişi sil” menüsüne geri dönülür. “Evet” tuşuna basıldığında, seçilen kişinin SIP adres bilgisi ile sunucuya Expires: 0 başlıklı bir SUBSCRIBE mesajı gönderilir. Sunucudan istemciye SUBSCRIBE mesajının alınıp, değerlendirildiğine ve başarıyla işlendiğine dair 100 Trying ve 200 OK mesajları gelir. Ardından, istemciye netas777@co10ss.com kişinin üyeliğinin iptal edildiğini bildiren bir NOTIFY mesajı gönderilir. Kişi silme işlemi başarıyla gerçekleştirildikten sonra, “Kişi, listenizden başarıyla silindi.” bilgisi ekrana basılarak işlem sonlandırılır ve “Ana Menü” ye dönülür.

Şekil 7.31’ de ise kişi sil uygulamasına ait istemci ile sunucu arasındaki SIP mesajları akış şeması görülmektedir. Bu şemadaki 47.168.94.20 istemcinin IP adresi, 47.168.16.129 ise sunucunun IP adresidir. 53685, istemcinin gelen SIP mesajlarını alabilmek için dinlediği lokal UDP portudur. 5060 ise, sunucunun gelen SIP mesajlarını alabilmesi için dinlediği standart SIP portudur.

Time	47.168.94.20	47.168.16.129	Comment
0,000	(53685)	(5060)	SIP: Request: SUBSCRIBE sip:netas777@co10ss.com
0,036	(53685)	(5060)	SIP: Status: 100 Trying
0,043	(53685)	(5060)	SIP: Status: 200 OK
0,053	(53685)	(5060)	SIP: Request: NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20

Şekil 7.31 Kişi sil SIP mesajları akış şeması

Bu akış şemasındaki SIP mesaj içerikleri ve konsol çıktısı aşağıdaki gibidir. GÖNDERİLEN MESAJ lar, istemciden sunucuya giden mesajlar, ALINAN MESAJ lar ise, sunucudan istemciye gelen mesajlardır.

Silinecek Kişi: netas777@co10ss.com

GÖNDERİLEN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SUBSCRIBE sip:netas777@co10ss.com SIP/2.0
 From: <sip:netas444@co10ss.com>;tag=cb27-4ed7
 To: <sip:netas777@co10ss.com>
 Cseq: 1 SUBSCRIBE
 Call-ID: 9322-cb27-4ed7-aaa14012@47.168.94.32
 Via: SIP/2.0/UDP 47.168.94.32:53685;branch=aaa14012
 Expires: 0
 Contact: <sip:netas444@47.168.94.32:53685>
 Max-Forwards: 70
 Supported: https
 Event: presence
 Accept: application/cpim-pidf+xml
 l:0

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SIP/2.0 100 Trying
 From: <sip:netas444@co10ss.com>;tag=cb27-4ed7
 To: <sip:netas777@co10ss.com>
 Call-ID: 9322-cb27-4ed7-aaa14012@47.168.94.32
 CSeq: 1 SUBSCRIBE
 Via: SIP/2.0/UDP 47.168.94.32:53685;received=47.168.94.20;branch=aaa14012
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

SIP/2.0 200 OK
 From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=cb27-4ed7
 To: <sip:netas777@co10ss.com>;tag=1219575763476
 Call-ID: 9322-cb27-4ed7-aaa14012@47.168.94.32
 CSeq: 1 SUBSCRIBE
 Via: SIP/2.0/UDP 47.168.94.32:53685;received=47.168.94.20;branch=aaa14012
 Contact: <sip:netas777@co10ss.com:5060;maddr=47.168.16.129>
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 12:21:38 UTC 2008

NOTIFY sip:netas444@47.168.94.32:53685;maddr=47.168.94.20 SIP/2.0
 From: "2165420777"<sip:2165420777@co10ss.com>;tag=1219575763476
 To: "netas444"<sip:netas444@co10ss.com>;tag=cb27-4ed7
 Call-ID: 9322-cb27-4ed7-aaa14012@47.168.94.32
 CSeq: 98 NOTIFY
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b34fd-6a46fd4e-67db2df3
 Content-Type: application/cpim-pidf+xml
 Contact: <sip:2165420777@co10ss.com:5060;maddr=47.168.16.129>
 Max-Forwards: 20
 Subscription-Expires: 0
 Event: presence
 Subscription-State: terminated;expires=0
 Content-Length: 272

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420777@co10ss.com"/>
<t>s3:Be+Right+Back</t>
```

```
<t>p3:Be+Right+Back</t>  
<t>w3:Be+Right+Back</t>  
<t>i3:Be+Right+Back</t>  
<t>d3:Be+Right+Back</t>  
<t>u3:Be+Right+Back</t>  
</presence>
```

Kişi sil uygulaması ile bir kişi silindiğinde,

- Kişi listesi
- Kişi sil
- Anlık ileti gönder
- Arama başlat

menülerindeki kişi listeleri de Şekil 7.32’ de olduğu gibi güncellenir.



Şekil 7.32 Kişi silindikten sonra güncellenen kişi listeleri emülatör çıktısı

7.1.2.6 Anlık İleti Gönderme

İstemci çevrimiçi olduğu sürece, anlık ileti gönderebilecektir. İstemci çevrimiçi durumunda iken anlık ileti gönder uygulamasına ait emülatör çıktısı Şekil 7.33’ te gösterilmiştir.



Şekil 7.33 Çevrimiçi durumunda anlık ileti gönder emülatör çıktısı

“Ana Menü” den “Anlık İleti Gönder” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Anlık İleti Gönder” menüsünü ekrana getirmektir. “Anlık İleti Gönder” menüsünde, anlık ileti gönderilecek kişiyi seçmek için elemanları kişiler olan “Kime” başlıklı bir açılır liste, istemcinin mesajını yazacağı 100 karakterlik “Mesaj” başlıklı bir metin kutusu, anlık ileti göndermekten vazgeçip “Ana Menü” ye dönmek için “Geri” tuşu ve seçilen kişiye metin kutusunda yazılan mesajı göndermek için “Gönder” tuşu tanımlanmıştır.

Şekil 7.34’ teki örnekte, listeden iletinin gönderileceği kişi olarak netas888@co10ss.com seçilerek, metin kutusuna “Merhaba” içerikli bir ileti yazılmış ve “Gönder” tuşuna basılmıştır. “Gönder” tuşuna basıldığı anda istemciden sunucuya ileti içeriği ile birlikte bir MESSAGE talep mesajı gönderilir. Sunucudan istemciye MESSAGE talep mesajının alınıp,

değerlendirildiğine dair 100 Trying cevap mesajı gelir. netas888@co10ss.com kullanıcısı ÇEVRİMDIŞI durumunda olduğundan dolayı, sunucu istemciye netas888@co10ss.com ulaşamadığına dair 480 Temporarily Unavailable cevap mesajı gönderir. Ekranı “netas888 ulaşamıyor! Tekrar deneyiniz.” uyarısı ile birlikte iletinin GÖNDERİLEMEDİ ği bilgisi bastırılır.



Şekil 7.34 Anlık ileti gönderilemedi emülatör çıktısı

Şekil 7.35’ te ise yukarıdaki örnek anlık ileti gönderme uygulamasına ait istemci ile sunucu arasındaki SIP mesajları akış şeması görülmektedir. Bu şemadaki 47.168.94.20 istemcinin IP adresi, 47.168.16.129 ise sunucunun IP adresidir. 49607, istemcinin gelen SIP mesajlarını alabilmek için dinlediği lokal UDP portudur. 5060 ise, sunucunun gelen SIP mesajlarını alabilmesi için dinlediği standart SIP portudur.

Time	47.168.94.20	47.168.16.129	Comment
0,000	(49607) Request: MESSAGE sip:netas888@co10ss.com (text/plain)	(5060)	SIP: Request: MESSAGE sip:netas888@co10ss.com (text/plain)
0,033	(49607) Status: 100 Trying	(5060)	SIP: Status: 100 Trying
0,043	(49607) Status: 480 Tempora	(5060)	SIP: Status: 480 Temporarily Unavailable

Şekil 7.35 Anlık ileti gönderilemedi SIP mesajları akış şeması

Bu akış şemasındaki SIP mesaj içerikleri ve konsol çıktısı aşağıdaki gibidir. GÖNDERİLEN MESAJ lar, istemciden sunucuya giden mesajlar, ALINAN MESAJ lar ise, sunucudan istemciye gelen mesajlardır.

GÖNDERİLEN MESAJ - Sun Aug 24 14:03:12 UTC 2008

```
MESSAGE sip:netas888@co10ss.com SIP/2.0
From: <sip:netas444@co10ss.com>;tag=7a43-70a6
To: <sip:netas888@co10ss.com>
CSeq: 1 MESSAGE
Call-ID: 0924-7a43-70a6-f5ebe260@47.168.94.32
Via: SIP/2.0/UDP 47.168.94.32:49607;branch=f5ebe260
Contact: <sip:netas444@47.168.94.32:49607>
Max-Forwards: 70
Content-Type: text/plain;charset=UTF-8
Content-Length: 7
```

Merhaba

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

```
SIP/2.0 100 Trying
From: <sip:netas444@co10ss.com>;tag=7a43-70a6
To: <sip:netas888@co10ss.com>
Call-ID: 0924-7a43-70a6-f5ebe260@47.168.94.32
CSeq: 1 MESSAGE
Via: SIP/2.0/UDP 47.168.94.32:49607;received=47.168.94.20;branch=f5ebe260
Content-Length: 0
```

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

```
SIP/2.0 480 Temporarily Unavailable
From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=7a43-70a6
To: "SIPLineUser SIPLineUser"<sip:netas888@co10ss.com>;tag=1590623421
Call-ID: 0924-7a43-70a6-f5ebe260@47.168.94.32
CSeq: 1 MESSAGE
Via: SIP/2.0/UDP 47.168.94.32:49607;received=47.168.94.20;branch=f5ebe260
Content-Length: 0
```

netas888@co10ss.com kullanıcısı oturum açıp, ÇEVİRİMİÇİ olduktan sonra istemci ile aralarındaki anlık ileti Şekil 7.37’ deki gibidir. İstemci, herhangi bir şey yazıp “Gönder” tuşuna bastığı anda istemciden sunucuya ileti içeriği ile birlikte bir MESSAGE talep mesajı gönderilir. Sunucudan istemciye MESSAGE talep mesajının alınıp, değerlendirildiğine ve başarıyla karşı tarafa iletilmesine dair 100 Tying ve 200 OK cevap mesajları gelir.

İstemcinin ileti gönderebileceği gibi, netas888@co10ss.com kullanıcısı da istemciye ileti gönderebilir. netas888@co10ss.com ileti gönderdiğinde, sunucudan istemciye ileti içeriğiyle

birlikte MESSAGE talebi gelir. İstemci, gelen bu MESSAGE talep mesajını işleyerek, ileti içeriğini ekrana yazdırır ve iletiyi başarı bir şekilde aldığı bilgisini vermek için sunucuya 200 OK mesajı gönderir.

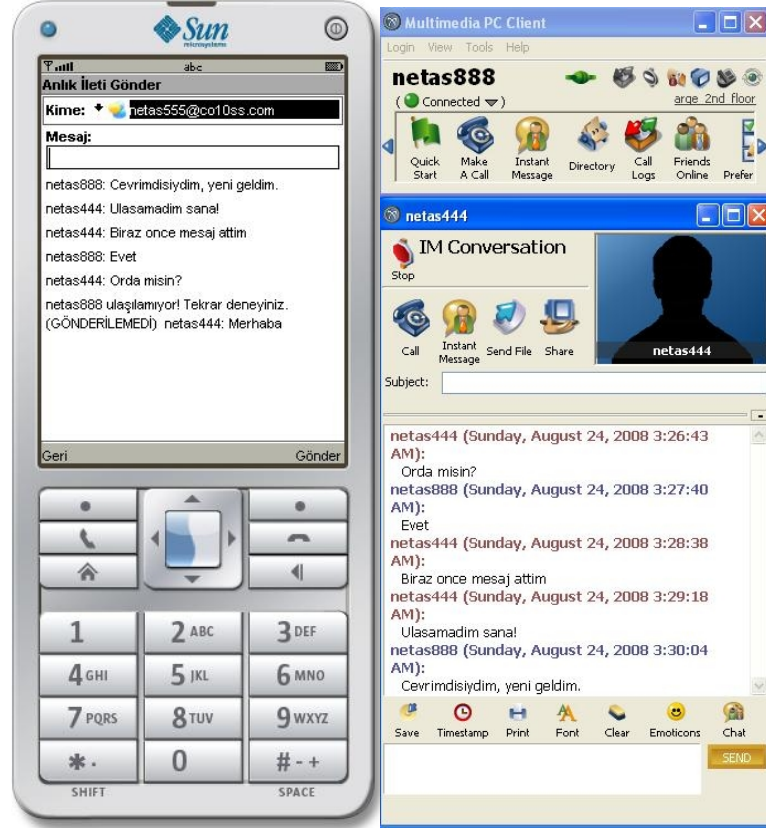
Her anlık ileti alındığında, istemciyi haberdar etmek amacıyla bir uyarı tonu basılır. Bu özellik sayesinde, istemci sürekli “Anlık İleti Gönder” menüsünde olmak zorunda değildir. Kişi listesini inceliyorken, kişi ekleyip, kişi siliyorken, arama yapıyorken, veya bir görüşme sırasındaiken de ileti alındı uyarı tonu ile uyarılarak ileti alabilir.

Dikkat edilecek olursa, en son gelen ileti istemci ekranında ilk satıra yazılır. Bu proje, cep telefonları gibi küçük boyuttaki taşınabilir sistemler için tasarlandığından dolayı, ekranı verimli kullanmak çok önemlidir. Okunabilirliği artırmak adına, gelen iletileri kolay takip edebilmek için bu uygulamada böyle bir yöntem tercih edilmiştir.

Ayrıca, anlık ileti uygulaması, görüldüğü üzere ileti geçmişi de tutmaktadır. Ekrana yazılabilecek gelen giden toplam ileti sayısı 50 lik vektör olarak belirlenmiştir. 51. iletiden sonra, geçmişte ilk gelen iletiden itibaren toplam ileti sayısı 50 yi geçmeyecek şekilde iletiler silinir.



Şekil 7.36 Karşı kullanıcının oturum açma ekranı



Şekil 7.37 İki istemci arasındaki anlık iletişime

Şekil 7.38’ de ise yukarıdaki örnek anlık ileti gönderme uygulamasına ait istemci ile sunucu arasındaki SIP mesajları akış şeması görülmektedir. Bu şemadaki 47.168.94.20 istemcinin IP adresi, 47.168.16.129 ise sunucunun IP adresidir. 49607, istemcinin gelen SIP mesajlarını alabilmek için dinlediği lokal UDP portudur. 5060 ise, sunucunun gelen SIP mesajlarını alabilmesi için dinlediği standart SIP portudur.

Time	47.168.16.129	47.168.94.20	Comment
0,000	(5060) Request: NOTIFY sip:	(49607) SIP: Request: NOTIFY sip:netas444@47.168.94.32:49607;maddr=47.168.94.20	
6,797	(5060) Request: MESSAGE sip:	(49607) SIP: Request: MESSAGE sip:netas888@co10ss.com (text/plain)	
6,830	(5060) Status: 100 Trying	(49607) SIP: Status: 100 Trying	
7,473	(5060) Status: 200 OK	(49607) SIP: Status: 200 OK	
14,557	(5060) Request: MESSAGE sip:	(59296) SIP: Request: MESSAGE sip:netas444@47.168.94.32:59296;maddr=47.168.94.20 (text/plain)	
14,558	(5060) Status: 200 OK	(49607) SIP: Status: 200 OK	
57,393	(5060) Request: MESSAGE sip:	(49607) SIP: Request: MESSAGE sip:netas888@co10ss.com (text/plain)	
57,425	(5060) Status: 100 Trying	(49607) SIP: Status: 100 Trying	
57,483	(5060) Status: 200 OK	(49607) SIP: Status: 200 OK	
77,445	(5060) Request: MESSAGE sip:	(49607) SIP: Request: MESSAGE sip:netas888@co10ss.com (text/plain)	
77,477	(5060) Status: 100 Trying	(49607) SIP: Status: 100 Trying	
77,535	(5060) Status: 200 OK	(49607) SIP: Status: 200 OK	
107,958	(5060) Request: MESSAGE sip:	(54706) SIP: Request: MESSAGE sip:netas444@47.168.94.32:54706;maddr=47.168.94.20 (text/plain)	
107,960	(5060) Status: 200 OK	(49607) SIP: Status: 200 OK	

Şekil 7.38 Anlık ileti gönderildi SIP mesajları akış şeması

Bu akış şemasındaki SIP mesaj içerikleri ve konsol çıktısı aşağıdaki gibidir. GÖNDERİLEN MESAJ lar, istemciden sunucuya giden mesajlar, ALINAN MESAJ lar ise, sunucudan istemciye gelen mesajlardır.

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

```
NOTIFY sip:netas444@47.168.94.32:49607;maddr=47.168.94.20 SIP/2.0
From: "2165420888"<sip:2165420888@co10ss.com>;tag=1219575763488
To: "netas444"<sip:netas444@co10ss.com>;tag=f66c-f2a5
Call-ID: d9b1-f66c-f2a5-ae1e8fea@47.168.94.32
CSeq: 138 NOTIFY
Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b41fb-6a79bd63-730e0842
Content-Type: application/cpim-pidf+xml
Contact: <sip:2165420888@co10ss.com:5060;maddr=47.168.16.129>
Max-Forwards: 20
Subscription-Expires: 76446
Event: presence
Subscription-State: active;expires=76446
Content-Length: 194
```

```
<presence xmlns="urn:ietf:params:cpim-presence:" compressed="true">
<presentity id="2165420888@co10ss.com"/>
<t>s2:</t>
<t>p2:</t>
<t>w2:</t>
<t>i2:</t>
<t>d2:</t>
<t>u2:</t>
</presence>
```

2165420888 --> (ÇEVİRİMİÇİ)

GÖNDERİLEN MESAJ - Sun Aug 24 14:03:12 UTC 2008

```
MESSAGE sip:netas888@co10ss.com SIP/2.0
From: <sip:netas444@co10ss.com>;tag=0513-5756
To: <sip:netas888@co10ss.com>
CSeq: 1 MESSAGE
Call-ID: 9c17-0513-5756-3cba1d9a@47.168.94.32
Via: SIP/2.0/UDP 47.168.94.32:49607;branch=3cba1d9a
Contact: <sip:netas444@47.168.94.32:49607>
Max-Forwards: 70
Content-Type: text/plain;charset=UTF-8
Content-Length: 11
```

Orda misin?

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

```
SIP/2.0 100 Trying
From: <sip:netas444@co10ss.com>;tag=0513-5756
```

To: <sip:netas888@co10ss.com>
 Call-ID: 9c17-0513-5756-3cba1d9a@47.168.94.32
 CSeq: 1 MESSAGE
 Via: SIP/2.0/UDP 47.168.94.32:49607;received=47.168.94.20;branch=3cba1d9a
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

SIP/2.0 200 OK
 From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=0513-5756
 To: "SIPLineUser SIPLineUser"<sip:netas888@co10ss.com>;tag=1d4-f64-b20-f51fc396
 Call-ID: 9c17-0513-5756-3cba1d9a@47.168.94.32
 CSeq: 1 MESSAGE
 Via: SIP/2.0/UDP 47.168.94.32:49607;received=47.168.94.20;branch=3cba1d9a
 Contact: <sip:netas888@co10ss.com:5060;maddr=47.168.16.129>
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

MESSAGE sip:netas444@47.168.94.32:49607;maddr=47.168.94.20 SIP/2.0
 From: "SIPLineUser SIPLineUser"<sip:netas888@co10ss.com>;tag=1d5-f64-b24-f51fe121
 To: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>
 Call-ID: 211ef15fb1710f1913d7541bc648326cdc290ca7@47.168.16.129
 CSeq: 1 MESSAGE
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b4209-6a79f60f-40a2122f
 Content-Type: text/plain;charset=UTF-8
 Contact: <sip:netas888@47.168.16.129:5060>
 Expires: 90
 Max-Forwards: 69
 Accept-Language: en-us
 Content-Length: 4

Evet

KİMDEN : netas888@co10ss.com
 MESAJ : Evet
 MESAJ UZUNLUĞU : 4

GÖNDERİLEN MESAJ - Sun Aug 24 14:03:12 UTC 2008

SIP/2.0 200 OK
 f: "SIPLineUser SIPLineUser"<sip:netas888@co10ss.com>;tag=1d5-f64-b24-f51fe121
 t: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=5e79
 CSeq: 1 MESSAGE
 i: 211ef15fb1710f1913d7541bc648326cdc290ca7@47.168.16.129
 v: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b4209-6a79f60f-40a2122f
 User-Agent: MobileSIPClient
 l: 0

GÖNDERİLEN MESAJ - Sun Aug 24 14:03:12 UTC 2008

MESSAGE sip:netas888@co10ss.com SIP/2.0
 From: <sip:netas444@co10ss.com>;tag=797c-20f1
 To: <sip:netas888@co10ss.com>
 CSeq: 1 MESSAGE
 Call-ID: 6e89-797c-20f1-0c3629cd@47.168.94.32
 Via: SIP/2.0/UDP 47.168.94.32:49607;branch=0c3629cd
 Contact: <sip:netas444@47.168.94.32:49607
 Max-Forwards: 70
 Content-Type: text/plain;charset=UTF-8
 Content-Length: 22

Biraz once mesaj attim

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

SIP/2.0 100 Trying
 From: <sip:netas444@co10ss.com>;tag=797c-20f1
 To: <sip:netas888@co10ss.com>
 Call-ID: 6e89-797c-20f1-0c3629cd@47.168.94.32
 CSeq: 1 MESSAGE
 Via: SIP/2.0/UDP 47.168.94.32:49607;received=47.168.94.20;branch=0c3629cd
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

SIP/2.0 200 OK
 From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=797c-20f1
 To: "SIPLineUser SIPLineUser"<sip:netas888@co10ss.com>;tag=1d6-f64-b20-f5208939
 Call-ID: 6e89-797c-20f1-0c3629cd@47.168.94.32
 CSeq: 1 MESSAGE
 Via: SIP/2.0/UDP 47.168.94.32:49607;received=47.168.94.20;branch=0c3629cd
 Contact: <sip:netas888@co10ss.com:5060;maddr=47.168.16.129>
 Content-Length: 0

GÖNDERİLEN MESAJ - Sun Aug 24 14:03:12 UTC 2008

MESSAGE sip:netas888@co10ss.com SIP/2.0
 From: <sip:netas444@co10ss.com>;tag=8fc5-88f0
 To: <sip:netas888@co10ss.com>
 CSeq: 1 MESSAGE
 Call-ID: 4baf-8fc5-88f0-3fa48460@47.168.94.32
 Via: SIP/2.0/UDP 47.168.94.32:49607;branch=3fa48460
 Contact: <sip:netas444@47.168.94.32:49607
 Max-Forwards: 70
 Content-Type: text/plain;charset=UTF-8
 Content-Length: 16

Ulasamadim sana!

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

SIP/2.0 100 Trying
 From: <sip:netas444@co10ss.com>;tag=8fc5-88f0
 To: <sip:netas888@co10ss.com>
 Call-ID: 4baf-8fc5-88f0-3fa48460@47.168.94.32
 CSeq: 1 MESSAGE
 Via: SIP/2.0/UDP 47.168.94.32:49607;received=47.168.94.20;branch=3fa48460
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

SIP/2.0 200 OK
 From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=8fc5-88f0
 To: "SIPLineUser SIPLineUser"<sip:netas888@co10ss.com>;tag=1d7-f64-b20-f520d786
 Call-ID: 4baf-8fc5-88f0-3fa48460@47.168.94.32
 CSeq: 1 MESSAGE
 Via: SIP/2.0/UDP 47.168.94.32:49607;received=47.168.94.20;branch=3fa48460
 Contact: <sip:netas888@co10ss.com:5060;maddr=47.168.16.129>
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 14:03:12 UTC 2008

MESSAGE sip:netas444@47.168.94.32:49607;maddr=47.168.94.20 SIP/2.0
 From: "SIPLineUser SIPLineUser"<sip:netas888@co10ss.com>;tag=1d8-f64-b24-f5214e1b
 To: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>
 Call-ID: cfd4125fb10013c313d7541bbd89e2b84d13c6ac@47.168.16.129
 CSeq: 1 MESSAGE
 Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b4267-6a7b62e7-25b45a01
 Content-Type: text/plain;charset=UTF-8
 Contact: <sip:netas888@47.168.16.129:5060>
 Expires: 90
 Max-Forwards: 69
 Accept-Language: en-us
 Content-Length: 28

Cevrimdisiyydim, yeni geldim.

KİMDEN : netas888@co10ss.com
 MESAJ : Cevrimdisiyydim, yeni geldim.
 MESAJ UZUNLUĞU : 28

GÖNDERİLEN MESAJ - Sun Aug 24 14:03:12 UTC 2008

SIP/2.0 200 OK
 f: "SIPLineUser SIPLineUser"<sip:netas888@co10ss.com>;tag=1d8-f64-b24-f5214e1b
 t: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=94a4
 CSeq: 1 MESSAGE
 i: cfd4125fb10013c313d7541bbd89e2b84d13c6ac@47.168.16.129
 v: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1b4267-6a7b62e7-25b45a01
 User-Agent: MobileSIPClient
 l: 0

Anlık ileti uygulaması, birden fazla kişi ile aynı ekranda haberleşmeyi de sağlamaktadır. Şekil 7.39’ da olduğu gibi, istemci netas888@co10ss.com ile iletişirken, netas666@co10ss.com’ dan gelen bir iletiyi de alabilmekte ve netas666@co10ss.com’ a da ileti gönderebilmektedir.



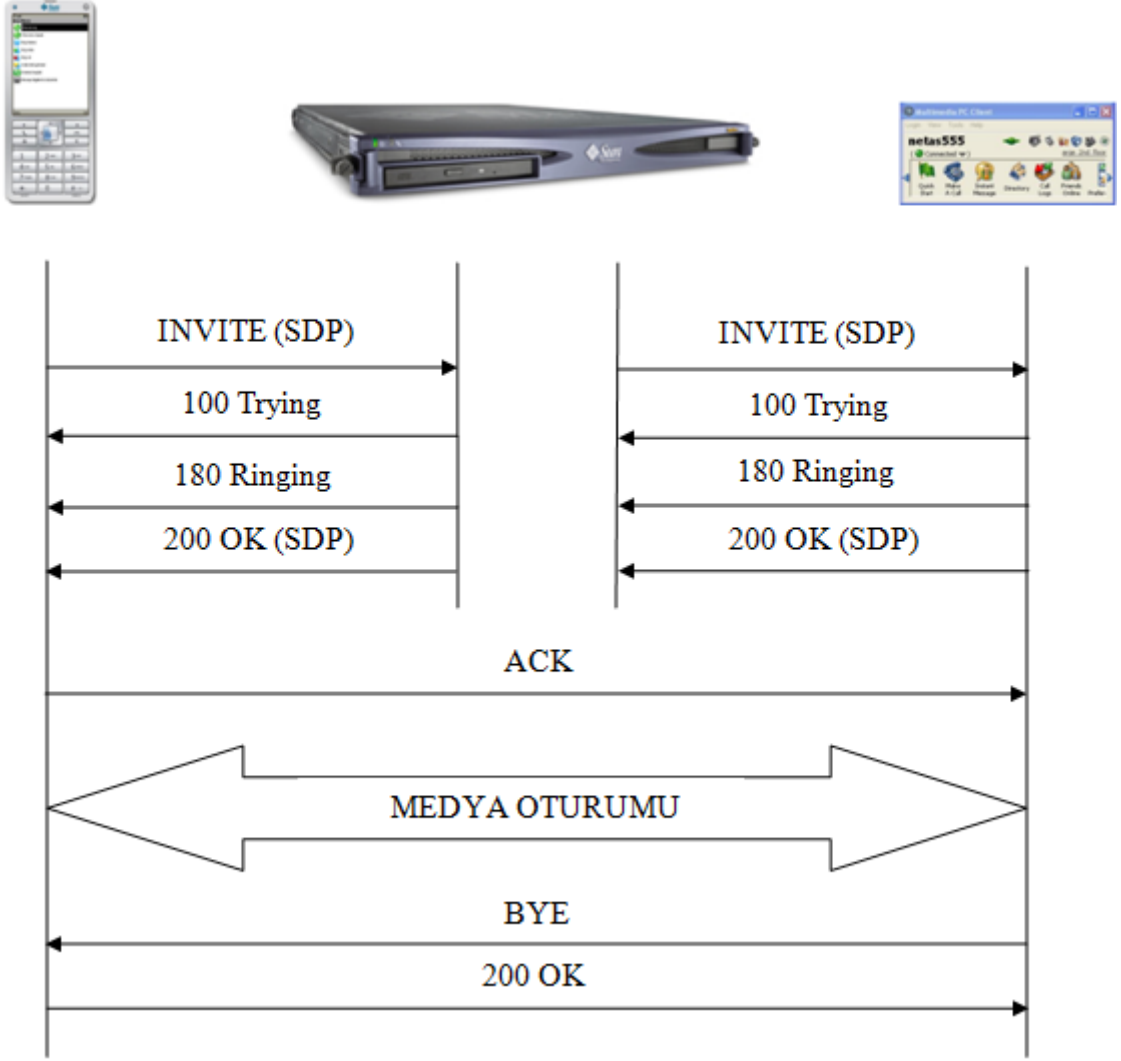
Şekil 7.39 Çoklu anlık iletişime

7.1.2.7 Arama Başlatma

Bu projenin arama başlatma uygulamasında, taşınabilir sistemlerde SIP sinyalleşmesi aracılığı ile iki SIP istemci arasında RTP protokolü kullanılarak internet protokolü üzerinden gerçek zamanlı ses taşınması işleminin gerçekleştirilmesi amaçlanmıştır.

7.1.2.7.1 SIP Mimari Yapısı

Bu uygulamaya ait SIP mimari yapısı, Şekil 7.40’ da gösterilmiştir.



Şekil 7.40 Arama başlat SIP mimari yapısı

Bir arama başlatabilmek için, iki istemci arasındaki sinyalleşme SIP protokolü aracılığı ile sağlanır. Görüşmeye başlamadan önce her iki istemci de, görüşme sırasında kullanacakları

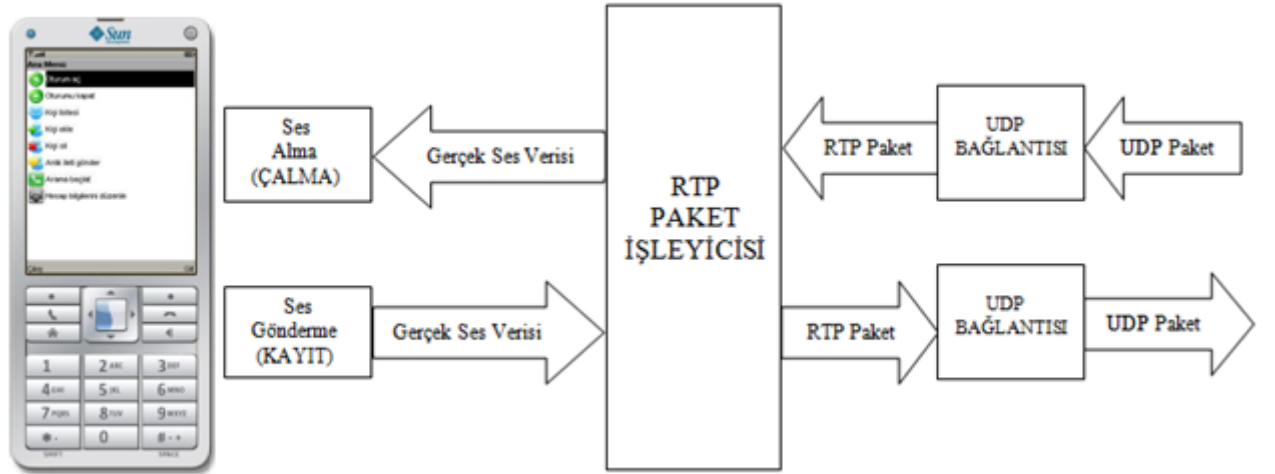
- Medya tipi
- Kodek
- Paketleme zamanı
- Medya verisinin alınacağı IP adresi ve port bilgisi

gibi oturum detay bilgilerini birbirlerine bildirmek zorundadırlar. Bu oturum detay bilgileri, “Oturum Tanımlama Protokolü” olarak adlandırılan SDP yapısı ile bildirilir. Bildirim yapıldıktan sonra, iki istemci ortak bir medya tipinde anlaşarak görüşmeye başlarlar.

İstemci, bir arama başlatmak istediğinde kendi oturum bilgilerini aranan istemciye INVITE talep mesajının içerisinde SDP yapısı ile bildirir (Handley vd., 2006). Sunucudan, arayan istemciye INVITE talep mesajının alınıp, değerlendirildiğine dair 100 Trying ve aranan istemcinin çaldığına dair 180 Ringing cevap mesajları gelir. Aranan istemci, aramayı kabul etmek ve kendi oturum bilgilerini aranan istemciye bildirmek için 200 OK mesajı gönderir. Aranan istemci kendi oturum bilgilerini, 200 OK mesajı içerisinde SDP yapısı ile bildirir (Handley vd., 2006). Akabinde, arayan istemciden, aranan istemciye ACK mesajı gönderilerek oturum kurulur ve görüşme başlar.

7.1.2.7.2 RTP Mimari Yapısı

SIP, bir sinyalleşme protokolüdür, veri taşıma protokolü değildir. Bu projede, medya verisi taşıma işlemi, RTP protokolü ile sağlanmıştır. Tasarlanan RTP protokol mimarisi Şekil 7.41’ de gösterilmiştir.



Şekil 7.41 Arama başlat RTP mimari yapısı

Şekil 7.41’ deki mimari yapıyı gerçeklemek amacıyla, diğer kullanıcıdan istemciye gelen sesi alabilmek için aşağıdaki işlemler adım adım gerçekleştirilmiştir.

- **UDP paketlerini almak için bağlantı oluşturmak:** Arayan kullanıcıdan gelen UDP paketlerini alabilmek için, bir UDP bağlantısı oluşturulur. İstemci, kullanıcıdan gelen ses verisini içeren UDP paketlerini 60000 portunda dinler.

```
String url = "datagram://:60000";
DatagramConnection dc = (DatagramConnection)Connector.open(url);
```

- **UDP paketlerini almak:** UDP bağlantısı oluşturulduktan sonra, istemci UDP paketlerini alır.

```
Datagram dg = null;
dg = dc.newDatagram(length);
dc.receive(dg);
```

- **UDP Paketlerinden RTP Paketlerini çıkartmak:** Alınan UDP paketlerinden, RTP paketleri elde edilir. Elde edilen RTP paketinin büyüklüğü görüşme sırasında kullanılan kodlama formatına göre değişiklik gösterir. Bu uygulamada, medya G.711(PCMU) ses kodeği üzerinden kurulmuştur. G.711(PCMU) ses kodeğinin genel özellikleri aşağıda listelenmiştir.

- Band Genişliği : 64000 bps : 8000 byte per second
- Örnekleme Oranı : 8 bit per sample : 1 byte per sample
- Örnekleme Frekansı : 8 kHz
- Paket Boyutu : 20 ms

PCMU formatında 1 sn' de, toplamda 8000 byte büyüklüğünde bir veri elde edilir. Her bir paket boyutunun 20 ms olduğu düşünülürse, 1 sn' de her biri 160 byte boyutunda 50 tane RTP paketi oluşur. UDP paketinden çıkartılan bu RTP paketinin büyüklüğü 172 byte dır.

```
byte[] rtpPacket = packet.getData();
```

- **RTP Paketlerinden gerçek ses verisini çıkartmak:** UDP paketinden çıkartılan, RTP paketinin büyüklüğü 172 byte dır. 160 byte gerçek ses verisini oluştururken, 12 byte RTP başlık bilgisini içerir. Gerçek ses verisini elde etmek için, RTP paketinin ilk 12 byte lık kısmı çıkartılır.
- **Gerçek ses verisini PCMU formatından PCM8 formatına dönüştürmek:** Elde edilen gerçek ses verisi PCMU formatındadır. Bu projede kullanılan Sun WTK 2.5.2 emülatörünün J2ME kütüphanesi, ses kodeklerinden sadece 8-bit PCM formatını desteklemektedir (Inc. Sun Microsystems, 2004a; 2004b; Knudsen ve Li, 2005). Dolayısıyla, elde edilen PCMU formatındaki gerçek ses verisi, PCM8 formatına dönüştürülür.

```
RTPPayload = audioConversion.ulaw2pcm8(RTPPayload, 0, 160, false);
```

- **PCM8 formatındaki gerçek ses verisine WAV başlığı eklemek:** PCM8 formatına dönüştürülen gerçek ses verisini, istemciye iletebilmek için, ses verisinin başına 44 byte lık bir WAV başlığı eklenir. WAV başlığında, ses verisinin formatına ait kullanılan kodek tipi, kanal sayısı, örnekleme frekansı gibi bilgiler verilir (Craig, 2005).
- **WAV başlığı eklenmiş PCM8 formatındaki gerçek ses verisini çalmak:** WAV başlığı eklenmiş gerçek ses verisi, girdi akışı şeklinde çalınmaya hazırdır.

```
ByteArrayInputStream in=new ByteArrayInputStream(playableVoiceData);
player = Manager.createPlayer(in, "audio/x-wav");
player.realize();
player.start();
```

Ve böylece aranan kullanıcıdan gelen ses, istemciye iletilmiş olur.

Şekil 7.41’ deki mimari yapıyı gerçeklemek amacıyla, istemciden aranan kullanıcıya sesi iletebilmek için aşağıdaki işlemler adım adım gerçekleştirilmiştir.

- **Ses kayıt etmek:** İstemciden aranan kullanıcıya ses verisi göndermek için aşağıdaki yapı oluşturulur.

```
Player p = Manager.createPlayer
("capture://audio?encoding=pcm&rate=8000&bits=8&channels=1");
p.realize();
rc = (RecordControl)p.getControl("RecordControl");
output = new ByteArrayOutputStream();
rc.setRecordStream(output);
rc.startRecord();
p.start();
```

Bu projede kullanılan Sun WTK 2.5.2 emülatörünün J2ME kütüphanesi, ses kodeklerinden sadece 8-bit PCM formatını desteklediğinden dolayı, ses PCM8 formatında kaydedilir.

- **Kayıt edilen sesten WAV başlığını çıkartmak:** Çıktı akışı şeklinde kaydedilen sesi, aranan kullanıcıya gönderebilmek için 44 byte lık WAV başlığı çıkartılır.

- **WAV başlığı çıkartılmış PCM8 formatındaki ses verisini PCMU formatına dönüştürmek:** WAV başlığı çıkartılmış PCM8 formatındaki ses, aranan kullanıcının anlayacağı formata, diğer bir deyişle SDP’ de anlaşılan kodek formatına, PCMU formatına dönüştürülür.

```
recordedVoiceData =
    audioConversion.pcm82ulaw(recordedVoiceData, 0, 160, false);
```

- **WAV başlığı çıkartılmış PCMU formatına dönüştürülmüş ses verisini paketlere ayırmak:** PCMU formatına dönüştürülmüş ses verisini, aranan kullanıcıya gönderebilmek için sesi paketlere ayırmak gerekmektedir. Daha önce de bahsedildiği gibi, PCMU formatında kaydedilmiş 1 sn lik ses, her birisi 160 byte büyüklüğünde 150 pakete ayrılabilir.
- **Ayrılan her bir pakete RTP başlığı eklemek:** Ayrılmış her paketin başına 12 byte büyüklüğünde RTP başlığı eklenerek, RTP paketleri oluşturulur.
- **Oluşturulan RTP paketlerinden UDP paketleri oluşturmak:** Oluşturulan RTP paketlerini UDP bağlantısı ile gönderebilmek için, RTP paketlerinden UDP paketleri oluşturulur.

```
Datagram dg = null;
dg = dc.newDatagram(RTPPacket, RTPPacket.length);
```

Buradaki RTP paket uzunluğu 172’ dir.

- **UDP Paketlerini göndermek için bağlantı oluşturmak:** Oluşturulan UDP paketlerini, aranan kullanıcıya gönderebilmek için, bir UDP bağlantısı oluşturulur. Bağlantı, aranan kullanıcının, 200 OK cevap mesajının SDP kısmında belirttiği IP adresi ve port bilgisi kullanılarak oluşturulur. Aranan kullanıcı, istemciden gidecek UDP paketlerini bu port numarasında dinler.

```
String url = "datagram://" + calleeIp + ":" + calleePort;
DatagramConnection dc = (DatagramConnection)Connector.open(url);
```

Dikkat edilecek olursa, arama uygulamalarında bağlantı direk istemciler arasında kurulur. Paketler, sunucu üzerinden iletilmez. Sunucu, sadece SIP sinyalleşmesini kontrol eder.

- **UDP Paketlerini göndermek:** UDP bağlantısı oluşturulduktan sonra, istemci aranan kullanıcıya ses verisi içeren UDP paketlerini gönderir.

```
dc.send(dg);
```

Ve böylece istemcinin gönderdiği ses, aranan kullanıcıya iletilmiş olur.

7.1.2.7.3 Gerçeklenen Arama Senaryoları

İstemci çevrimiçi olduğu sürece, arama başlatabilecektir. İstemci çevrimiçi durumunda iken arama başlat uygulamasına ait emülatör çıktısı Şekil 7.42’ de gösterilmiştir.



Şekil 7.42 Arama başlat emülatör çıktısı

“Ana Menü” den “Arama başlat” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Arama başlat” menüsünü ekrana getirmektir. “Arama başlat” menüsünde, arama yapılacak kişiyi seçmek için elemanları kişiler olan bir seçenek listesi, arama yaptıktan vazgeçip “Ana Menü” ye dönmek için “Geri” tuşu ve seçilen kişiyi arama işlemini gerçekleştirmek için “Arama başlat” tuşu tanımlanmıştır.

İstemci arama listesinden netas555@co10ss.com seçip, “Arama başlat” tuşuna bastıktan sonra, sunucuya bir INVITE mesajı gönderir. Sunucudan istemciye, INVITE mesajının alınıp değerlendirildiğine dair 100 Trying cevap mesajı gelir. Bu sırada ekrana “Aranıyor...” bilgisi basılır. Akabinde, sunucu aranan kullanıcının çaldığına dair 180 Ringing cevap mesajı gönderir ve istemciyi bilgilendirmek için bu cevap mesajı işlenerek ekrana “Çalıyor...” bilgisi basılır.

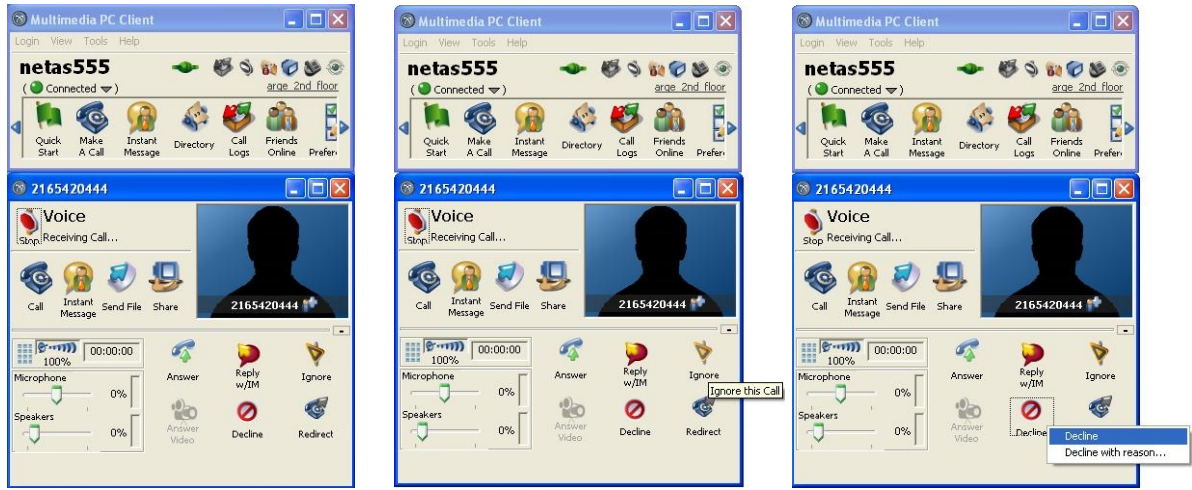


Şekil 7.43 Aranıyor – Çalıyor emülatör çıktısı

Bu aşamadan sonra, arama başlat uygulamasında, 6 tane senaryo gerçekleşmiştir. Bu senaryolar aşağıda listelenmiştir.

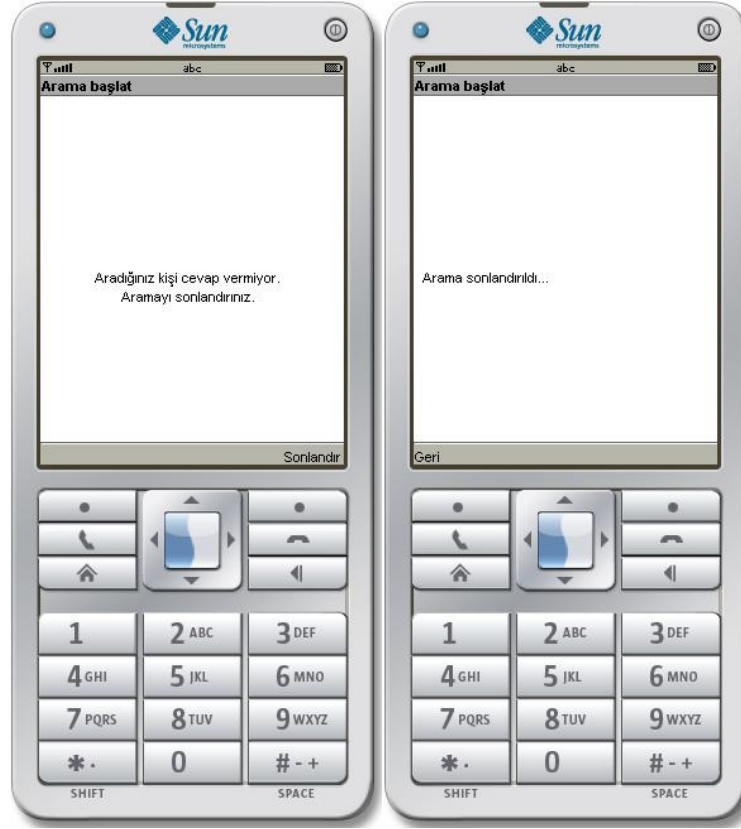
- İstemci aradı, aranan kullanıcı cevap vermedi.
- İstemci aradı, aranan kullanıcı IGNORE etti.
- İstemci aradı, aranan kullanıcı DECLINE etti.
- İstemci aradı, daha sonra aramaktan vazgeçti, aranan kullanıcı cevaplamaadan aramayı sonlandırdı.
- İstemci aradı, aranan kullanıcı cevap verdi, istemci görüşmeyi sonlandırdı.
- İstemci aradı, aranan kullanıcı cevap verdi, aranan kullanıcı görüşmeyi sonlandırdı.

İlk 3 senaryo için, istemci davranışı aynıdır. netas555@co10ss.com Şekil 7.44’ te gösterildiği gibi yapılan aramaya cevap vermeyebilir, yapılan aramayı ihmal veya red edebilir.



Şekil 7.44 Aranan kullanıcı aramayı kabul etmedi ekran çıktısı

Bu durumlarda, sunucu istemciye aramanın kabul edilmediğine dair 183 Session Description cevap mesajı döner. Gelen bu cevap mesajı işlenerek ekrana, “Aradığınız kişi cevap vermiyor. Aramayı sonlandırınız.” bilgisi yazılır. Programlanan “Sonlandır” tuşuna basıldığında, istemciden sunucuya INVITE talebinin iptal edildiğine dair bir CANCEL mesajı gönderilir. Sunucu, CANCEL talep mesajının başarıyla değerlendirildiğine dair istemciye bir 200 OK mesajı ve talebin iptal edildiğine dair 487 Request Terminated mesajı gönderir. İstemci de , ACK onay mesajı göndererek işlemi sonlandırır ve ekrana “Arama sonlandırıldı...” bilgisi basılır.



Şekil 7.45 Arama kabul edilmedi emülatör çıktısı

Şekil 7.46’ da ise yukarıdaki uygulamaya ait istemci ile sunucu arasındaki SIP mesajları akış şeması görülmektedir. Bu şemadaki 47.168.94.20 istemcinin IP adresi, 47.168.16.129 ise sunucunun IP adresidir. 60930, istemcinin gelen SIP mesajlarını alabilmek için dinlediği lokal UDP portudur. 5060 ise, sunucunun gelen SIP mesajlarını alabilmesi için dinlediği standart SIP portudur.

Time	47.168.94.20	47.168.16.129	Comment
0,000	(60930)	Request: INVITE sip:5420555@co10ss.com, with session description	SIP/SDP: Request: INVITE sip:5420555@co10ss.com, with session description
0,047	(60930)	Status: 100 Trying	SIP: Status: 100 Trying
0,903	(60930)	Status: 180 Ringing	SIP: Status: 180 Ringing
30,378	(60930)	Status: 183 Session Description, with session description	SIP/SDP: Status: 183 Session Description, with session description
35,536	(60930)	Request: CANCEL sip:5420555@co10ss.com	SIP: Request: CANCEL sip:5420555@co10ss.com
35,563	(60930)	Status: 200 OK	SIP: Status: 200 OK
35,575	(60930)	Status: 487 Request Terminated	SIP: Status: 487 Request Terminated
36,082	(60930)	Status: 487 Request Terminated	SIP: Status: 487 Request Terminated
36,599	(60930)	Request: ACK sip:5420555@co10ss.com	SIP: Request: ACK sip:5420555@co10ss.com
37,106	(60930)	Request: ACK sip:5420555@co10ss.com	SIP: Request: ACK sip:5420555@co10ss.com

Şekil 7.46 Arama kabul edilmedi SIP mesajları akış şeması

Bu akış şemasındaki SIP mesaj içerikleri ve konsol çıktısı aşağıdaki gibidir. GÖNDERİLEN MESAJ lar, istemciden sunucuya giden mesajlar, ALINAN MESAJ lar ise, sunucudan istemciye gelen mesajlardır.

GÖNDERİLEN MESAJ - Sun Aug 24 18:21:19 UTC 2008

```
INVITE sip:5420555@co10ss.com SIP/2.0
v: SIP/2.0/UDP 47.168.94.32:60930;branch=d1f40f26
t: <sip:5420555@co10ss.com>
f: <sip:netas444@co10ss.com>;tag=39fe-ecc0
i: 18b8-39fe-ecc0-d1f40f26@47.168.94.32
CSeq: 1 INVITE
m: <sip:netas444@47.168.94.32:60930>
Accept: application/sdp
Max-Forwards: 70
Expires: 180
Allow: ACK, MESSAGE, NOTIFY, INVITE, BYE, CANCEL, REFER, OPTIONS, INFO,
PUBLISH, PRACK, UPDATE
Accept-Language: en-us
c: application/sdp
l: 276
```

```
v=0
o=netas444 3009556789 3009556789 IN IP4 47.168.94.32
s=aksamyemegi
c=IN IP4 47.168.94.32
t=0 0
m=audio 60000 RTP/AVP 0 8 18 110 111
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:110 telephone-event/8000
a=rtpmap:111 X-nt-inforeq/8000
a=ptime:20
```

ALINAN MESAJ - Sun Aug 24 18:21:19 UTC 2008

```
SIP/2.0 100 Trying
f: <sip:netas444@co10ss.com>;tag=39fe-ecc0
t: <sip:5420555@co10ss.com>
Call-ID: 18b8-39fe-ecc0-d1f40f26@47.168.94.32
CSeq: 1 INVITE
v: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=d1f40f26
Content-Length: 0
```

ALINAN MESAJ - Sun Aug 24 18:21:19 UTC 2008

```
SIP/2.0 180 Ringing
From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=39fe-ecc0
To: <sip:5420555@co10ss.com>;tag=2789
Call-ID: 18b8-39fe-ecc0-d1f40f26@47.168.94.32
CSeq: 1 INVITE
Via: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=d1f40f26
Contact: <sip:5420555@co10ss.com:5060;maddr=47.168.16.129>
```

Content-Length: 0

ALINAN MESAJ - Sun Aug 24 18:21:19 UTC 2008

SIP/2.0 183 Session Description
 From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=39fe-ecc0
 To: <sip:5420555@co10ss.com>;tag=26390
 Call-ID: 18b8-39fe-ecc0-d1f40f26@47.168.94.32
 CSeq: 1 INVITE
 Via: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=d1f40f26
 Content-Type: application/sdp
 Contact: <sip:5420555@co10ss.com:5060;maddr=47.168.16.129>
 Content-Length: 258

v=0
 o=NNMAS 2 167 IN IP4 47.160.69.51
 s=aksamyemegi
 e=unknown@invalid.net
 c=IN IP4 47.160.69.51
 t=0 0
 m=audio 21144 RTP/AVP 0 110
 c=IN IP4 47.160.69.51
 a=rtpmap:0 PCMU/8000
 a=rtpmap:110 telephone-event/8000
 a=fmtp:110 0-15
 a=ptime:20
 a=sendrecv

GÖNDERİLEN MESAJ - Sun Aug 24 18:21:19 UTC 2008

CANCEL sip:5420555@co10ss.com SIP/2.0
 v: SIP/2.0/UDP 47.168.94.32:60930;branch=d1f40f26
 t: <sip:5420555@co10ss.com>
 f: <sip:netas444@co10ss.com>;tag=39fe-ecc0
 i: 18b8-39fe-ecc0-d1f40f26@47.168.94.32
 CSeq: 1 CANCEL
 l: 0

Arama sonlandırıldı...

ALINAN MESAJ - Sun Aug 24 18:21:19 UTC 2008

SIP/2.0 200 OK
 f: <sip:netas444@co10ss.com>;tag=39fe-ecc0
 t: <sip:5420555@co10ss.com>;tag=6111f400-8110a82f-13c4-1b78c5-42d70669-1b78c5
 Call-ID: 18b8-39fe-ecc0-d1f40f26@47.168.94.32
 CSeq: 1 CANCEL
 v: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=d1f40f26
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 18:21:19 UTC 2008

SIP/2.0 487 Request Terminated
 From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=39fe-ecc0

To: <sip:5420555@co10ss.com>;tag=26390
 Call-ID: 18b8-39fe-ecc0-d1f40f26@47.168.94.32
 CSeq: 1 INVITE
 Via: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=d1f40f26
 Content-Length: 0

ALINAN MESAJ - Sun Aug 24 18:21:19 UTC 2008

SIP/2.0 487 Request Terminated
 From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=39fe-ecc0
 To: <sip:5420555@co10ss.com>;tag=26390
 Call-ID: 18b8-39fe-ecc0-d1f40f26@47.168.94.32
 CSeq: 1 INVITE
 Via: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=d1f40f26
 Content-Length: 0

GÖNDERİLEN MESAJ - Sun Aug 24 18:21:19 UTC 2008

ACK sip:5420555@co10ss.com SIP/2.0
 f: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=39fe-ecc0
 t: <sip:5420555@co10ss.com>;tag=26390
 CSeq: 1 ACK
 i: 18b8-39fe-ecc0-d1f40f26@47.168.94.32
 v: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=d1f40f26
 m: <sip:netas444@47.168.94.32:60930>
 User-Agent: MobileSIPClient
 l: 0

GÖNDERİLEN MESAJ - Sun Aug 24 18:21:19 UTC 2008

ACK sip:5420555@co10ss.com SIP/2.0
 f: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=39fe-ecc0
 t: <sip:5420555@co10ss.com>;tag=26390
 CSeq: 1 ACK
 i: 18b8-39fe-ecc0-d1f40f26@47.168.94.32
 v: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=d1f40f26
 m: <sip:netas444@47.168.94.32:60930>
 User-Agent: MobileSIPClient
 l: 0

4. senaryoda, istemci aramaktan vazgeçip, aranan kullanıcı aramayı cevaplamadan aramayı iptal edebilir. Bu işlemi, ekranda “Çalıyor...” bilgisi yazıyor iken “Sonlandır” tuşuna basarak gerçekleştirir. “Sonlandır” tuşuna basıldığı anda, sunucuya CANCEL mesajı gönderilerek işlem sonlanır.



Şekil 7.47 İstemci aramayı iptal etti emülatör çıktısı

Şekil 7.48’ de ise yukarıdaki uygulamaya ait istemci ile sunucu arasındaki SIP mesajları akış şeması görülmektedir. Bu şemadaki 47.168.94.20 istemcinin IP adresi, 47.168.16.129 ise sunucunun IP adresidir. 60930, istemcinin gelen SIP mesajlarını alabilmek için dinlediği lokal UDP portudur. 5060 ise, sunucunun gelen SIP mesajlarını alabilmesi için dinlediği standart SIP portudur.

Time	47.168.94.20	47.168.16.129	Comment
0,000	(60930) Request: INVITE sip:5420555@co10ss.com, with session description	(5060)	SIP/SDP: Request: INVITE sip:5420555@co10ss.com, with session description
0,046	(60930) Status: 100 Trying	(5060)	SIP: Status: 100 Trying
0,763	(60930) Status: 180 Ringing	(5060)	SIP: Status: 180 Ringing
6,903	(60930) Request: CANCEL sip:5420555@co10ss.com	(5060)	SIP: Request: CANCEL sip:5420555@co10ss.com
6,931	(60930) Status: 200 OK	(5060)	SIP: Status: 200 OK
6,936	(60930) Status: 487 Request Terminated	(5060)	SIP: Status: 487 Request Terminated
7,446	(60930) Status: 487 Request Terminated	(5060)	SIP: Status: 487 Request Terminated
7,980	(60930) Request: ACK sip:5420555@co10ss.com	(5060)	SIP: Request: ACK sip:5420555@co10ss.com
8,458	(60930) Request: ACK sip:5420555@co10ss.com	(5060)	SIP: Request: ACK sip:5420555@co10ss.com

Şekil 7.48 İstemci aramayı iptal etti SIP mesajları akış şeması

Bu akış şemasındaki SIP mesaj içerikleri ve konsol çıktısı aşağıdaki gibidir. GÖNDERİLEN MESAJ lar, istemciden sunucuya giden mesajlar, ALINAN MESAJ lar ise, sunucudan istemciye gelen mesajlardır.

GÖNDERİLEN MESAJ - Sun Aug 24 18:21:19 UTC 2008

```
INVITE sip:5420555@co10ss.com SIP/2.0
v: SIP/2.0/UDP 47.168.94.32:60930;branch=f5b665ac
t: <sip:5420555@co10ss.com>
f: <sip:netas444@co10ss.com>;tag=d754-8bd7
i: 1d07-d754-8bd7-f5b665ac@47.168.94.32
CSeq: 1 INVITE
m: <sip:netas444@47.168.94.32:60930>
Accept: application/sdp
Max-Forwards: 70
Expires: 180
Allow: ACK, MESSAGE, NOTIFY, INVITE, BYE, CANCEL, REFER, OPTIONS, INFO,
PUBLISH, PRACK, UPDATE
Accept-Language: en-us
c: application/sdp
l: 276
```

```
v=0
o=netas444 3009556789 3009556789 IN IP4 47.168.94.32
s=aksamyemegi
c=IN IP4 47.168.94.32
t=0 0
m=audio 60000 RTP/AVP 0 8 18 110 111
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:110 telephone-event/8000
a=rtpmap:111 X-nt-inforeq/8000
a=ptime:20
```

ALINAN MESAJ - Sun Aug 24 18:21:19 UTC 2008

```
SIP/2.0 100 Trying
f: <sip:netas444@co10ss.com>;tag=d754-8bd7
t: <sip:5420555@co10ss.com>
Call-ID: 1d07-d754-8bd7-f5b665ac@47.168.94.32
CSeq: 1 INVITE
v: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=f5b665ac
Content-Length: 0
```

ALINAN MESAJ - Sun Aug 24 18:21:19 UTC 2008

```
SIP/2.0 180 Ringing
From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=d754-8bd7
To: <sip:5420555@co10ss.com>;tag=94802
Call-ID: 1d07-d754-8bd7-f5b665ac@47.168.94.32
CSeq: 1 INVITE
Via: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=f5b665ac
Contact: <sip:5420555@co10ss.com:5060;maddr=47.168.16.129>
Content-Length: 0
```

GÖNDERİLEN MESAJ - Sun Aug 24 18:21:19 UTC 2008

CANCEL sip:5420555@co10ss.com SIP/2.0
v: SIP/2.0/UDP 47.168.94.32:60930;branch=f5b665ac
t: <sip:5420555@co10ss.com>
f: <sip:netas444@co10ss.com>;tag=d754-8bd7
i: 1d07-d754-8bd7-f5b665ac@47.168.94.32
CSeq: 1 CANCEL
l: 0

Arama sonlandırıldı...

ALINAN MESAJ - Sun Aug 24 18:21:19 UTC 2008

SIP/2.0 200 OK
f: <sip:netas444@co10ss.com>;tag=d754-8bd7
t: <sip:5420555@co10ss.com>;tag=61141510-8110a82f-13c4-1b7cb1-3876b276-1b7cb1
Call-ID: 1d07-d754-8bd7-f5b665ac@47.168.94.32
CSeq: 1 CANCEL
v: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=f5b665ac
Content-Length: 0

ALINAN MESAJ - Sun Aug 24 18:21:19 UTC 2008

SIP/2.0 487 Request Terminated
From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=d754-8bd7
To: <sip:5420555@co10ss.com>;tag=94802
Call-ID: 1d07-d754-8bd7-f5b665ac@47.168.94.32
CSeq: 1 INVITE
Via: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=f5b665ac
Content-Length: 0

ALINAN MESAJ - Sun Aug 24 18:21:19 UTC 2008

SIP/2.0 487 Request Terminated
From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=d754-8bd7
To: <sip:5420555@co10ss.com>;tag=94802
Call-ID: 1d07-d754-8bd7-f5b665ac@47.168.94.32
CSeq: 1 INVITE
Via: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=f5b665ac
Content-Length: 0

GÖNDERİLEN MESAJ - Sun Aug 24 18:21:19 UTC 2008

ACK sip:5420555@co10ss.com SIP/2.0
f: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=d754-8bd7
t: <sip:5420555@co10ss.com>;tag=94802
CSeq: 1 ACK
i: 1d07-d754-8bd7-f5b665ac@47.168.94.32
v: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=f5b665ac
m: <sip:netas444@47.168.94.32:60930>
User-Agent: MobileSIPClient
l: 0

GÖNDERİLEN MESAJ - Sun Aug 24 18:21:19 UTC 2008

```

ACK sip:5420555@co10ss.com SIP/2.0
f: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=d754-8bd7
t: <sip:5420555@co10ss.com>;tag=94802
CSeq: 1 ACK
i: 1d07-d754-8bd7-f5b665ac@47.168.94.32
v: SIP/2.0/UDP 47.168.94.32:60930;received=47.168.94.20;branch=f5b665ac
m: <sip:netas444@47.168.94.32:60930>
User-Agent: MobileSIPClient
l: 0

```

5. ve 6. senaryolarda, aranan kullanıcı (netas555@co10ss.com) çağrıyı cevaplar, sunucudan istemciye aranan kullanıcının çağrıyı cevapladığına dair 200 OK mesajı gelir ve istemci ekranına “.....Görüşülüyor.....” bilgisi bastırılarak görüşme başlatılır. Görüşme her iki istemci tarafından da sonlandırılabilir.



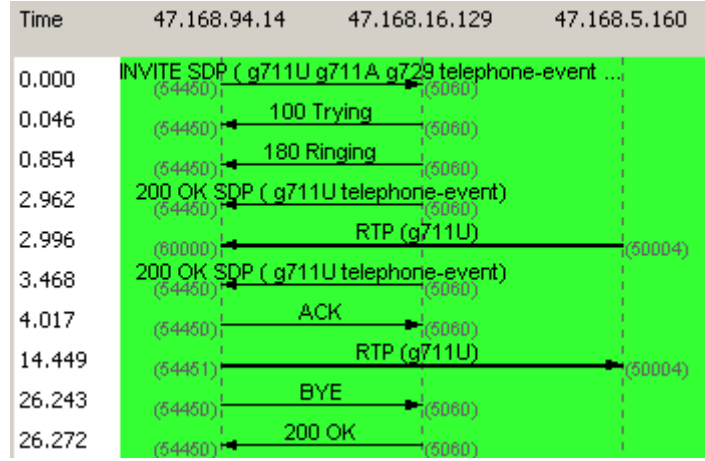
Şekil 7.49 Aranan kullanıcı aramayı cevapladı ekran çıktısı



Şekil 7.50 Arama cevaplandı emülatör çıktısı

Şekil 7.51’ de ise aranan kullanıcının aramayı cevaplamasına ait istemci ile sunucu arasındaki SIP mesajları akış şeması görülmektedir. Bu şemada,

- 47.168.94.14 : İstemcinin IP Adresi
- 47.168.16.129 : Sunucunun IP Adresi
- 47.168.5.160 : Aranan Kullanıcının IP Adresi
- 54450 : İstemcinin gelen SIP mesajlarını alabilmek için dinlediği lokal UDP portu
- 5060 : Sunucunun gelen SIP mesajlarını alabilmesi için dinlediği standart SIP portu
- 60000 : İstemcinin gelen RTP paketlerini alabilmek için dinlediği RTP portu
- 50004 : Aranan kullanıcının gelen RTP paketlerini alabilmek için dinlediği RTP portu



Şekil 7.51 Arama cevaplandı SIP mesajları akış şeması

Bu akış şemasındaki SIP mesaj içerikleri ve konsol çıktısı aşağıdaki gibidir. GÖNDERİLEN MESAJ lar, istemciden sunucuya giden mesajlar, ALINAN MESAJ lar ise, sunucudan istemciye gelen mesajlardır.

Gelen RTP paketlerini almak için MobileSIPClient in dinlemekte olduğu lokal RTP PORTU: 60000

GÖNDERİLEN MESAJ - Sun Aug 24 23:32:27 UTC 2008

```
INVITE sip:5420555@co10ss.com SIP/2.0
v: SIP/2.0/UDP 47.168.94.14:54450;branch=a401235b
t: <sip:5420555@co10ss.com>
f: <sip:netas444@co10ss.com>;tag=6a6b-73b3
i: a313-6a6b-73b3-a401235b@47.168.94.14
CSeq: 1 INVITE
m: <sip:netas444@47.168.94.14:54450>
Accept: application/sdp
Max-Forwards: 70
Expires: 180
Allow: ACK, MESSAGE, NOTIFY, INVITE, BYE, CANCEL, REFER, OPTIONS, INFO,
PUBLISH, PRACK, UPDATE
Accept-Language: en-us
c: application/sdp
l: 276
```

```
v=0
o=netas444 3009556789 3009556789 IN IP4 47.168.94.14
s=aksamyemegi
c=IN IP4 47.168.94.14
t=0 0
m=audio 60000 RTP/AVP 0 8 18 110 111
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:110 telephone-event/8000
```

a=rtpmap:111 X-nt-inforeq/8000
a=ptime:20

ALINAN MESAJ - Sun Aug 24 23:32:27 UTC 2008

SIP/2.0 100 Trying
f: <sip:netas444@co10ss.com>;tag=6a6b-73b3
t: <sip:5420555@co10ss.com>
Call-ID: a313-6a6b-73b3-a401235b@47.168.94.14
CSeq: 1 INVITE
v: SIP/2.0/UDP 47.168.94.14:54450;branch=a401235b
Content-Length: 0

ALINAN MESAJ - Sun Aug 24 23:32:27 UTC 2008

SIP/2.0 180 Ringing
From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=6a6b-73b3
To: <sip:5420555@co10ss.com>;tag=82052
Call-ID: a313-6a6b-73b3-a401235b@47.168.94.14
CSeq: 1 INVITE
Via: SIP/2.0/UDP 47.168.94.14:54450;branch=a401235b
Contact: <sip:5420555@co10ss.com:5060;maddr=47.168.16.129>
Content-Length: 0

ALINAN MESAJ - Sun Aug 24 23:32:27 UTC 2008

SIP/2.0 200 OK
From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=6a6b-73b3
To: <sip:5420555@co10ss.com>;tag=82052
Call-ID: a313-6a6b-73b3-a401235b@47.168.94.14
CSeq: 1 INVITE
Via: SIP/2.0/UDP 47.168.94.14:54450;branch=a401235b
Content-Type: application/sdp
Contact: <sip:5420555@co10ss.com:5060;maddr=47.168.16.129>
Allow: UPDATE
Content-Length: 248

v=0
o=netas555 3009556789 3009556789 IN IP4 47.168.5.160
s=aksamyemegi
e=unknown@invalid.net
c=IN IP4 47.168.5.160
t=0 0
m=audio 50004 RTP/AVP 0 110
c=IN IP4 47.168.5.160
a=rtpmap:0 PCMU/8000
a=rtpmap:110 telephone-event/8000
a=ptime:20

ALINAN MESAJ - Sun Aug 24 23:32:27 UTC 2008

SIP/2.0 200 OK
From: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=6a6b-73b3
To: <sip:5420555@co10ss.com>;tag=82052
Call-ID: a313-6a6b-73b3-a401235b@47.168.94.14
CSeq: 1 INVITE
Via: SIP/2.0/UDP 47.168.94.14:54450;branch=a401235b
Content-Type: application/sdp
Contact: <sip:5420555@co10ss.com:5060;maddr=47.168.16.129>

Allow: UPDATE
Content-Length: 248

```
v=0
o=netas555 3009556789 3009556789 IN IP4 47.168.5.160
s=aksamyemegi
e=unknown@invalid.net
c=IN IP4 47.168.5.160
t=0 0
m=audio 50004 RTP/AVP 0 110
c=IN IP4 47.168.5.160
a=rtpmap:0 PCMU/8000
a=rtpmap:110 telephone-event/8000
a=ptime:20
```

GÖNDERİLEN MESAJ - Sun Aug 24 23:32:27 UTC 2008

```
ACK sip:5420555@co10ss.com SIP/2.0
f: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=6a6b-73b3
t: <sip:5420555@co10ss.com>;tag=82052
CSeq: 1 ACK
i: a313-6a6b-73b3-a401235b@47.168.94.14
v: SIP/2.0/UDP 47.168.94.14:54450;branch=a401235b
m: <sip:netas444@47.168.94.14:54450>
User-Agent: MobileSIPClient
l: 0
```

```
calleeIp   = 47.168.5.160
calleePort = 50082
```

İstemci INVITE mesajının SDP kısmında, yapmak istediği görüşmenin medya tipini(audio), kullanabileceği kodek tiplerini(rtpmap), paketleme zamanı bilgisini(ptime), medya verisinin alınacağı IP adresini(47.168.94.14) ve portunu(60000) tanımlamıştır.

```
v=0
o=netas444 3009556789 3009556789 IN IP4 47.168.94.14
s=aksamyemegi
c=IN IP4 47.168.94.14
t=0 0
m=audio 60000 RTP/AVP 0 8 18 110 111
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:110 telephone-event/8000
a=rtpmap:111 X-nt-inforeq/8000
a=ptime:20
```

Aranan kullanıcı da kendi ortam bilgilerini 200 OK mesajının SDP kısmında tanımladıktan sonra Payload Tipi "0" olan G.711(PCMU) kodeğinde karar kılınarak ortak bir medya kurulmuş ve görüşme başlatılmıştır.

```

v=0
o=netas555 3009556789 3009556789 IN IP4 47.168.5.160
s=aksamyemegi
e=unknown@invalid.net
c=IN IP4 47.168.5.160
t=0 0
m=audio 50004 RTP/AVP 0 110
c=IN IP4 47.168.5.160
a=rtpmap:0 PCMU/8000
a=rtpmap:110 telephone-event/8000
a=ptime:20

```

Time	47.168.94.14	47.168.16.129	47.168.5.160	Comment
14.449	(54451)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1122334455, Seq=5000, Time=100000000
14.454	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30554, Time=1215543834
14.477	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30555, Time=1215543994
14.493	(54451)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1122334455, Seq=5001, Time=100000180
14.494	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30556, Time=1215544154
14.513	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30557, Time=1215544314
14.536	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30558, Time=1215544474
14.540	(54451)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1122334455, Seq=5002, Time=100000320
14.556	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30559, Time=1215544634
14.573	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30560, Time=1215544794
14.597	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30561, Time=1215544954
14.601	(54451)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1122334455, Seq=5003, Time=100000480
14.615	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30562, Time=1215545114
14.632	(54451)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1122334455, Seq=5004, Time=100000640
14.633	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30563, Time=1215545274
14.657	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30564, Time=1215545434
14.675	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30565, Time=1215545594
14.685	(54451)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1122334455, Seq=5005, Time=100000800
14.693	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30566, Time=1215545754
14.716	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30567, Time=1215545914
14.723	(54451)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1122334455, Seq=5006, Time=100000960
14.732	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30568, Time=1215546074
14.757	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30569, Time=1215546234
14.775	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30570, Time=1215546394
14.795	(80000)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1335116377, Seq=30571, Time=1215546554
14.801	(54451)	Payload type=ITU-T	(50004)	RTP: Payload type=ITU-T G.711 PCMU, SSRC=1122334455, Seq=5007, Time=100001120

Şekil 7.52 RTP paket akış şeması

Gelen ve giden RTP paketlerinden bir konsol kesiti:

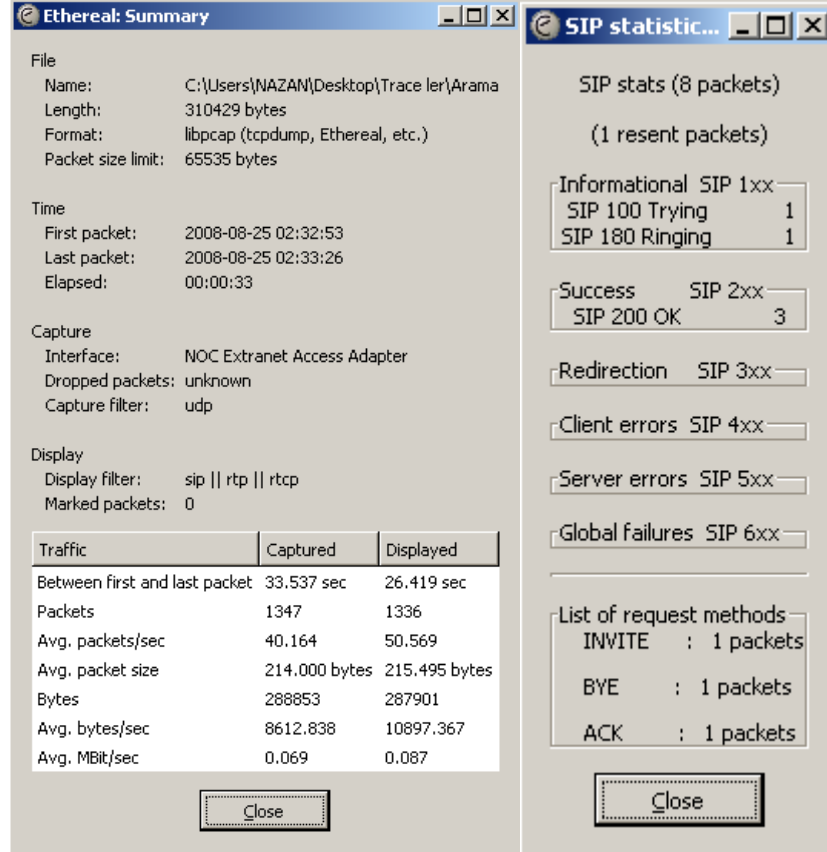
```

RTPPacket 5000: [ ssrc=0x1122334455, timestamp=100000000, payload type=0 ] with media length: 160
RTPPacket 30426: [ ssrc=0x1335116377, timestamp=1215523354, payload type=0 ] with media length: 160
RTPPacket 5001: [ ssrc=0x1122334455, timestamp=100000160, payload type=0 ] with media length: 160
RTPPacket 30427: [ ssrc=0x1335116377, timestamp=1215523514, payload type=0 ] with media length: 160
RTPPacket 5002: [ ssrc=0x1122334455, timestamp=100000320, payload type=0 ] with media length: 160
RTPPacket 30428: [ ssrc=0x1335116377, timestamp=1215523674, payload type=0 ] with media length: 160
RTPPacket 5003: [ ssrc=0x1122334455, timestamp=100000480, payload type=0 ] with media length: 160
RTPPacket 30429: [ ssrc=0x1335116377, timestamp=1215523834, payload type=0 ] with media length: 160
RTPPacket 5004: [ ssrc=0x1122334455, timestamp=100000640, payload type=0 ] with media length: 160
RTPPacket 30430: [ ssrc=0x1335116377, timestamp=1215523994, payload type=0 ] with media length: 160

```

SSRC=0x1122334455 olan paketler istemciden aranan kullanıcıya gönderilen RTP paketleri, SSRC=0x1335116377 olan paketler ise, aranan kullanıcıdan istemciye gönderilen RTP paketleridir.

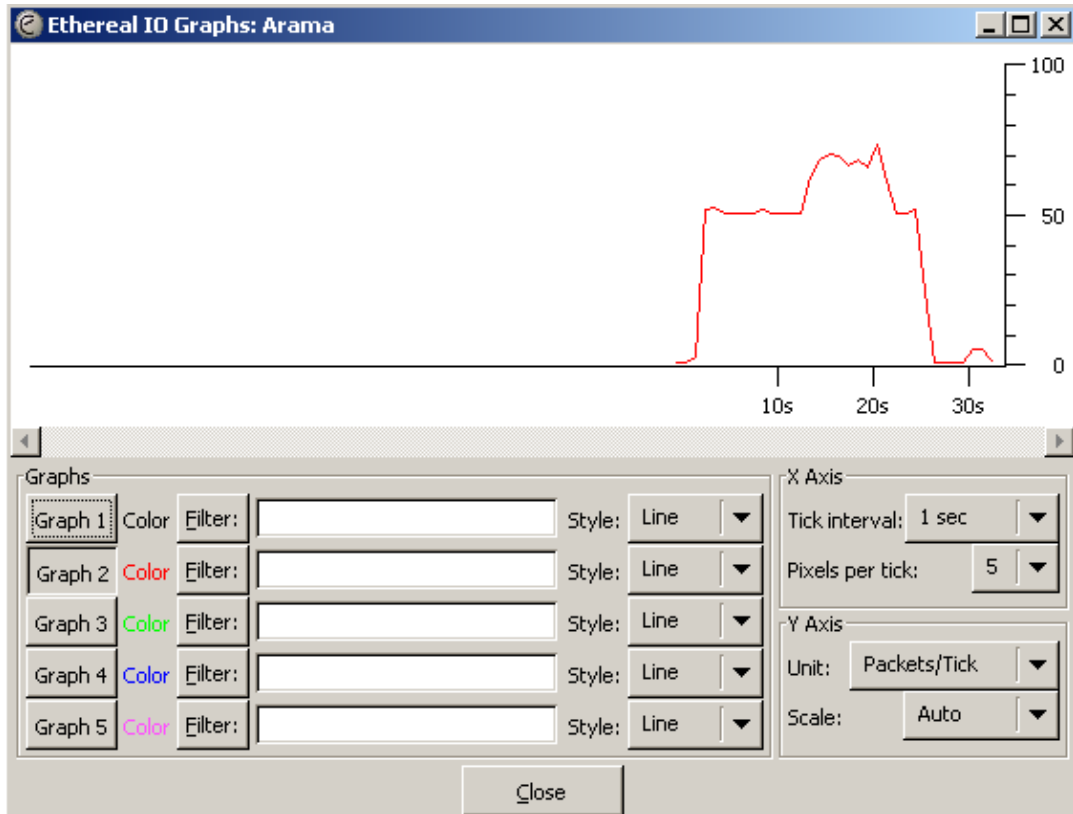
Aşağıda, Ethereal aracından faydalanılarak hazırlanan bu aramaya ait istatistiksel grafikler yer almaktadır.



Şekil 7.53 Arama özet bilgisi

Ethereal: Protocol Hierarchy Statistics								
Protocol	% Packets	Packets	Bytes	Mbit/s	End Packets	End Bytes	End Mbit/s	
Frame	100.00%	1336	287901	0.087	0	0	0.000	
Ethernet	100.00%	1336	287901	0.087	0	0	0.000	
Internet Protocol	100.00%	1336	287901	0.087	0	0	0.000	
User Datagram Protocol	100.00%	1336	287901	0.087	0	0	0.000	
Session Initiation Protocol	0.60%	8	4205	0.001	8	4205	0.001	
Real-Time Transport Protocol	98.95%	1322	282908	0.086	1322	282908	0.086	
Real-time Transport Control Protocol	0.45%	6	788	0.000	0	0	0.000	
Real-time Transport Control Protocol	0.45%	6	788	0.000	5	650	0.000	
Real-time Transport Control Protocol	0.07%	1	138	0.000	1	138	0.000	

Şekil 7.54 Protokol hiyerarşi istatistiği



Şekil 7.55 Arama sırasında sistem IO grafiği

Ethereal: RTP Streams

Detected 2 RTP streams. Choose one for forward and reverse direction for analysis

Src IP addr	Src port	Dest IP addr	Dest port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter (ms)	Mean Jitter (ms)	Pb?
47.168.5.160	50004	47.168.94.14	60000	1335116377	ITU-T G.711 PCMU	1172	0 (0.0%)	29.01	3.24	2.66	
47.168.94.14	54451	47.168.5.160	50004	1122334455	ITU-T G.711 PCMU	150	0 (0.0%)	372.37	55.02	33.94	

Forward: 47.168.94.14:54451 -> 47.168.5.160:50004, SSRC=1122334455
Reverse: 47.168.5.160:50004 -> 47.168.94.14:60000, SSRC=1335116377

Unselect Find Reverse Save As Mark Packets Prepare Filter Copy Analyze Close

Şekil 7.56 RTP akışları listesi

Ethereal: VoIP Calls

Detected 1 VoIP Call. Selected 0 Calls.

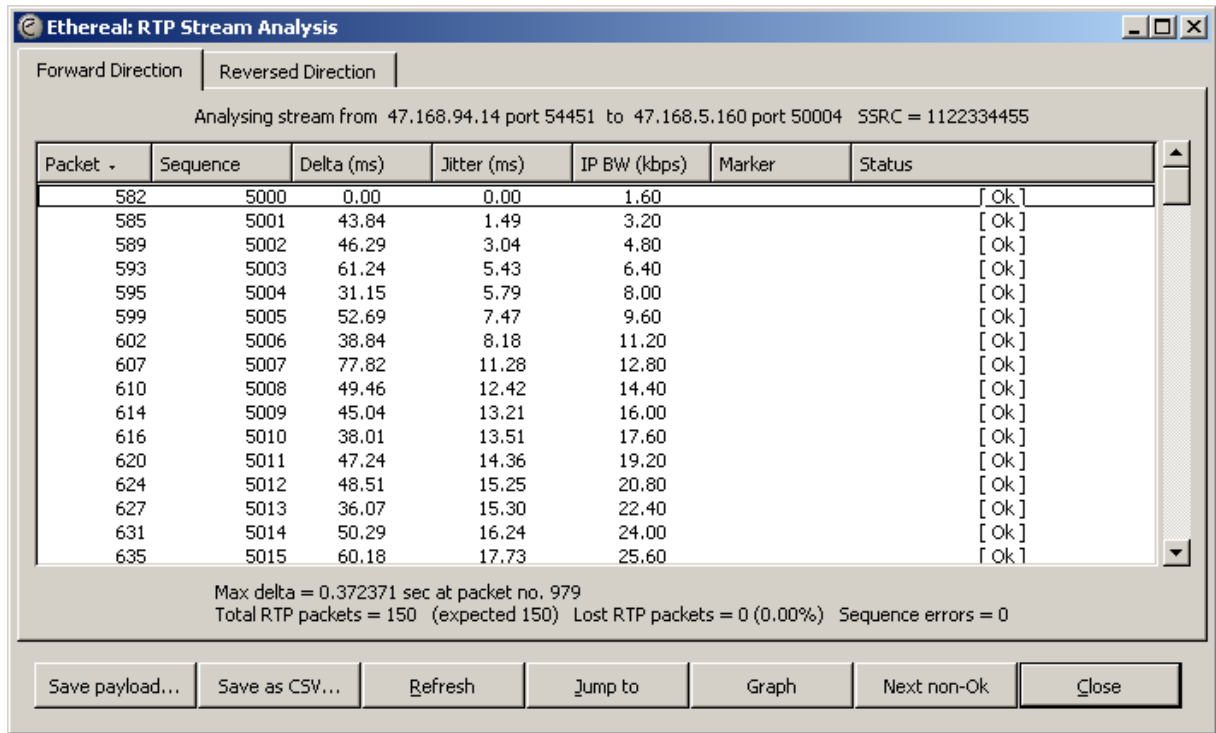
Start Time	Stop Time	Initial Speaker	From	To	Protocol	Packets	State	Comments
0.0	26.27	47.168.94.14	sip:netas444@co10ss.com	sip:5420555@co10ss.com	SIP	8	COMPLETED	

Select one call.

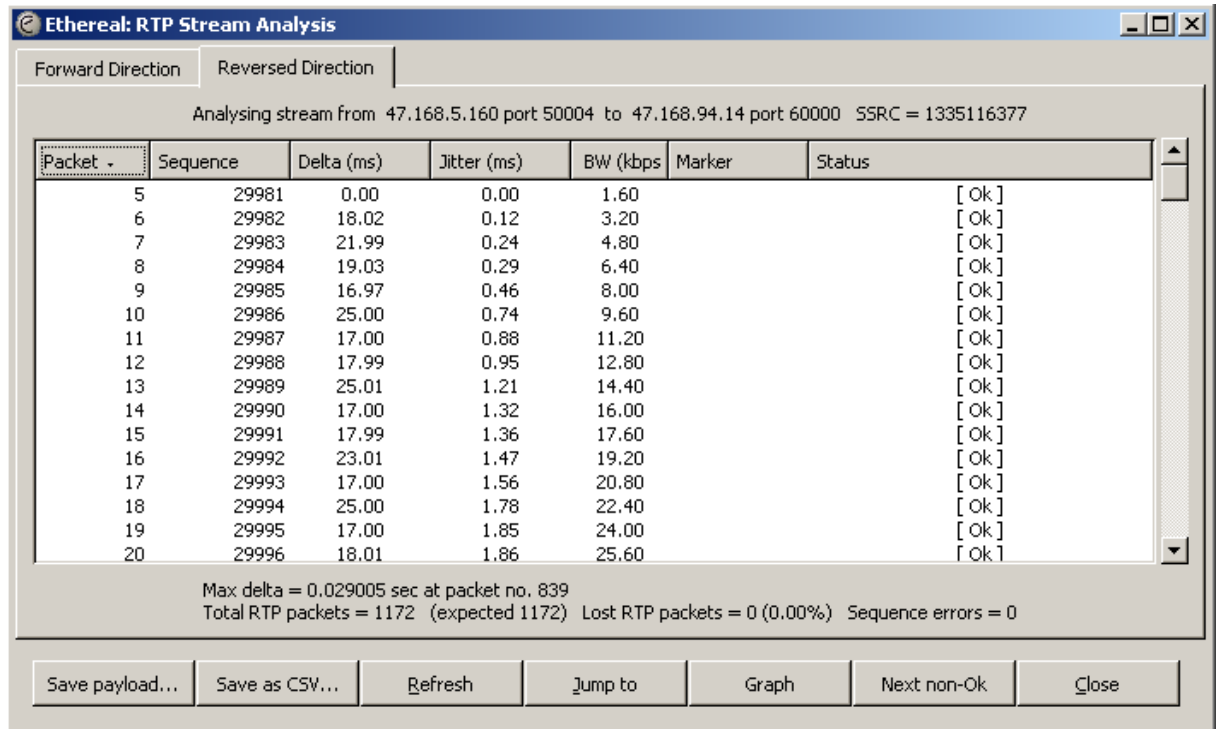
Total: Calls: 1 Start packets: 0 Completed calls: 1 Rejected calls: 0

Prepare Filter Graph Close

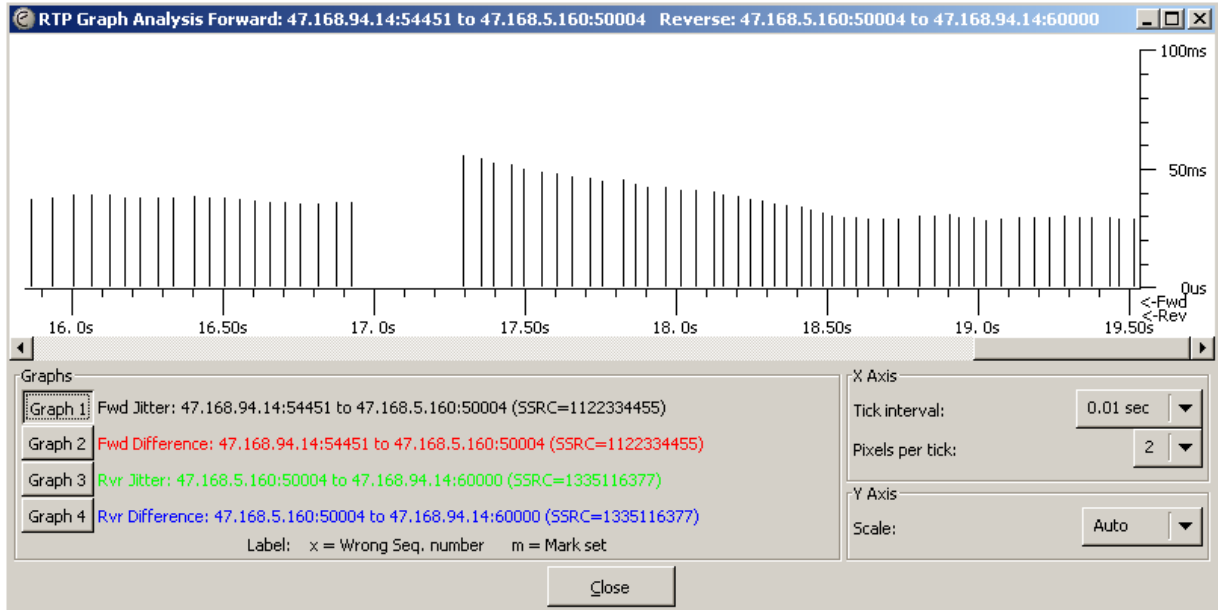
Şekil 7.57 VoIP aramalar listesi



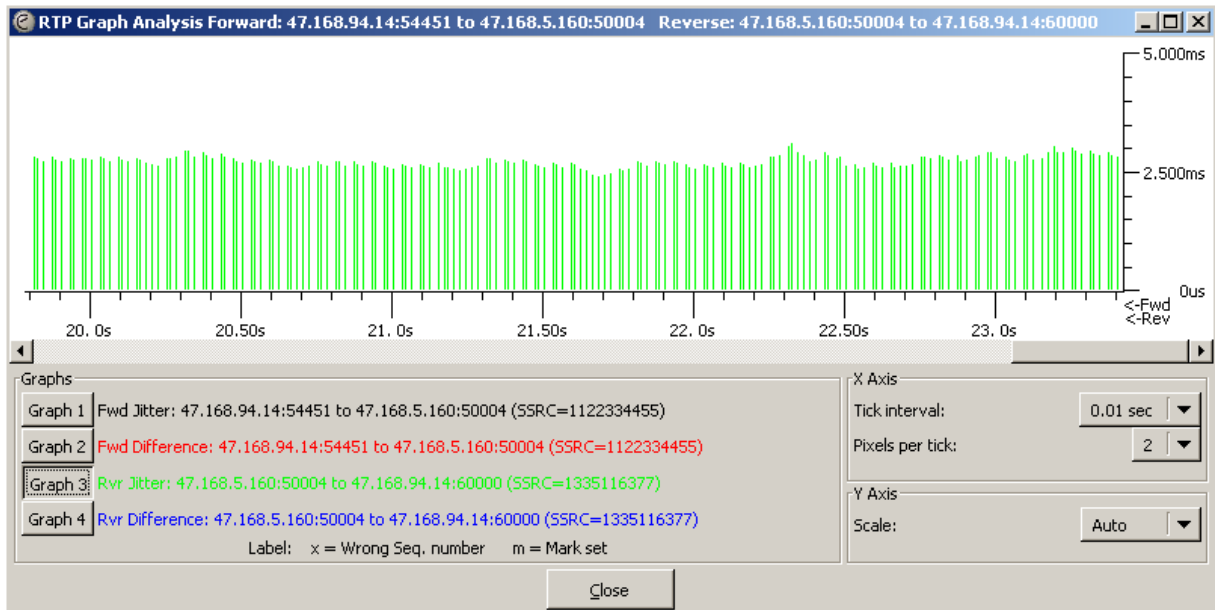
Şekil 7.58 RTP akış analizi (A->B)



Şekil 7.59 RTP akış analizi (B->A)



Şekil 7.60 Jitter grafiği (A->B)

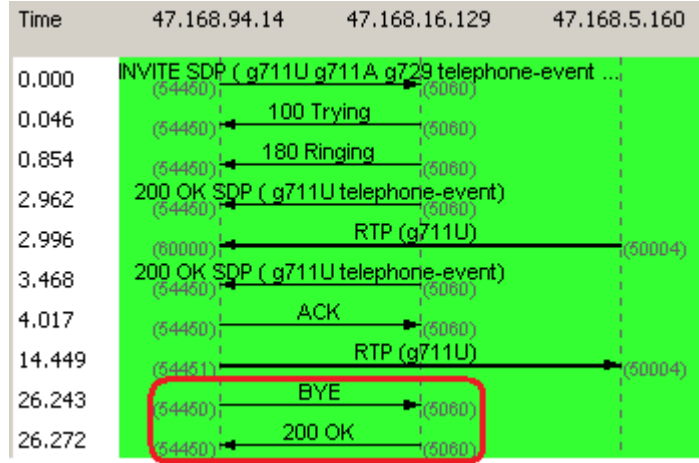


Şekil 7.61 Jitter grafiği (B->A)

Bu görüşme, istemci tarafından sonlanabilir. İlgili şekiller ve SIP mesajları aşağıda gösterilmiştir.



Şekil 7.62 İstemci görüşmeyi sonlandırdı emülatör çıktısı



Şekil 7.63 İstemci görüşmeyi sonlandırdı SIP mesaj akışı

GÖNDERİLEN MESAJ - Sun Aug 24 23:32:27 UTC 2008

BYE sip:5420555@co10ss.com SIP/2.0
 f: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=6a6b-73b3
 t: <sip:5420555@co10ss.com>;tag=82052
 CSeq: 2 BYE
 i: a313-6a6b-73b3-a401235b@47.168.94.14
 v: SIP/2.0/UDP 47.168.94.14:54450;branch=a4012

Max-Forwards: 70
 User-Agent: MobileSIPClient
 1: 0

Arama sonlandırıldı...

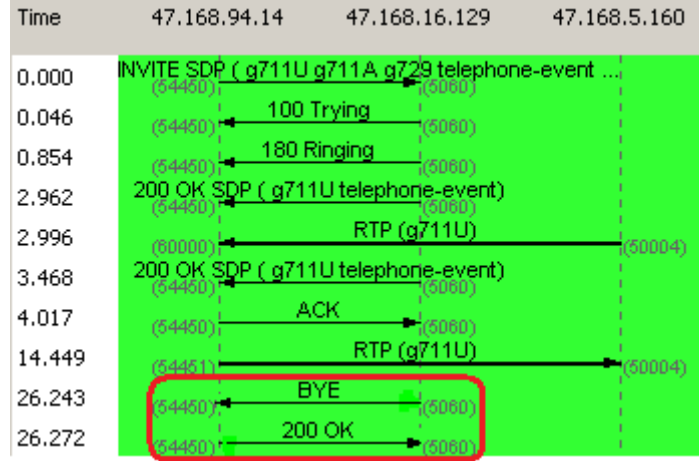
ALINAN MESAJ - Sun Aug 24 23:32:27 UTC 2008

SIP/2.0 200 OK
 f: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=6a6b-73b3
 t: <sip:5420555@co10ss.com>;tag=82052
 Call-ID: a313-6a6b-73b3-a401235b@47.168.94.14
 CSeq: 2 BYE
 v: SIP/2.0/UDP 47.168.94.14:54450;branch=a4012
 Content-Length: 0

Bu görüşme, aranan kullanıcı tarafından sonlanabilir. İlgili şekiller ve SIP mesajları aşağıda gösterilmiştir.



Şekil 7.64 Aranan kullanıcı görüşmeyi sonlandırdı emülatör çıktısı



Şekil 7.65 Aranan kullanıcı görüşmeyi sonlandırdı SIP mesaj akışı

ALINAN MESAJ - Mon Aug 25 00:03:02 UTC 2008

```

BYE sip:netas444@47.168.94.14:59190 SIP/2.0
From: <sip:5420555@co10ss.com>;tag=2024
To: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=93f1-c472
Call-ID: a70d-93f1-c472-a9990778@47.168.94.14
CSeq: 2 BYE
Via: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1bc867-6c86d4b3-3a12059b
Max-Forwards: 20
Content-Length: 0

```

GÖNDERİLEN MESAJ - Mon Aug 25 00:03:02 UTC 2008

```

SIP/2.0 200 OK
f: <sip:5420555@co10ss.com>;tag=2024
t: "SIPLineUser SIPLineUser"<sip:netas444@co10ss.com>;tag=93f1-c472
CSeq: 2 BYE
i: a70d-93f1-c472-a9990778@47.168.94.14
v: SIP/2.0/UDP 47.168.16.129:5060;branch=z9hG4bK-1bc867-6c86d4b3-3a12059b
User-Agent: MobileSIPClient
l: 0

```

7.1.2.8 Hesap Bilgilerini Düzenleme

İstemci çevrimiçi durumdaki iken, oturum açıktır. Dolayısıyla istemcinin, hesap bilgilerini düzenlemesi engellenmiştir. İstemci çevrimiçi durumunda iken hesap bilgilerini düzenle uygulamasına ait emülatör çıktısı Şekil 7.66’ da gösterilmiştir.



Şekil 7.66 Çevrimiçi durumda hesap bilgilerini düzenle emülatör çıktısı

“Ana Menü” den “Hesap bilgilerini düzenle” seçildikten sonra programlanan sağ “Git” tuşuna basılır. Tuşun görevi, “Hesap bilgilerini düzenle” menüsünü ekrana getirmektir. Ancak istemci çevrimiçi olduğu için, “Hesap bilgilerinizi düzenleyebilmek için oturumu kapatmanız gerekmektedir.” uyarısı ekrana basılarak “Ana Menü” ye geri dönülür.

8. SONUÇLAR

Bu projede, taşınabilir sistemler için SIP sinyalleşmesi aracılığı ile VoIP istemci uygulaması başarıyla gerçekleştirilmiştir. Kullanıcılar arası haberleşme, SIP protokolü ile sağlanmıştır. Diğer kullanıcıdan istemciye gelen ve istemciden diğer kullanıcıya gönderilen SIP mesajlarını kontrol edebilmek için bir sunucu - istemci mimari alt yapısı oluşturulmuştur. Sunucu ile istemci arasındaki haberleşme bir UDP bağlantı kurularak sağlanmıştır. İstemci, gelen SIP mesajlarını lokal UDP portundan alarak işlemiştir. Göndereceği SIP mesajlarını da, oluşturduğu UDP bağlantısı ile sunucunun standart SIP portuna göndermiştir. Böylelikle SIP sinyalleşmesi başarılı bir şekilde gerçekleştirilmiştir.

Sinyalleşme SIP protokolü ile sağlanırken, gerçek zamanlı veri transferi RTP protokol yapısı oluşturularak sağlanmıştır.

Gerçek ses verisini içeren UDP paketlerini alabilmek için bir UDP bağlantısı oluşturulmuştur. Bağlantı kurulduktan sonra gelen UDP paketlerinden RTP paketleri elde edilmiştir. Bu paketlerden de 12 byte lık RTP başlık kısmı çıkartılarak gerçek ses verisi elde edilmiştir. J2ME platformu, sadece 8-bit PCM kodlamasını desteklemektedir. Oturum G.711 PCMU kodlamasını üzerinden kurulduğu için gelen RTP paketleri PCMU formatından PCM8 formatına dönüştürülmüştür. PCM8 formatına dönüştürülmüş gerçek ses verisine, istemcinin gelen sesi duyabilmesi için WAV başlık bilgisi eklenmiş, işlem başarıyla tamamlanmıştır. Böylelikle istemcinin diğer kullanıcıdan gelen sesi duyması sağlanmıştır.

İstemciden diğer kullanıcıya ses iletiminde de, benzer şekilde yapı izlenmiştir. İstemcinin kaydedilen ses verisinden WAV başlık bilgisi çıkartılmıştır. WAV başlığı çıkartılmış PCM8 formatındaki kaydedilmiş ses verisi, diğer kullanıcının anlayacağı formata, PCMU formatına dönüştürülmüştür. PCMU formatına dönüştürülmüş ses verisi, diğer kullanıcıya iletmek üzere paketlere ayrılmıştır. Ayrılan her pakete 12 byte lık RTP başlığı eklenerek RTP paketleri oluşturulmuştur. Oluşturulan her RTP paketinden UDP paketleri elde edilmiştir. Bu UDP paketlerini iletmek için diğer kullanıcının IP adresine ve UDP paketlerini dinlediği RTP portuna bir UDP bağlantı oluşturulmuş ve paketler gönderilerek, işlem tamamlanmıştır.

Bu projede, oluşturulan SIP ve RTP mimari yapısı ile istemci, hesap oluşturma, oturum açma, oturum kapatma, kişi listesini görüntüleme, anında kişi durum bilgisi değişikliğini alma, kişi

ekleme, kişi silme, hesap bilgilerini düzenleme, anlık ileti gönderebilme ve arama başlatma uygulamalarını gerçekleyebilmiştir.

Bu projenin arama başlatma uygulaması, VoIP standartları ile uyumlu çalışsa da ileride geliştirebilecek yönleri vardır:

- J2ME platformunda RTP protokol desteği yoktur. Diğer bir deyişle, duraksız akış (streaming) desteklenmemektedir. Bu sebepten, farklı çözümler üretmek gerekir. Bu sorun iki olası yöntemle çözülebilir:
 - DataSource ve SourceStream sınıflarını kullanarak bir RTP çatısı oluşturmak
 - 2 tampon bellekli bir RTP çatısı oluşturmak

T1 zamanında, RTP paketleri birinci belleğe depolanır. T2 zamanında, birinci belleğe depolanmış RTP paketleri istemciye dinletir. Bu sırada gelen RTP paketleri ikinci bellekte depolanır. T3 zamanında, birinci bellek daha önce depoladığı RTP paketlerini istemciye dinletmiş, yeni RTP paketlerini almaya hazırdır. İkinci bellek ise T2 zamanında depoladığı RTP paketlerini T3 zamanında istemciye dinletir. Böyle bir döngü oluşturularak duraksız akış(streaming) özelliği kazandırabilir. Dezavantajı, ilk belleğe yazma işleminde, bellek dolana kadarlık bir süre gecikmesinin oluşması olacaktır.
- Tek yönlü (Half-Duplex) olarak tasarlanan proje, ileride geliştirilip çift yönlü (Full-Duplex) bir haberleşmeye olanak sağlayabilir.

KAYNAKLAR

- Andreasen, F. ve Foster, B., (2003), "Media Gateway Control Protocol (MGCP) Version 1.0", IETF Recommendation RFC 3435, pages 5-33.
- Baughner, M., McGrew, D., Naslund, M., Carrara, E. ve Norrman., K., (2004), "The Secure Real-Time Transport Protocol (SRTP)", IETF Recommendation RFC 3711.
- Bellamy, J.C., (2000), Digital Telephony, 3rd Edition, John Wiley & Sons Inc., New York.
- Braden, R., Zhang, L., Berson, S., Herzog, S. ve Jamin, S., (1997), "Resource Reservation Protocol (RSVP) -- Version 1 Functional Specification", IETF Recommendation RFC 2205.
- Burdet, L., Stuedi, P., Frei, A. ve Alonso, G., (2006), "Symphone: Design and implementation of a VoIP peer for Symbian Mobile phones using bluetooth and SIR", The ACM Digital Library.
- Craig, (2005), "WAVE PCM Soundfile Format",
<http://ccrma.stanford.edu/CCRMA/Courses/422/projects/WaveFormat/>.
- Goyal, V., (2006), "Experiments in Streaming Content in Java ME", Inc. Sun Microsystems,
<http://today.java.net/pub/a/today/2006/08/22/experiments-in-streaming-java-me.html>.
- Handley, M., Jacobson, V. ve Perkins, C., (2006), "SDP: Session Description Protocol", IETF Recommendation RFC 4566.
- Huitema, C., (2003), "Real Time Control Protocol (RTCP) attribute in Session Description Protocol (SDP)", IETF Recommendation RFC 3605.
- Ibe, O.C., (2002), Converged Network Architectures: Delivering Voice over IP, ATM and Framelay, John Wiley & Sons Inc., New York.
- Inc. Sun Microsystems, (2004a), "Java ME Technology APIs & Docs",
<http://java.sun.com/javame/reference/apis.jsp>.
- Inc. Sun Microsystems, (2004b), "The Java ME Platform",
<http://java.sun.com/javame/index.jsp>
- Knudsen J. ve Li S., (2005), Beginning J2ME From Novice to Prefessional, 3rd Edition, Apress, New York.
- Muller, H., Glasmann, J. ve Kellerer, W., (2001), "Service Development and Deployment in H.323 and SIP", Sixth IEEE Symposium on Computers and Communications (ISCC '01).
- Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. ve Schooler, E., (2002), "SIP: Session Initiation Protocol", IETF Recommendation RFC 3261.
- Schildt H., (2003), Herkes İçin Java 2 (Çev. Ü. Yazıcı), ALFA, İstanbul.

Schulzrinne, H. ve Casner, S., (2003a), “RTP Profile for Audio and Video Conferences with Minimal Control”, IETF Recommendation RFC 3551.

Schulzrinne, H., Casner, S., Frederick, R. ve Jacobson, V., (2003b), “RTP: A Transport Protocol for Real-Time Applications”, IETF Recommendation RFC 3550.

Vaudreuil, G. ve Parsons, G., (2004), “Toll Quality Voice – 32 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM) MIME Sub-type Registration”, IETF Recommendation RFC 3802.

3GPP, (2008), “Transparent end-to-end Packet Switched Streaming Service(PSS); Protocols and codecs”, (Release 8), TS 26.234 v.8.0.0.

EKLER

- | | |
|------|---|
| Ek 1 | Ses Kodeklerinin Teknik Özellikler Listesi |
| Ek 2 | Ses Kodeklerinin Yük(Payload) Tipleri Listesi |

Ek 1 Ses Kodeklerinin Teknik Özellikler Listesi

name of encoding	sample/frame	bits/sample	sampling rate	ms/frame	default ms/packet
DVI4	sample	4	var.		20
G722	sample	8	16,000		20
G723	frame	N/A	8,000	30	30
G726-40	sample	5	8,000		20
G726-32	sample	4	8,000		20
G726-24	sample	3	8,000		20
G726-16	sample	2	8,000		20
G728	frame	N/A	8,000	2.5	20
G729	frame	N/A	8,000	10	20
G729D	frame	N/A	8,000	10	20
G729E	frame	N/A	8,000	10	20
GSM	frame	N/A	8,000	20	20
GSM-EFR	frame	N/A	8,000	20	20
L8	sample	8	var.		20
L16	sample	16	var.		20
LPC	frame	N/A	8,000	20	20
MPA	frame	N/A	var.	var.	
PCMA	sample	8	var.		20
PCMU	sample	8	var.		20
QCELP	frame	N/A	8,000	20	20
VDVI	sample	var.	var.		20

Ek 2 Ses Kodeklerinin Yük(Payload) Tipleri Listesi

PT	encoding name	media type	clock rate (Hz)	channels
0	PCMU	A	8,000	1
1	reserved	A		
2	reserved	A		
3	GSM	A	8,000	1
4	G723	A	8,000	1
5	DVI4	A	8,000	1
6	DVI4	A	16,000	1
7	LPC	A	8,000	1
8	PCMA	A	8,000	1
9	G722	A	8,000	1
10	L16	A	44,100	2
11	L16	A	44,100	1
12	QCELP	A	8,000	1
13	CN	A	8,000	1
14	MPA	A	90,000	
15	G728	A	8,000	1
16	DVI4	A	11,025	1
17	DVI4	A	22,050	1
18	G729	A	8,000	1
19	reserved	A		
20	unassigned	A		
21	unassigned	A		
22	unassigned	A		
23	unassigned	A		
dyn	G726-40	A	8,000	1
dyn	G726-32	A	8,000	1
dyn	G726-24	A	8,000	1
dyn	G726-16	A	8,000	1
dyn	G729D	A	8,000	1
dyn	G729E	A	8,000	1
dyn	GSM-EFR	A	8,000	1
dyn	L8	A	var.	var.
dyn	RED	A		
dyn	VDVI	A	var.	1

ÖZGEÇMİŞ

Doğum tarihi 23.04.1983

Doğum yeri Uşak

Lise 1994-2001 Uşak Orhan Dengiz Anadolu Lisesi

Lisans 2001-2005 Eskişehir Osmangazi Üniversitesi Mühendislik Fak.
Elektrik ve Elektronik Mühendisliği Bölümü

Yüksek Lisans 2005-2008 Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü
Elektronik ve Haberleşme Müh. Anabilim Dalı
Haberleşme Programı

Çalıştığı kurum(lar)

2006-Devam ediyor Nortel Netaş A.Ş.