



**T.C.
HALIÇ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
MÜHENDİSLİK ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ PROGRAMI**

**SIP PROTOKOLÜ İLE İNTERNET ÜZERİNDEN
IVR ETKİLEŞİMLİ ÇAĞRI MERKEZİ HİZMETİ**

YÜKSEK LİSANS TEZİ

**Hazırlayan
Mehmet KARABAYIR
22.05.2008.039**

**Danışmanı
Yrd.Doç.Dr.Yüksel BAL**

İstanbul – 2010

ÖNSÖZ

İnternet üzerinden SIP protokolünden faydalanarak IVR etkileşimli çağrı merkezi hizmetinin nasıl verileceği konusunda tez olarak hazırlanmıştır.

Çağrı merkezlerine kazandırılacak ekstra özellikler sayesinde minimum kaynak tüketimiyle nasıl maksimum kazanç sağlanımı elde edilebileceği işlenmiştir.

Haliç Üniversitesindeki eğitim hayatım boyunca tecrübelerini ve değerli bilgilerini bizimle paylaşan çok değerli hocalarıma,

Bitirme projesi olarak seçtiğim bu konuda ilgisini ve yardımını benden hiçbir zaman esirgemeyen tez danışmanı hocam Yüksel Bal'a,

Bugünlere gelmemde en önemli paya sahip, benden hiçbir zaman sevgilerini ve desteklerini esirgemeyen aileme

Sonsuz teşekkürlerimi sunarım.

İstanbul, 2010

Mehmet Karabayır

İÇİNDEKİLER

	Sayfa No
KISALTMALAR LİSTESİ	i
TABLolar LİSTESİ	ii
ŞEKİLLER LİSTESİ	iii
ÖZET	iv
ABSTRACT	vi
1. GİRİŞ	1
1.1. Tezin Amacı	1
1.2. Tezin Kapsamı	2
1.3. Tezin Hedefleri.....	3
1.4. Kullanım Alanları & Avantajları	3
2. ÖZEL SANTRALLER (PBX)	5
2.1. Genel Bakış	5
2.2. Özel Santral Türleri	5
2.3. Özel Santral Sunucu Yazılımları	8
3. MEGACO	11
4. GERÇEK ZAMANLI İLETİM PROTOKOLÜ (RTP)	15
4.1. Genel Bakış	15
4.2. RTP Güvenliği	16
5. GERÇEK ZAMANLI KONTROL PROTOKOLÜ (RTCP)	17
6. ÇİFT YÖNLÜ ÇOKLU FREKANS KODLAMA (DTMF)	19
7. OTURUM BAŞLATMA PROTOKOLÜ (SIP)	23
7.1. Genel Bakış	23
7.2. Oturum Tanımlama Protokolü (SDP).....	25
7.3. Paket Yapısı	28

7.4. İstekler, Yanıtlar ve Durum Kodları.....	28
7.5. Protokol Güvenliği	39
7.6. SIP ve H.323 protokollerinin karşılaştırılması	43
8. SES SINIFLARI.....	46
9. UYGULAMA	48
9.1. Asterisk Genel Bakış, Kurulum ve Konfigürasyonu	48
9.2. Çağrı Akış Planlaması ve Çağrı Grupları	64
9.3. Asterisk Ağ Geçidi Arayüz Programlaması (AGI).....	75
9.4. İnternet Telefonu Uygulamaları	80
9.5. Sesli Yanıt Sistemi Web Servisleri	81
10. SONUÇ.....	83
11. KAYNAKLAR	86
12. ÖZGEÇMİŞ	87

KISALTMALAR LİSTESİ

İngilizce Kısaltmalar

ACW	: Acknowledge
AGI	: Asterisk Gateway Interface
DS1	: Digital Signal 1
DTMF	: Dual Tone Multi Frequency
GSM	: Global System for Mobile Communications
IETF	: Internet Engineering Task Force
IP	: Internet Protocol
ISDN	: Integrated Services Digital Network
ITU	: International Telecommunication Union
IVR	: Interactive Voice Response
PBX	: Private Branch Exchange
PRI	: Primary Rate Interface
PSTN	: Public Switched Telephone Network
RFC	: Request For Comments
RTP	: Real-Time Transport Protocol
RTCP	: Real-Time Transport Control Protocol
SIP	: Session Initiation Protocol
SRTP	: Secure Real-Time Transport Protocol
SSH	: Secure Shell
UDP	: User Datagram Protocol
VOIP	: Voice over Internet Protocol

Türkçe Kısaltmalar

KOBI	: Küçük – Orta – Büyük İşletme
------	--------------------------------

TABLULAR LİSTESİ

	Sayfa No.
Tablo 4.1.1 RTP Paket Şeması.....	15
Tablo 7.2.1 SDP mesajı öge detayları	27
Tablo 7.2.2 Zaman dilimi öge detayları.....	27
Tablo 7.2.3 Medya katmanı öge detayları.....	28
Tablo 7.4.1 SIP vs H323 Karşılaştırma	45
Tablo 9.1.1 Asterisk Çağrı Performans Testi.....	49
Tablo 9.1.2 Asterisk için Gereken Kütüphaneler	51
Tablo 9.1.3 SIP Kullanıcı Hesap Bilgileri.....	61

ŞEKİLLER LİSTESİ

	Sayfa No.
Şekil 2.2.1 Tipik bir santralin ana bileşenleri	7
Şekil 3.1 MEGACO / H.248	12
Şekil 6.1 DTMF tuşlama frekans gösterimi	19
Şekil 7.1.1 Oturum Başlatma Örneği	24
Şekil 7.2.1 SDP Mesajı.....	26
Şekil 7.4.1 SIP İstek Mesajı	29
Şekil 7.4.2 SIP Yanıt Mesajı	30
Şekil 7.4.3 Çağrı sırasında UAC – UAS davranışları.....	32
Şekil 7.4.4 SIP Oturum Başlatma	33
Şekil 7.4.5 Farklı etki alanında bulunan bir cihazla oturum başlatma	35
Şekil 7.4.6 SIP Çağrı Kurulumu.....	38
Şekil 7.4.7 SIP Yanıt Mesajı Detayı	41
Şekil 7.4.8 SIP Yanıt Mesajı Detayı (Authorization)	42
Şekil 9.1.1 Güvenlik Duvarı Konfigürasyonu Ana Ekran.....	55
Şekil 9.1.2 Güvenlik Duvarı Konfigürasyonu Detay Ekranı	55
Şekil 9.1.3 SIP Çağrı Başlatma (canreinvite=no).....	62

GENEL BİLGİLER

Adı ve Soyadı : Mehmet Karabayır
Anabilim Dalı : Mühendislik
Programı : Bilgisayar Mühendisliği
Tez Danışmanı : Yrd.Doç.Dr.Yüksel Bal
Tez Türü ve Tarihi : Yüksek Lisans – Haziran 2010

SIP PROTOKOLÜ İLE İNTERNET ÜZERİNDEN IVR ETKİLEŞİMLİ ÇAĞRI MERKEZİ HİZMETİ

ÖZET

Gelişen teknolojiyle beraber firmalar arasında rekabet, kimin daha iyi servis verebildiğiyle ölçülebilir hale gelmiştir. Artık bir şeyleri satmak tek başına yeterli olmamaktadır. Günümüzde alıcılar satın almış oldukları ürünlerin satış sonrası hizmetlerine de bir hayli önem vermektedir. Bir çok firma müşterilerinde güvenilir imajını sağlayabilmek için satış sonrası destek hizmetine yönelmiştir. Bu sebepten dolayı da çağrı merkezi hizmetleri giderek önem kazanmıştır.

Çağrı merkezlerinin gelişimi de iletişim araçlarının gelişmesiyle birlikte daha da büyümektedir. Çağrı merkezlerinde en önemli hedeflerden bir tanesi ulaşılabilirliğin artırılmasının sağlanmasıdır. Çağrı merkeziyle telekom operatörlerinin iletişiminin fiziksel bağlantılarla sağlanması, personelin hizmet verebilmesi için gerekli mekanın kurulması, iletişimin sağlanabilmesi için gerekli olan diğer altyapı maliyetleri, kaybedilen insan gücünün ve zaman sarfiyatının ciddi oranlara ulaşmasından dolayı bu tez çalışmasında bu kayıpların nasıl kazanıma dönüştürülebileceği anlatılmaktadır.

Konum bağımlılığı sebebiyle şirketler seçmiş oldukları personelleri şirketin yakın çevresinden seçmeye dikkat etmektedirler. Bu şekilde davranarak zamandan ve taşıma giderlerinden tasarruf yapmaya çalışmaktadırlar. Ancak personellere konum sağlamaları gerektiğinden dolayı, ne kadar tasarruf yapılmaya çalışılırsa çalışılsın, yine de önemli oranda kayıplar halen bulunmaktadır.

Çağrı merkezinde standart bilgilendirme işlerinin insanlar tarafından yaptırılmasından dolayı da insan gücü kaybı oldukça fazladır. Bunun yerine sesli yanıt sistemlerinden faydalanarak kaybedilen insan gücü kaybının farklı alanlarda kullanılarak geri kazanımının sağlanması da amaçlanmıştır.

Diğer bir önemli konu da iletişimin operatörle ilgili firma arasında kurulan fiziksel bağlantıdır. Bu fiziksel bağlantı çağrı merkezi hizmeti veren firmaları konum bağımlılığına itmektedir. Operatörle çağrı merkezi arasında kurulan bu fiziksel bağlantı sayesinde ilgili operatörün kullanıcılarının çağrı merkeziyle iletişim kurması sağlanabilmektedir. Oldukça maliyetli olan bu yatırım yerine internet üzerinden bağlantıların kurulmasının sağlanarak dünyanın herhangi bir yerinde ki herhangi bir kullanıcıya hizmet verilebilmesi de amaçlar arasındadır.

Tek merkezli çağrı merkezlerinin risk bölüştürmeleri yeterli düzeyde olmadığından dolayı herhangi bir aksilik anında sistemlerinin devre dışı kalması da diğer bir önemli maddedir. Çağrı merkezi müşteri temsilcilerinin hepsinin aynı konumda bulunması olası bir afet durumunda tüm çağrı merkezi hizmetinin devre dışı kalmasına neden olabilir. Tez çalışmasında bu risk, birden fazla konumda bulunan personel üzerinden hizmet vermesi sağlanarak dağıtılmaya çalışılmıştır.

Anahtar Kelimeler : SIP, IVR, Çağrı Merkezi, Konum bağımsızlığı.

GENERAL KNOWLEDGE

Name and Surname : Mehmet Karabayır
Field : Engineering
Program : Computer Engineering
Supervisor : Yrd.Doç.Dr.Yüksel Bal
Degree Awarded and Date : Master – Haziran 2010

IVR INTEGRATED CALL CENTER SERVICES WITH SIP PROTOCOL

ABSTRACT

With the improvement in technology, competition between companies are measured with who gives efficient services. Selling something is not enough alone. Today, customers are more interested about postsale support. Lots of companies try to provide reliability in the eyes of customers. As a result of this call centers are come into prominence.

Improvement of pulpits make also call centers improvements. One of the most important issue in a call center is having much accessibility. Providing physical connection between telecommunication operators and call centers, providing place for employees to serve, the other infrastructure to provide communication, as a result of so much loosing man power and time, in this project aimed at how to earn this losses back.

Because of fixed places, companies pay attention in selection of personnels and try to select personnels who are closer to company's place. Treating like this trying to provide economy of time and money. But because of obligation of place they haven't save money and time enough.

Also in call centers especially in static informational calls, call centers use personnel as a result of this there are much man power loses. Instead of this if they use interactive voice responses much effective they will prevent loses of man power and they will use this power in a different area in the work, thus they will gain man power back.

The other important issue is physical communication between telecommunicaton operators and call centers. This physical communication causes having fixed places for call centers. Through this connection people can contact with call centers. Instead of this fairly costly investment, connecting people over internet provides much economy both time and money and also make accessibility area much bigger than physical connections.

Due to in unicentral call centers risk sharings aren't enough, in a disaster they will be out of service. Having all the employees in the same place is a problem when a disaster occurs. In this project this risk sharing will be provided.

Keywords: SIP, IVR, Call Center, Mobility.

1. GİRİŞ

Bu tez çalışmasında çağrı merkezi hizmeti veren kurumların, yaygın olarak kullanmakta olduğu konum bazlı personel ve sistemlerin, şirket bütçelerinde ekstra maliyetlere sebep olmasının düzenlemeler yapılarak daha aza indirgenebilmesi sağlanmaktadır. Günümüz çağrı merkezleri, bu hizmeti verebilmek için büyük yatırımlar yaparak, belirlemiş oldukları konumlarda sunucu ve eleman barındırmaktadırlar. Tezde konumun ve elemanın sabit bir noktada bulunma zorunluluğu kaldırılarak giderlerin azaltılması sağlanmaktadır. Bunun yanı sıra insan gücü kazanımları, zaman kazanımı, kaynakların daha etkin bir şekilde kullanımı, risk bölüştürmesi, altyapı ve diğer giderlerin azaltılması, bağımlılıkların en aza indirgenmesi, esnek ve etkileşimli sistemlerin kurulabilmesi ve servis verilebilirlik alanının genişletilmesi de amaçlanmıştır. Çağrı merkezleri için oldukça önem arz eden bu konularda yapılabilecek iyileştirmeler uygulamalarıyla beraber hazırlanmıştır.

Günümüz müşterilerinin bilinçlenmesiyle beraber firmaların satış sonrası destek hizmetinin önemi oldukça artmıştır. Bu da satış sonrası destek hizmetlerinden biri olan çağrı merkezi hizmetini de doğru oranda etkilemiştir. Firmaların bu çağrı merkezi ihtiyaçlarının isterlerse kendi bünyelerinde de çok büyük masraflar yapmaksızın nasıl yapabilecekleri de bu tezde anlatılmaktadır.

1.1. Tezin Amacı

Bu tez çalışmasının amacı firmaların çağrı merkezi kurmak için harcamış olduğu yüklü miktarda ki giderlerin kısılmasını sağlamak, personellere mobilite (konum bağımsızlığı) kazandırmaktır. Bilinen IVR (Interactive Voice Response) uygulamaları, 2 Mbit/s transmisyon sistemleri üzerinden konuşma kanalları tahsis edilerek yapılmaktadır. Bu tez çalışmasında geliştirilecek uygulamalar ile müşterilerin IVR sistemine internet üzerinden erişimleri sağlanacak ve müşteri telefonu, sunucu, IVR sistemi arasında SIP protokolü kullanılmaktadır. Bu tez uygulamasında müşteri ile birlikte müşteri temsilcilerinin de mobil olması sağlanmaktadır.

Günümüzde çağrı merkezleri tez uygulamasında kullanılan sisteme benzer sistemler kullanılmaktadırlar. Ancak bu çağrı merkezleri çağrıları noktadan noktaya tahsis edilmiş hatlarla taşıdıklarından dolayı ara bağlantı yapabilmek için gerek süre açısından gerekse de maddi açıdan önemli ölçüde kayıp yaşamaktadır. Bunun yanı sıra müşteri temsilcilerini de kurmuş oldukları sistemle aynı konumda bulundurmaları gerektiğinden dolayı masraf kalemleri daha da yükselmektedir. Bu sistemde maliyetlerin azaltılması, riskin bölüştürülmesi, iş veriminin artırılması, konum bağımlılığının ortadan kaldırılması ve ara bağlantı için gerekli hattın tahsisi sırasında kaybedilen sürenin kazanılması amaçlanmaktadır.

1.2. Tezin Kapsamı

Bu tezde yüksek maliyetler harcanarak kurulan çağrı merkezlerinde görülen eksikliklerin çok daha düşük maliyetlerle nasıl giderilebileceğini, müşteri temsilcilerine ve müşterilere nasıl konum bağımsızlığının sağlanacağını, risk bölüştürmesinin nasıl yapılacağını, insan gücünün geri kazanımının nasıl sağlanacağını anlatmakta ve etkin çözüm önerileriyle bu etken faktörlerin nasıl giderilebileceğini savunmaktadır. Çözüm önerilerinin sunumunun yanı sıra bu önerilerin uygulamada nasıl kullanılacağını da anlatıldığı bu tez çalışmasında tezde anlatılan kadarı ile bir çağrı merkezi sisteminin de nasıl kurulduğunu göstermektedir. İhtiyaç duyulan tüm yazılımlarında geliştirilmesi bu tezin kapsamındadır. Kıyaslamalar sayesinde hangi öğenin nerede daha etkin olduğunun belirlenmesi sağlanmıştır.

1.3. Tezin Hedefleri

Konum bağımlılığının ortadan kaldırılarak çağrı merkezi hizmetinin nasıl verilebileceğinin hem teknik olarak hem de bilimsel açıdan yapılabilirliğinin anlatılması hedeflenmektedir. Aynı zamanda uygulamalarla da yapılabilirliğin söz de kalmayarak uygulamada da etkin bir şekilde nasıl çalıştığının gösterilmesi planlanmaktadır.

1.4. Kullanım Alanları & Avantajları

Bu tez çalışmasında en çok önem verilen değerlerden bir tanesi de tezin kullanım alanlarının varlığıdır. Yapılan araştırmalar, incelemeler, karşılaştırmalar ve uygulamalar günümüzde aktif olarak çalışmakta olan çağrı merkezlerinde ki işleyişi de kapsamakta ve bu işleyişte ne gibi düzenlemeler yapılarak daha etkin kullanılmalarının sağlanması anlatılmaktadır. Bu tez uygulamasının kullanılacağı aktif kullanım alanlarına örnek olarak, operatörlerin çağrı merkezlerini ya da bilinmeyen numaralar servislerini verebiliriz. Bunun yanı sıra satış sonrası destek vermek isteyen KOBİ'ler, telefon aracılığıyla sürekli bilgi alışverişi yapan okullar, garanti sorgulama servisleri, randevu servisleri gibi hizmetlere sahip kuruluşlar da bu tez çalışmasının kullanım alanları arasında gösterilebilir.

Tezin yapılışında belirlemiş olduğumuz amaçlar aynı zamanda tezin avantajları olarakta görülebilir. Konum bağımlılığının ortadan kaldırılması sağlanmış ve böylece personel ve zaman sarfiyatından kazanım sağlanmıştır.

Çağruların internet üzerine taşınmasıyla beraber servis verilebilirlik alanının genişletilmesi sağlanmıştır. Aktif çağrı merkezlerinde çağrılar anlaşma yapılan operatörle çağrı merkezi arasında kurulan özel hatlar üzerinden taşınmaktadır. Eğer bir müşteri, ilgili operatöre ait bir hatta sahip değilse çağrı merkeziyle irtibat kuramayacaktır. Yapılan bu tez çalışmasında çağrı taşıma ortamının dünya da yaygın bir şekilde kullanılmakta olan ve en büyük ağ olan internet üzerine taşınması sağlanmış ve böylece servis alanının sadece anlaşma yapılan operatörle sınırlandırılması ortadan kaldırılmıştır.

Çağruların internet üzerine taşınmasıyla beraber konum bağımlılığının dar bir alandan oldukça geniş bir alana taşınması sağlanmıştır. Bu özelliği altyapı olarak kullanarak, personelin de sisteme buldukları ortamdan bağlanmaları sağlanmıştır. Böylelikle personellerin farklı konumlarda olmaları sağlanmış ve yaşanabilecek herhangi bir bölgesel afet durumunda birden fazla konumda personel barındırıldığından dolayı risk bölüştürmesi de gerçekleştirilmiştir.

KOBI'ler için oldukça masraflı olan çağrı merkezi kurulumları yerine küçük çaplı çağrı merkezi hizmeti verilebilirliğinin sağlanması, sesli yanıt sistemi sayesinde arayan herhangi bir müşterinin isteklerinin kayıt altına alınması da sağlanmıştır. Bu da KOBI 'nin geliştirmiş olduğu sistemlere bağlı olarak vermiş olduğu hizmetlerinin sürekliliğinin artırılmasına katkıda bulunmaktadır.

Taşınabilir sistemler üzerine inşa edilen bu yapı sayesinde oldukça kolay bir şekilde sistemlerin yer değiştirmelerine de olanak tanınmıştır. Konum bağımlı çağrı merkezlerinde, operatörlerle ilgili çağrı merkezi arasında yapılan ses transferinin tahsis edilmiş olan noktadan noktaya hatlar üzerinden yapılmasından dolayı sistemlerin yer değiştirme olasılığı oldukça zordur. Ancak bu tez çalışmasında konum bağımlılığının ortadan kaldırılması sağlanmıştır ve bu sayede taşınabilirliğin de önü açılmıştır.

2. ÖZEL SANTRALLER (PBX)

2.1. Genel Bakış

PBX; Bir kurumun her bir telefonu ayrı ayrı kiralayıp ekstra bedel ödemesine gerek duymadan birden çok hattı karşılamaını sağlayabilen özel anahtar devreli telefon sistemleridir. Bu tür sistemlerde, şirkete bir hat grubu (trunk) tahsis edilir. Bu hat grubunu karşılayan sistemde dışarıdan içeriye, içeriden dışarıya doğru çağruların yönlendirilmesi sağlanır. Bu hizmet sadece çağrı ile sınırlı değildir. Bunun yanı sıra sesli posta, çağrı yönlendirme gibi farklı hizmetlere de olanak sağlayabilmektedir.

100 adet telefona ihtiyacı olan bir firma ele alındığında, eğer bu firma her bir telefon için bir hat çektirirse, kullansa da kullanmasa da her halükarda 100 hat için belirli sabit bir bedel ödemesi gerekir. Bu hatların hepsinin aynı anda kullanılma olasılığı göz önüne alındığında, bu tür bir masraf, şirketler açısından oldukça maliyetlidir. İşte tam burada PBX sistemler devreye girer. 100 hattın aynı anda kullanılma olasılığı göz önüne alındığında ve aynı anda maksimum %10 hattın kullanılabileceği düşünüldüğünde, şirkete 10 hat yeterli gelmektedir. Peki bu 10 hattı herkesin kullanmasını nasıl sağlanır ? PBX sistemler sayesinde içeriden dışarıya, dışarıdan içeriye yapılacak çağrılara verilecek rotalar sayesinde bu 10 hattı hem gelen çağrılarda hem de giden çağrılarda kullanabilir ve bu sayede şirkete maddi açıdan ekstra yük getirecek olan +90 hat masrafından kurtulunabilir. Bu sadece PBX sistemlerinden özelliklerinden bir tanesidir. Bunun yanı sıra PBX sistemler birçok farklı özelliklere de sahiptir.

2.2 Özel Santral Türleri (PBX Türleri)

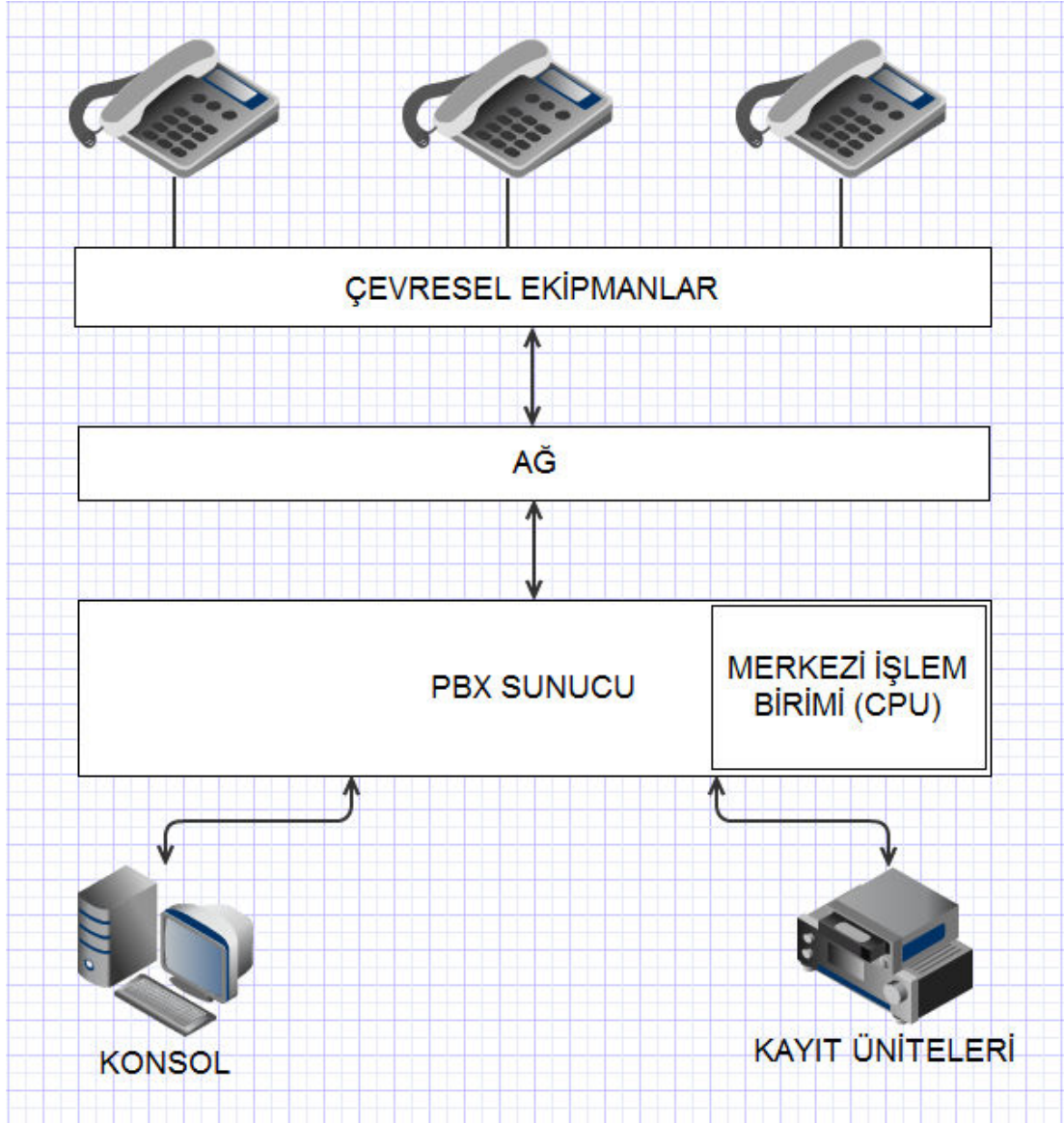
PBX sistemlerde iki tür mevcuttur. Bunlardan birisi analog pbx diğeri ise dijital pbx türleridir.

Analog PBX sistemler: Analog sistemler bir çağrıyı karşılayabilmek için analog bileşenler kullanmaktadır. Arayan kullanıcı telefonu kaldırdığı anda genellikle bir bilgisayar olan PBX sunucusuna bir giriş/çıkış sinyali gönderilir. Sinyal ana kontrole gönderildikten sonra, sistemden geriye bir arama tonu döndürülür. Daha sonrasında arama tonunu alan kullanıcı aramak istediği kişinin telefon numarasını tuşlar. Bu arama aşaması arayan kişinin konuşma yolunda dahili kablolarda yapılır. Bu dijitler (darbeli (pulse) ya da tonlamalı (tone)) kablolar üzerinden telefon sistemine doğru gönderilir.

Bundan sonra telefon sistemi kendi payına düşeni yapar ve mimariye uygun bir şekilde ürettiği isteği hat grubu (trunk) kartına iletir. Hat grubu kartı arayüz hizmeti görür ve merkez ofise doğru dış arama tonu için bir istekte bulunur. Daha sonra PBX hat grubu kartının arayüzünde arama tonunu karşılaması üzerine bu dijitleri tonlamayı veya darbeyi tekrar üreterek hat üzerinden merkez ofise gönderir.

Dijital PBX sistemler: Dijital bir PBX telekomunikasyon şirketine, Direk Dış Arama (Direct Outward Dial DOD) adı verilen bir hat grubu devresiyle bağlanırlar. Genel olarak kullanılan dijital PBX santraller T1 taşıyıcı sistem üzerinden işlem yapan 24 dijital hat grubuna sahip 256 dijital ses ve veri istasyonu sağlayan sistemlerdir. Bunun yanı sıra küçük PBX santrallerde kullanılabilirler. Ortalama küçük bir PBX santral 30 istasyon destekler fakat şirketin gelişen ihtiyaçlarına bağlı olarak kolaylıkla genişletilebilirler.

Tipik bir PBX santralde üç ana bileşen bulunmaktadır. Bu bileşenler; merkezi işlem birimi (CPU), ağ ve çevresel ekipmanlardır. [Şekil 2.2.1]



Şekil 2.2.1 Tipik bir santralin ana bileşenleri

Merkezi İşlem Birimi; kendi hafızasında sakladığı talimatları uygular, çevresel ekipmanların parçası olan, hat gruplarına ve PBX istasyonlara bağlanan, anahtarlama fonksiyonlarını kontrol eder. Operasyonlar, yönetim ve bakım fonksiyonları, bir sistem konsolundan uzaktan işlenir.

Sesi ve veriyi iletebilen bu özel telefonlar birçok özel çağrı işleme metotlarını da sağlayabilirler. Otomatik arama, Çağrı Yönlendirme, Çağrı Transfer, Hızlı Arama, Tekrar Arama gibi birçok metot kullanılabilir. Gün geçtikçe daha da artarak devam etmektedir. Örneğin gelişen dijital PBX sistemler sayesinde arayan numaranın hangi numara olduğu gösterilebilmektedir. Bununla beraber çağrıyı bekletme özelliği ya da özellikle kalite takibi yapabilmek isteyen, büyük kuruluşlarda hayli önem arz eden çağrının ses kaydının yapılabilmesi gibi birçok özellik dijital sistemlerle birlikte kazandırılmış ciddi gelişimlerdir.

Ağ; Yapılacak olan görüşmelerde iletimin sağlanacağı ortamı sağlar. Örneğin bir katılımcı diğer bir katılımcıyı aramak istediğinde yapılacak olan işlemlerin paketleri ağ üzerinden kullanıcılara iletilir, böylece bağlantı gerçekleştirilebilir.

Çevresel Ekipmanlar; Görüşme ortamının oluşturulmasında kullanılacak olan her türlü aygıt çevresel ekipman olarak adlandırılabilir. Görüşme esnasında arayan ve aranan tarafın kullanmış oldukları telefon, görüşmenin sağlanacağı hat grupları (trunks) gibi öğeler çevresel ekipmanlar için örnek verilebilir.

2.3 Özel Santral Sunucu Yazılımları

PBX sunucu olarak kullanabilmek için piyasa da birçok yazılım mevcuttur. Bu yazılımlardan biri seçilip bir bilgisayara kurularak PBX sunucu hizmeti verilebileceği gibi komple donanımsal PBX sunucuları da kullanılabilir. Peki ya niçin donanımsal PBX sunucuları tercih edilmeli? Donanımsal PBX sunucuları kendine özgün merkezi işlem birimi, dijital ses işleyicisi gibi modülleri sayesinde yazılımsal PBX sunucularına nazaran daha performanslı hizmet vermektedir. Şöyle düşünülebilir; Elinizde kişisel bir bilgisayara sahip olduğunuzu düşünün. Bu kişisel bilgisayara PBX sunucu yazılımlarından birisini kurun. Kişisel bilgisayarlarda ses dönüştürme, ses işleme gibi işlemleri yapması için herhangi bir DSP bulunmamaktadır. Bunun işlevini PBX yazılımları bilgisayarın işlemcisinden yapmaktadır. Peki bu durum donanımsal PBX sunucularında nasıldır? Donanımsal

PBX sunucularında bu kaynaklar, ses işlemek üzere özel olarak üretilmiş kaynaklardır. Ancak burada ki en büyük fark; donanımsal PBX sunucularında kullanılan bu kaynaklar ses işlemleri yapabilmek için özel işlemci, ses dönüştürme işlemleri için ayrı bir işleyici olan DSP'ler kullanılmaktadır. Eğer çok yüklü ses işlemleri yapılmıyorsa yazılımsal PBX sunucularının kullanılmasında herhangi bir sakınca yaşanmamaktadır. Buna örnek olarak aktif olarak hizmet vermekte olan ve ücretsiz yazılımsal PBX sunucularla hizmet veren çağrı merkezleri verilebilir.

Açık kaynak kodun yaygınlaşması ve geliştiricilerin açık kaynak kodlu projelerde yer almaya rağbet göstermeleri, açık kaynak kodlu PBX yazılımların gelişmesine önemli derecede katkıde bulunmuştur. Ücretsiz olmalarına karşın oldukça başarılı olan bu PBX sunucuları sayesinde küçük işletmelerin de çok düşük maliyetlerle PBX santraller kurmalarına olanak doğmuştur. Ayrıca ilgili firmalar tarafından geniş kütüphaneler oluşturulması da bu sektörün büyümesinde önemli rol oynamaktadır. Yaygın olarak kullanılan 3 PBX sunucu yazılımı mevcuttur. Tez projesinin uygulamasında Asterisk PBX Sunucu yazılımı kullanılmıştır ancak bilgilendirme olması açısından diğer PBX sunucularına da değinmekte yarar var.

Piyasada sıklıkla kullanılan yazılımsal PBX Sunucu yazılımları;

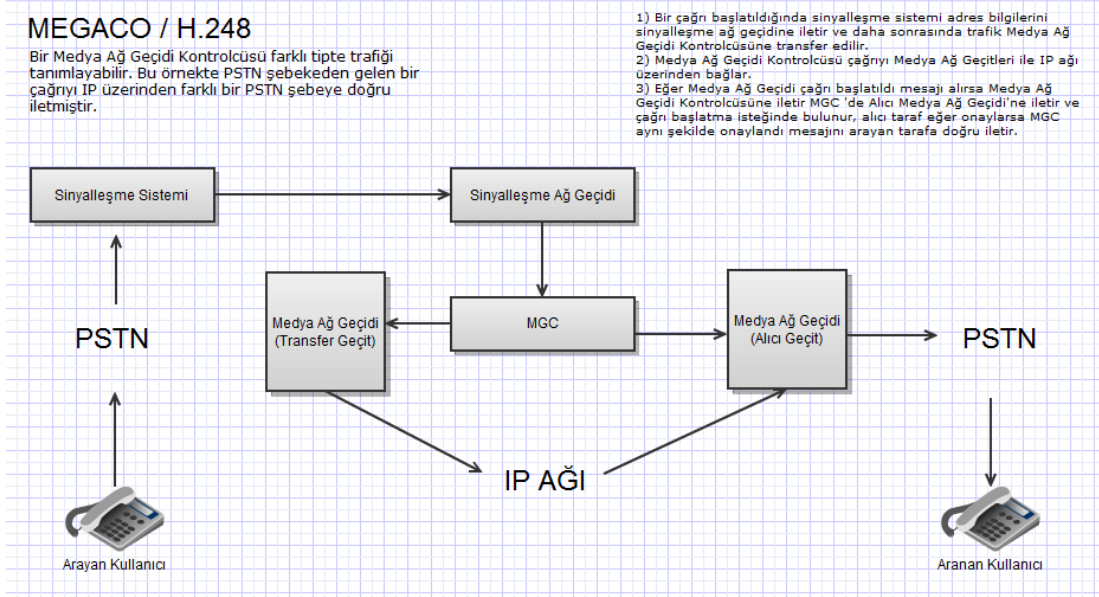
Asterisk PBX : 1999 yılında Digium firması tarafından üretilmiştir.Sesli posta, konferans görüşmeleri, interaktif sesli yanıt sistemi, otomatik çağrı dağıtımı gibi fonksiyonları desteklemektedir. Asterisk'in kendine ait belirlemiş olduğu AGI (Asterisk Gateway Interface) sayesinde çağrı akış planları oluşturulabilmektedir.

Free PBX: Asterisk'i kendisine taban olarak almış, kullanıcı arayüzüne sahip bir PBX Sunucu yazılımıdır. Uzman kullanıcılar tarafından kullanılabilen konsol ekranlarını, son kullanıcılarında rahatça yönetmesini sağlamak amacıyla kullanıcı arayüzleri geliştirmişler ve bunda da oldukça başarılı olmuşlardır. Kendi sitesinde belirtilene göre 5.000.000 üzerinde yüklenme, 500.000 üzeri aktif telefon sistemlerinde kullanılma oranlarına sahipler.

Free Switch: Anthony Minessale önderliğinde Free-Switch grubu tarafından üretilmiştir. C dilinde yazılmıştır ve Windows, Mac OSX, Linux, *BSD ve diğer Unix işletim sistemleri üzerinde çalıştırılabilmektedir.

3. MEDYA GEÇİŞ KONTROLÜ (MEGACO)

Megaco/H.248, yeni gelişen ip ağlarıyla var olan genel aktarmalı telefon şebekeleri arasında ses, faks veya multimedya iletişimin yapılabilmesini sağlayan bir standarttır. Megaco/H.248 analog hatlar ve telefon arayüzleri, internet telefonları, atm arayüzleri, anons sunucuları ve benzeri aygıtlar gibi ağ geçidi uygulamalarının tek bir yaklaşımla kontrolünün yapılmasını sağlamaktadır. Herhangi bir uç-seviye çağrı kontrol mekanizmasına bağlanmamıştır bu sayede SIP protokolüyle veya H.323 protokolüyle kullanılabilir. Medya ağ geçiti kontrolleri birbirleriyle noktadan noktaya haberleştiklerinde SIP veya H323 protokolünü kullanarak, medya ağ geçitleriyle haberleşirken de Megaco/H.248 kullanarak haberleşirler. Sinyalleşme lojiği medya ağ geçidi kontrolünde, medya lojiği ise medya ağ geçidinde bulunur. Bir medya ağ geçidi kontrolcüsü birden fazla medya ağ geçidini kontrol edebilir. Medya ağ geçitleri çok fazla komplike değıllerdir, bu sayede de fiyat olarak uygundur. Son kullanıcı ile ilişkilendirildiklerinde entegrasyonu kolay olsun diye medya ağ geçidi kontrolcüsünde özellik değışiklikleri yapılabilir. Medya ağ geçidi kontrolcüsüne çağrı iletme, çağrı yönlendirme, çağrı tutma, transfer etme ve konferans yapabilme gibi çağrı kontrol yetenekleri ve özellikleri entegre edilmiştir. Diđer medya ağ geçidi kontrolcüleriyle ya da genel aktarmalı telefon şebekeleriyle iletişim kurabilmek için uç seviyenin protokollerini de kendisinde bulundurabilir. Örneğin genel aktarmalı telefon şebekesinde sinyalleşme için SS7 kullanılıyorsa, gerekli iletişimin sağlanabilmesi için medya ağ geçidi kontrolcüsü tarafında da SS7 sinyalleşme kullanılarak haberleşme sağlanabilir. Megaco/H.248 mimarisi [Şekil 3.1] 'de gösterilmektedir.



Şekil 3.1 MEGACO / H.248

Megaco; sonlandırmalar ve içerikler olmak üzere iki temel elemandan oluşur. Sonlandırmalar 'da kendi içinde iki türe ayrılır. Bunlardan birincisi fiziksel sonlandırmalardır. Fiziksel sonlandırmalar; analog veya isdn hatlar veya SIP cihazlarına yapılan IP bağlantılar gibi ağ geçidinde soketlerin karşılanmasıdır. Devam etmeyen sonlandırmalar ise çağrı esnasında kurulur ve sadece çağrı devam ettiği sürece kullanılır. Bir RTP paket akışı devam etmeyen sonlandırmalar için örnek verilebilir.

Normal bir oturum içerik başına iki adet sonlandırmaya sahiptir ama konferans görüşmelerinde bu sayı çok daha fazlası olabilir. Medya ağ geçit kontrolü altındaki medya geçidi ilk sonlandırma eklendiğinde içeriği oluşturur ve isimlendirir ve oturum süresince aktif tutar. Oturum sonlandığı zaman ilgili içeriği de sonlandırır. Medya ağ geçidi ağır yük oluşturan koşullar için veya acil aramalar için içerikleri önceliklendirebilmektedir. Bir sonlandırma birden fazla sayıda akışa sahip olabilir, bu durumda içerikte çoklu içerik olabilir. Örneğin bir video akışı ele alınsın. Bir video akışında aynı zamanda ses ve görüntü akışı sağlanmaktadır. Bunun yanı sıra veri akışı da olabilmektedir. Medya geçitlerinin diğer bir özelliği de bir içerikte yapabilecekleri sonlandırma sayısıdır. Örneğin sadece noktadan-noktaya bağlantı kurma iznine sahip olan bir medya ağ geçidi her bir içerik başına maksimum iki sonlandırma gerçekleştirebilir.

Megaco/H.248 'te içeriklerin, sonlandırmaların ve kontrol sinyalleşmelerinin sağlanabilmesi için bazı komutlar kullanılmaktadır. Komut yapısı kolay ve işlemlere dönüştürülmüş yapıda olduklarından dolayı mesaj başlıklarında ki yükü düşürmektedir. Protokol 7 adet temel komuta sahiptir. Bunlar;

- 1- Add : Bir içeriğe sonlandırma ekler.
- 2- Subtract : İçerikte bulunan sonlandırmayı kaldırır.Son sonlandırma da kaldırıldıktan sonra içeriğin serbest kalmasını sağlar.
- 3- Move : İçerikler arasında sonlandırmaların taşınmasını sağlar.
- 4- Notify : Medya ağ geçidinin bilgilendirilmesini sağlar. Örneğin uç kullanıcının telefonu kapatması durumunda bu konudan ağ geçidini haberdar eder.
- 5- Audit
- 6- ServiceChange

Megaco için paket tanımı; düzenli karakteristiktan oluşan, sonlandırmalar için özellik seti şeklindedir. Örneğin bir analog döngü olaylarıyla (açık-kapalı durumu, çalıyor durumu) birlikte bir pakete sahiptir. Medya ağ geçidi kayıp paketler, operasyonel ölçümler gibi istatistiksel bilgileri saklar ve bunları medya ağ geçidi kontrolcüsüne raporlar.

Bir çağrı oturumu bir dizi mesaj serisi içermektedir. Katılımcılardan bir tanesi diğer bir katılımcıyı aradığında Medya Ağ Geçidi sinyali yakalar ve Medya Ağ Geçidi Kontrolcüsünü durum hakkında bilgilendirir. Bu bilgilendirme sonrasında Medya Ağ geçidi kendi içerisinde bulunan veritabanında giriş kontrolüyle ilgili metotları çalıştırır. Örneğin çağrıyı başlatan kullanıcının giriş izni var mı gibi kontrolleri yapar. Medya Ağ Geçidi Kontrolcüsü eğer kullanıcının izni varsa medya ağ geçidine arama yapan kullanıcıya arama tonu göndermesiyle ilgili ve aranan dijitalerin toplanmasıyla ilgili talimatlar verir. Tuşlama tamamlandığında Medya Ağ Geçidi tuşlanmış olan dijitaleri yönlendirme işleminin yapılabilmesi için Medya Ağ Geçidi Kontrolcüsüne gönderir. Bu işlem sonucunda aynı Medya Ağ Geçidi tarafından kontrol edilen bir başka istasyonla veya farklı bir TDM veya IP paketlemeli telefon hatlarıyla uzaktaki bir Medya Ağ Geçidiyle ya da Genel Aktarmalı Telefon Şebekesiyle bağlantıyı kurabilir. Daha sonrasında Medya Ağ Geçidi Kontrolcüsü aranan tarafta ki Medya Ağ Geçidi'ne telefon çalma sesi vermesi

ve çağrıyı başlatan tarafta ki Medya Ağ Geçidi'ne de çalma sesini döndürmesi talimatlarını verir. Oturum sağlandığı zaman da Medya Ağ Geçidi Kontrolcüsü zamanlamayı başlatır. Taraflardan herhangi biri çağrıyı sonlandırdığında veritabanında ilgili istasyonun kaydı boş konumuna çekilir.

4. GERÇEK ZAMANLI İLETİM PROTOKOLÜ (RTP)

4.1. Genel Bakış

Gerçek Zamanlı İletim Protokolü paket anahtarlama ağılarda gerçek zamanlı veri (video ya da ses akışı) transferinde kullanılmaktadır. SIP protokolü kullanılarak başlatılacak çağrılarda ses akışı RTP aracılığıyla yapılacaktır. Öncelikli olarak çok yönlü iletişim için tasarlanmıştır ancak tek yönlü iletimde de kullanılabilir. RTP, UDP ve IP 'nin üstünde bulunur. RTP paketlerinde kullanılan en önemli iki bilgi sıra numarası ve zaman bilgisi alanlarıdır. Sıra numarası bilgisi paketlerin karşı tarafa doğru sıralamayla ulaşip ulaşmadığının, zaman bilgisi alanı da paketlerin varış zamanlarının bilgisini tutar. [Tablo 4.1.1]

Tablo 4.1.1 RTP Paket Şeması

Versiyon		IHL		Servis Tipi		Toplam Uzunluk	
Tanımlama				Bayraklar		Parça Ofseti	
Yaşam Süresi			Protokol		Başlık Sağlaması		
Kaynak Adres							
Hedef Adres							
Seçenekler						Takviye	
Kaynak Port				Hedef Port			
Uzunluk				Sağlama			
V=2	P	X	CC	M	PT	Sıra Numarası	
Zaman Bilgisi							
Senkronizasyon Kaynak Tanımlayıcısı (SSRC)							

RTP internet telefonu gibi medya talebi istendiğinde kullanılır. Bir veri bölümü ve bir de kontrol bölümünden oluşur. Kontrol bölümü Gerçek Zamanlı Kontrol Protokolü (RTCP) olarak adlandırılır. Herhangi bir zaman diliminde RTP göndericisi aynı anda yalnızca bir tip taşıma yükü (payload) gönderebilir, buna rağmen örneğin ağ sıkışıklığını ayarlamak için iletim sırasında taşıma yükü tipini değiştirebilir.

4.2 RTP Güvenliği

Bir sinyalleşme esnasında RTP ses paketlerinin karşı tarafa iletiminden sorumludur. Paket numaralamaya ve zamanlamayı takip ederek yük taşımayı gözlemler. Ses iletimi gibi gerçek zamanlı iletim sağlayan uygulamalarda bu oldukça önemlidir. Fakat RTP gerçek zamanlı iletimin servis kalitesini garantilemez. Bunun yerine RTP alt katmanlarındaki servislerine güvenir.

VOIP sistemlerde sinyalleşme bazlı saldırı tipleri bulunmaktadır. Bu saldırı tiplerinden bir tanesi de medya iletimindeki açıkları baz alan RTP ilişkili saldırılardır. Saldırganlar alıcı tarafa boş (gereksiz) içerikli RTP paketleri gönderebilirler. Bu paketler kendilerine ait taşıma yükü ve başlık bilgilerine sahip olan rastgele numaralandırılmış içeriklerle hedefe doğru saldırı da kullanılırlar. Bu tip bir saldırı da alıcı taraf gelen paketleri anlamlandırmaya çalışacak ve bu iş için belli bir işlem gücü harcayacaktır. Bundan dolayı da çok daha fazla işlem yapacak olan alıcı cihazın çökmesine ya da en ucuz kurtulmayla ses kalitesinin bozulmasına neden olabilecektir. Bu tür bir atağı yakalamanın en iyi yolu gelen paketlerin sıralama numaralarının takip edilmesi ve eğer ardışık sırada gelmiyorlarsa bir saldırı olarak değerlendirilmesi ve koruma moduna geçilmesi gerekir. [5]

RTP paketlerinin korunması için RTP'nin bir uzantısı olan Güvenli RTP (SRTP) geliştirilmiştir. SRTP ile alıcı ve gönderen arasında yapılacak olan eş-zamanlı veri transferi sırasındaki RTP paketlerinin temel güvenlik önlemleri sağlanmaktadır. Bu güvenlik önlemleri gizlilik, mesaj kimlik doğrulaması ve Gerçek Zamanlı Transfer Kontrol Protokolü ile ilişkilendirilmiş olan RTP trafiği için yanıtı korumasıdır.

5. GERÇEK ZAMANLI KONTROL PROTOKOLÜ

Gerçek Zamanlı Kontrol Protokolü, RTP protokolü birlikte çalışması için tasarlanmış bir protokoldür. Bir RTP oturumunda, istemciler veri iletiminin kalitesinin ve üyelik bilgilerinin geri bildirimlerinin iletilmesi için sunucuya periyodik olarak RTCP paketleri gönderirler. Kontrol bilgisi taşıyan RTCP paketleri 5 tiptedir. Bunlar;

- 1- Alıcı Raporu : Alıcı raporu veri iletimi hakkında alınan en yüksek paket numarası, paket kayıpları, varış titremeleri ve istemci ile sunucu arasında ki dönüş zamanı gecikmelerini hesaplamak için gerekli olan bilgileri içerir.
- 2- Gönderen Raporu : Alıcı raporu bilgilerini, medya senkronizasyonu hakkında bilgi, kümülatif paket sayacı ve gönderilen byte sayısı gibi bilgileri içerir.
- 3- Kaynak Tanımlama Öğeleri : Kaynakları tanımlayacak bilgileri içerir.
- 4- Güle güle (Bye) :Katılımcıların bitirdiğini gösterir.
- 5- Uygulamaya Özel Metotlar : Yeni uygulamalar veya yeni özelliklerin geliştirilmesi gibi tecrübeleri sağlar.

Bu kontrol bilgileri sayesinde RTCP şunları sağlar:

Qos İzleme ve Sıkışıklık Kontrolü: Sunucular, istemcilerden gelen geri bildirimlere bağlı olarak iletimi ayarlayabilirler. İstemciler sıkışıklığın yerel mi yoksa bölgesel mi yoksa evrensel mi olduğunu belirleyebilirler. Ağ performansı böylece çok yönlü dağıtımdan gelen raporlara bağlı olarak ölçülebilir.

Kaynak Kimlik Saptaması: RTP veri paketlerinde, kullanıcılar için elverişli olmayan 32-bit rastgele üretilmiş tanımlayıcılar bulunur. Kaynak tanımlama paketleri kullanıcı adı, telefon numarası, e-posta adresi gibi metinsel bilgiler içerir.

Medya Eş zamanlaması: Ses ve Video için uç senkronizasyonlarda kullanılır.

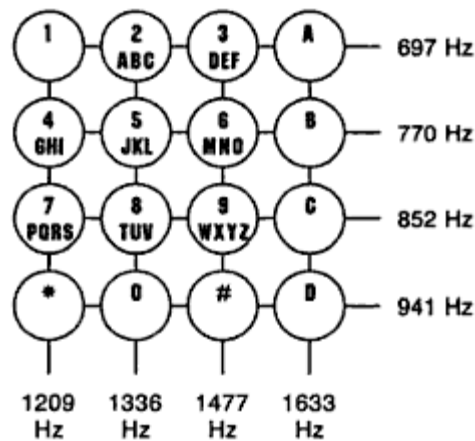
Kontrol Bilgisi Ölçeklendirmesi: Katılımcıların sayısı arttığında aşamalar trafiği kontrol ederek ağ kaynaklarının yoğun kullanımını engelleyebilir.

6. ÇİFT TONLU ÇOKLU FREKANS KODLAMA (DTMF)

DTMF; Ses frekans bantları içinde bulunan analog telefon hatları üzerinde bir telefon seti ile diğer bir iletişim aygıtının telekomünikasyon sinyalleşmesinde kullanılır. Bir telefonun tuş takımında bulunan 16 dijitin tonlama kombinasyonlarının şifrenmesini sağlayan, ses iletişimde ki medya akışında kolaylıkla algılanabilecek ve üretilebilecek standart bir mekanizmadır.

DTMF tonları kesin tanımlanmış iki tip tekil kombinasyon dizisinden oluşmaktadır. Bu tipler 4 yüksek tonlama ve 4 düşük tonlamadır. Görüşme esnasında görüşmede bulunan farklı medya kaynaklarının (insan sesi gibi) kazara bu kombinasyonlardan birini oluşturması, bir düşük ve bir yüksek tonlama kullanıldığında dijitlerin sağlıklı algılanmasını sağladığından dolayı pek mümkün olmayan bir olaydır.

Bu tonlamaların bir tuş takımına entegrasyonu oldukça rahat yapılabilmektedir. Aşağıdaki örnekte de göreceğiniz üzere bir tuş takımının her kolonu dörtlük bir yüksek-frekans tonlamayla, her satırı da dörtlük bir düşük-frekanslı tonlamayla ilişkilendirilmiştir.



Şekil 6.1 DTMF tuşlama frekans gösterimi

DTMF tonlamada yaygın olarak ilk 3 kolon kullanılmaktadır. A'dan D'ye etiketlenmiş olan son kolon ise askeri tonlamalarda kullanılmaktadır. Bir çok telefonun tuş takımında bu son kolonda ki tuş dizisi bulunmamaktadır.

Bir telefon ağında yaygın olarak şu tip tonlamalar kullanılır:

Arama Tonu : Arama tonu yeni bir çağrının başlayacağını ve yeni bir komutun gelebileceğini bildirir. Başlangıç bağlantı durumuyla ilişkilendirilir.

Ödeme Tonu: Ödeme tonu kredi kartı veya arama kartı gibi bilgilerin gelmesinin beklendiği bir çağrı işlem türüdür. Gong tonu olarak ta bilinmektedir.

Meşgul Tonu: Çağrının yapıldığı hedef aygıtın o an için müsait olmadığını belirten tonlamadır. Çağrı durumu olarak ta bağlantı sağlanamadığından dolayı bağlantı durumu başarısız (fail) olarak ayarlanır.

Yeniden Düzenleme Tonu: Bir telefon sisteminde gerekli ağ arayüz aygıtı veya anahtarlama sistemi uygun olmadığında gelen çağrının bloklanmasını sağlar. Buna yanlışı çevrilen bir numara da sebebiyet verebilir. Meşgul tonunda olduğu gibi bağlantı sağlanamadığından dolayı bağlantı durumu başarısız olarak ayarlanır.

Özel Bilgi Tonlamaları: Çağrı kurulmasına geciktiren/engel olan bazı özel durumlar için kullanılan tonlama türüdür. Bu tonlamalarda her zaman için problemi tanımlayan önceden kaydedilmiş bir bilgilendirme mesajı verilmektedir. Bağlantı durumu olarak başarısız konumundadır. Aktif olarak kullanılan 4 özel bilgi tonlamaları şunlardır:

Boş Tonu : Aranana numara tahsis edilmemiş.

Önleme Tonu : Aranana numaraya yapılan tüm çağrılar önlenir. (Genellikle aranana numara değişmiş olur)

Devre Yok Tonu: Çağrı için uygun bir devre olmadığında verilen ton.

Yeniden Düzenleme Tonu: Numara eksik ya da yanlış çevrildiğinde verilen tonlama.

Geri Arama Tonu: Çağrı işleyicinin çağrıyı bir başka ağıta bağlamaya çalıştığında verilen ton. Arayan kişinin bir arama yaptığında çağrı cevaplanana kadar veya düşene kadar ki geçen sürede dinlemiş olduğu çalma sesidir. Bağlantı durumu olarak alarm (alerting) konumundadır.

Bip Tonu: Bip tonu genellikle sesli yanıt cihazlarında, sesli posta sistemlerinde veya farklı bir medya kayıt aygıtı sistemlerinde verilen tondur. Bip tonu arayan kişi için kaydın başladığını belirtmek amacıyla kullanılır. (Örnek: Şuanda yerimde yokum, lütfen mesajınızı bırakınız. (Bip Tonu verilir))

Kayıt Uyarı Tonu: Kısa bir 1400Hz tonudur. Konuşma esnasında kayıt yapılıyorsa verilen kayıt uyarısı tonu sıklıkla kanuni işlemlerle uğraşan kurumlarda kullanılır. Bunun yanı sıra bankalar ya da çağrı merkezleri gibi kuruluşlar tarafından da yaygın olarak kullanılmaktadır.

Tez projesinde yapılacak uygulama da IVR ile etkileşim için DTMF tonlama kullanılacak. IVR sistemi kullanan iletişim ortamlarında DTMF tonlamaların sağlıklı bir şekilde iletilmesi oldukça önemlidir. Bir çok IVR sisteminde, IVR tarafından soru sorulur ve buna uygun olarak kullanıcının tuşlama yaparak yanıt vermesi beklenir. Burada en önemli değer hangi tuşa basıldığının anlaşılmasıdır, süresi veya ne zaman basıldığı bilgisi pek önemli değildir. Ama başka bir IVR senaryosunda zamanlama da önem kazanabilir. Örneğin IVR kullanıcıya seçim yapması için bir liste okur ve bir zaman aşımı süresi belirler. Bu süre içerisinde herhangi bir tuşlama yapılmaması durumunda sistem kullanıcının görüşmesini sonlandırarak, kullanılan kaynakların boşa çıkarılmasını sağlayabilir.

Bu tür IVR sistemleriyle etkileşimli çalışabilmek için DTMF tonlarını yakalayabilmesi için bazı özel prosedürler geliştirilmiştir:

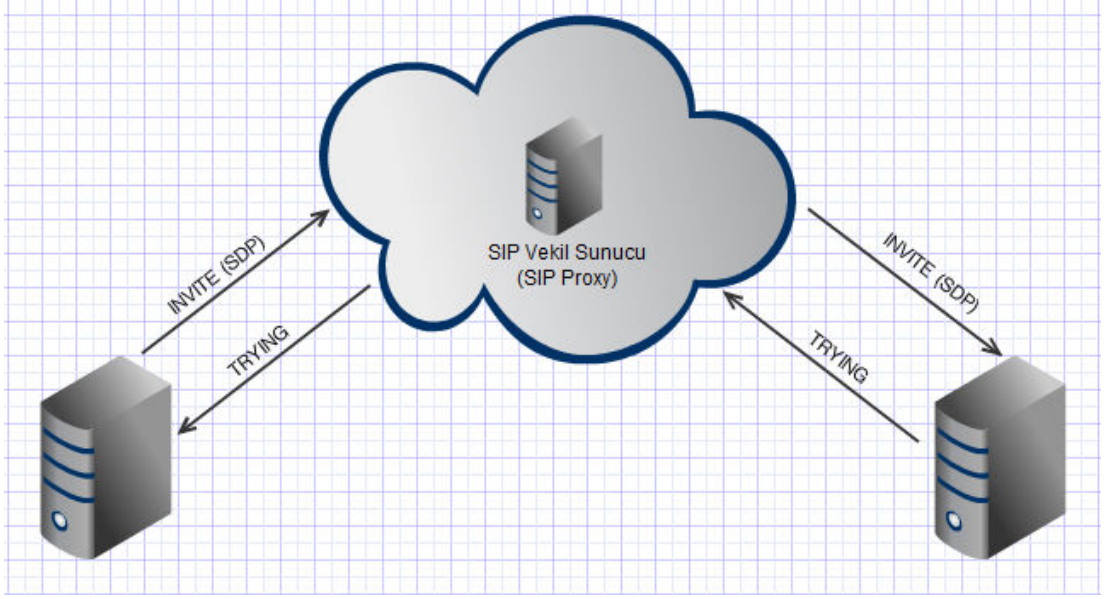
- H323 protokolü genellikle bu DTMF tonlamayı yakalayabilmek için sinyalleşme (H.245 Kullanıcı Girdi Bulgusu) kanalını kullanır. Anahtarlama metotlarını entegre etmeye ihtiyaç duyan uygulama sunucuları için oldukça etkili bir metottur. H.323v4 sürümünden bu yana eski yöntem alternatif olarak RTP kanalı üzerinden tamamen kodlanabilir formda DTMF tonlamanın iletiminde RFC2833 'ün kullanımına da olanak sağlanmıştır.
- SIP genel olarak iki metot kullanmaktadır. Bu iki metot; henüz yeteri kadar standartlaştırılmamış, BİLGİ ve UYARI mesajlarını baz alan bir sinyalleşme metodu ve RFC2833 'tür.

7. OTURUM BAŞLATMA PROTOKOLÜ (SIP)

7.1 Genel Bakış

SIP; IETF tarafından ortaya çıkarılmış multimedya iletişim ortamlarında sıklıkla kullanılan bir sinyalleşme protokolüdür. [1] İnternet üzerinden sesli ya da görüntülü görüşme gibi servislerin sinyalleşme altyapısını oluşturmaktadır. Bir uygulama katmanı kontrol prokolü olan SIP'te bir veya birden fazla katılımcı arasında oturum oluşturma, düzenleme ve oturumu sonlandırma gibi olanaklar sağlanmaktadır. Bu oturumlar internet üzerinden bir telefon görüşmesi ya da telekonferans gibi multimedya servisler olabilir.

Oturum yönetimi yapan ve katılımcılar arasında veri alışverişi sağlayan bir çok internet uygulaması bulunmaktadır. Bu uygulamalarda yaşanan en büyük sıkıntı katılımcılar arasında ki iletişimin sağlıklı bir şekilde çalışmasının sağlanmasındadır. Kullanıcıların ortam değiştirmeleri, birden fazla isim tarafından adreslenebilir olmaları, farklı türde medya aygıtları kullanıyor olmaları ortak bir oturum kurulmasında engel olabilmektedir. SIP, bu protokollerle, internet üzerinden katılımcıların birbirlerini bulabilmesini ve ortak işaretleşmeyle haberleşebilmesini sağlar. Oturum katılımcılarının konumlarını tespit edebilmek ve diğer özellikler için SIP kullanıcıların oturum açabilecekleri, oturum davetleri ve başka istekleri gönderebilecekleri vekil ağ sunucuları yapısı (Proxy servers) oluşturur. [Şekil 7.1.1] Bu sayede kullanıcıların konum bağımsız bir şekilde birbirleriyle iletişimlerini, birbirleri arasında veri iletişiminin sağlanmasını sağlar.



Şekil 7.1.1 Oturum Başlatma Örneği

Bu vekil sunucular; katılımcıların oturum açmalarını, bir kaynak katılımcının göndermiş olduğu isteğin hedef katılımcının o anki konumuna yönlendirilmesini, kullanıcıların kimlik bilgilerini kontrol ederek servislere erişimlerinin onaylanmasını, servis sağlayıcının çağrı yönlendirme ilkelerinin uygulanmasını ve kullanıcılara bu özellikleri kullanabilmesini sağlar.

SIP, multimedya iletişimin kurulmasında ve sonlandırılmasında 5 kriteri desteklemektedir.

- 1- Kullanıcı konumu: iletişim için kullanılacak olan son kullanıcı sisteminin belirlenmesi
- 2- Kullanıcı uygunluğu: aranan katılımcıyla iletişimin sağlanabilmesi için aranan katılımcının uygunluğu
- 3- Kullanıcı yetenekleri: Medya aygıtının ve aygıtın parametrelerinin belirlenmesi
- 4- Oturum kurulumu: arayan ve aranan katılımcının oturum parametrelerinin kurulması
- 5- Oturum yönetimi: servislerin başlatılması, oturumun kurulması, düzenlenmesi ya da sonlandırılması

SIP komple entegre bir iletişim sistemi değil. Komple bir iletişim mimarisinin kurulabilmesi için diğer bazı IETF protokollerinin de kullanılması gerekmektedir. Bir iletişim mimarisinin kurulabilmesi için genellikle kullanılan protokoller RTP (Real-Time Transport) , RTSP (Real-Time Streaming Protocol), MEGACO (Media Gateway Control Protocol), PSTN (Public Switched Telephone Network) ve SDP (Session Description Protocol) şeklindedir.

Kısaca bu entegre mimarinin bileşenlerini açıklamak gerekirse;

- RTP : Gerçek zamanlı iletimi sağlar.
- RTSP : Akışı olan medyanın ulaşp ulaşmadığının kontrolünü sağlar.
- MEGACO : İnternet protokollü ağların üzerindeki ve telefon ağlarının üzerindeki medya geçitlerinin kontrolünü sağlar.
- PSTN : Devre aktarmalı telefon ağları.
- SDP : Gerçek zamanlı içerik akışının başlangıç oturum bilgilerini (parametrelerini) tanımlar.

7.2 Oturum Tanımlama Protokolü (SDP)

Multimedya oturumların tanımlanabilmesi için IETF tarafından tasarlanmış bir protokoldür. SDP medya akışının oturum bilgilerini taşır ve böylece hedef katılımcının kurulum bilgileri öğrenmesi sağlanır. SDP oturumda ki medyanın kodlamasıyla ilgilenmez, sadece ilgili oturumun temel manada tanımlamasını belirtir. Herhangi bir iletim prokolü içermez. Bu sebepten dolayı da SDP ‘nin yaygın olarak kullanılışı, diğer protokollerin mesaj gövdesine gömülerek yapılabilmektedir. IETF RFC 2327 ‘de belirtildiğine göre SDP şu bilgileri içerir:

- Oturumun adı ve amacı
- Oturumun başlama zamanı
- Medya kapsamı
- Medyayı alabilmesi için gerekli olan adres, soket numarası ve biçimi

Gerekli SDP ifadelerini sağlayabilmek için başka bilgilerde içerebilir. Bir SDP oturumu bir çok metinden oluşabilir. SDP oturumlarında ki mesaj tipleri şu şekildedir:

{tip} = {değer}

Tip alanı sadece 1 karakterden oluşmaktadır ve küçük büyük harf duyarlılığına sahiptir. Değer alanı ise yapılandırılmış bir metin olabilir. Herhangi bir belirtim yapılmadığı sürece değer alanı da harf duyarlılığına sahiptir. Eşittir (=) işaretinin sağında ve solunda boşluk olmaması gerekmektedir.[2] SDP’ de farklı türde tanımlamalar olabilir. [Şekil 7.2.1] ‘de örnek bir SDP mesajı gösterilmektedir.

```
<----->
v=0
o=root 23722 23722 IN IP4 212.174.31.27
s=session
c=IN IP4 212.174.31.27
t=0 0
m=audio 19382 RTP/AVP 18 101
a=rtpmap:18 G729/8000
a=fmtp:18 annexb=no
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-16
a=silenceSupp:off - - - -
a=ptime:20
a=sendrecv
<----->
```

Şekil 7.2.1 SDP Mesajı

Tablo 7.2.1 SDP mesajı öge detayları

Alan	Tanımı	Z/İ (Zorunlu / İsteğe Bağlı)
V	Protokol versiyonu	Z
O	Başlatan ve oturum tanımlayıcıları {kullanıcı adı} {oturum numarası} {versiyon} {ağ tipi} {adres tipi} {adres}	Z
S	Oturum adı	Z
İ	Oturum bilgisi	İ
U	Tanımlamanın URI'si	İ
E	E-posta adresi	İ
P	Telefon numarası	İ
C	Bağlantı bilgisi	İ
B	Bant genişliği bilgisi	İ
Z	Zaman dilimi ayarları	İ
K	Şifreleme anahtarı	İ
A	Oturum özellikleri	İ

Tablo 7.2.2 Zaman dilimi öge detayları

Alan	Tanımı	Z/İ (Zorunlu / İsteğe Bağlı)
T	Oturumun aktif olduğu zaman	Z
R	Tekrar zamanı	İ

Tablo 7.2.3 Medya katmanı öge detayları

Alan	Tanımı	Z/İ (Zorunlu / İsteğe Bağlı)
M	Medya adı ve iletim adresi	Z
İ	Medya başlığı	Z
C	Bağlantı bilgisi	Z
B	Bant genişliği bilgisi	İ
K	Şifreleme anahtarı	İ
A	Oturum özellikleri	İ

7.3 Paket Yapısı

SIP mesajlaşma modeli http istek/yanıt işlem modeliyle benzerlerdir. İstemci isteğinden oluşan her işlemde sunucu da istekle ilgili metotlar çalışır ve en az bir tane olmak üzere istemciye yanıt döndürülür.[1] Bu mesajlar RFC-822'ye göre kodlanmış metin mesajlarıdır.

→ Mesaj Formatı:

Tüm mesajlar için genel formatı şu şekildedir;

- Başlangıç çizgisi
- Bir veya daha fazla başlık bilgisi
- Boş bir çizgi
- Mesaj gövdesi (isteğe bağlı)

Her satır bir CRLF ile bitmek zorundadır.

7.4 İstekler, Yanıtlar ve Durum Kodları

→ İstekler:

SIP 6 tip istek kullanmaktadır.[3] Bunlar;

- INVITE : Bir çağrı oturumunda bir servisin ya da kullanıcının çağrıya katılması için davette bulunulması.
- ACK : INVITE isteğine dönen sonuç cevabını kullanıcının aldığı doğrulanması.
- BYE : Görüşmenin sonlandırılması. (Bu mesaj arayan tarafından da aranan tarafından da gönderilebilir)
- CANCEL : Bekleyen aramaların iptal edilmesi. (Daha önceden kabul edilmiş ve kurulmuş olan çağrıyı sonlandırmaz)
- OPTIONS : Sunucu yeteneklerinin sorgulanması.
- REGISTER : Bir SIP sunucusu ile birlikte mesaj başlığında TO kısmında listelenmiş adresin kayıt edilmesi.

```
<--- Reliably Transmitting (NAT) to 212.79.96.68:5060 --->
SIP/2.0 200 OK
Via: SIP/2.0/UDP 212.79.96.68:5060;received=212.79.96.68
From: "905417263673" <sip:905417263673@212.79.96.68:5060>;tag=10968e5e82ed2f108afb000000000000
To: "5411824" <sip:5411824@212.174.31.25:5060>;tag=as3089ce63
Call-ID: 8947476C48BD11DF8E3300E0F5564950@212.79.96.68:5060
CSeq: 7023 INVITE
User-Agent: Asterisk PBX
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY
Supported: replaces
Contact: <sip:5411824@212.174.31.25>
Content-Type: application/sdp
Content-Length: 257
```

Şekil 7.4.1 SIP İstek Mesajı

→ SIP istek uzantıları:

- SUBSCRIBE : Kullanıcının bir olay tetiklendiğinde ilgili olaydan bilgilendirilmesini için gerekli katılımın sağlanması.
- NOTIFY : Katılmış bir olaydan dolayı bilgilendirilmenin sağlanması.
- MESSAGE : SIP aynı zamanda çevrimiçi mesajlaşmayı da sağlamaktadır. Kullanıcıların birbirlerine mesaj gönderebilmesi için kullanılır.
- INFO : Oturum esnasındaki bilgilendirmenin sağlanması. (Kullanıcı aktiviteleri gibi)
- SERVICE : SOAP mesajlarının taşınmasının sağlanması.
- NEGOTIATE : Güvenlik mekanizması ya da algoritmalar gibi çeşitli parametreler üzerinde anlaşmanın sağlanması.

- REFER : Mesajı gönderen kişinin, istekte sağlanan iletişim bilgilerini kullanarak bir üçüncü partiyle kontak kurmasıyla ilgili alıcı tarafın bilgilendirilmesini sağlar.

→ Yanıtlar:

SIP yanıtları durum kodu ve sonuç ifadelerinden oluşur. [3]

Durum Kodu : İstek sonrasında oluşan 3 basamaklı sayı.

Sonuç İfadesi : Gelen yanıtın metinsel tanımını içerir.

Durum Kodları;

- SIP 1xx : Bilgilendirme Yanıtları
- SIP 2xx : Başarı Yanıtları
- SIP 3xx : Yönlendirme Yanıtları
- SIP 4xx : İstemci Hatası Yanıtları
- SIP 5xx : Sunucu Hatası Yanıtları
- SIP 6xx : Evrensel Hata Yanıtları

```
<----->
aste-risk*CLI>
<--- SIP read from 212.79.96.68:5060 --->
ACK sip:11824@212.174.31.25:5060;transport=UDP SIP/2.0
From: "5396352686" <sip:5396352686@212.79.96.68:5060>;tag=7ade7c4f82ed2f1089
fe000000000000
To: "11824" <sip:11824@212.174.31.25:5060>;tag=as1e4dd547
Via: SIP/2.0/UDP 212.79.96.68:5060
Max-Forwards: 70
Call-ID: F29732EF48BC11DF8E3300E0F5564950@212.79.96.68:5060
CSeq: 7005 ACK
Content-Length: 0
<----->
```

Şekil 7.4.2 SIP Yanıt Mesajı

→ SIP Başlıkları:

SIP protokolü bir mesajında birden fazla mesaj başlığı içerebilmektedir. Bu mesajlar alıcı tarafın gelen isteği en iyi şekilde anlayabilmesini ve nasıl işlenmesi gerektiğinin bilgilerini sağlar. Ele alınacak olursa 3 tip SIP başlığından söz edilebilir.

Genel Başlıklar : Bazı başlıklar hem istekte hem de yanıtta kullanılabilirler. Bu tip başlıklar için genel başlıklar denilir. Bu tip başlıklar temel bilgiler içerirler. Örneğin FROM tanımlayıcısı mesajı gönderenin bilgilerini içerir.

İstek Başlıkları : Bu tip başlıklar sucunun isteği gönderen hakkında ya da istek hakkında bilgileri sağlarlar. Örneğin SUBJECT tanımlayıcısı açılan oturum konusunun metinsel tanımını sağlar.

Yanıt Başlıkları : Gönderilen isteğe döndürülecek olan yanıtın durum bilgisinde taşınamayan daha detaylı bilgilendirmenin taşınmasını sağlar. Örneğin RETRY-AFTER tanımlayıcısı, eğer kullanıcı o an meşgul ya da müsait değilse, müsait olduğunda bilgilendirilmeyi sağlar.

→ SIP Bileşenleri:

SIP mimarisi aşağıda bahsedilen bileşenleri içermektedir.

User Agent

Proxy Server

Redirect Server

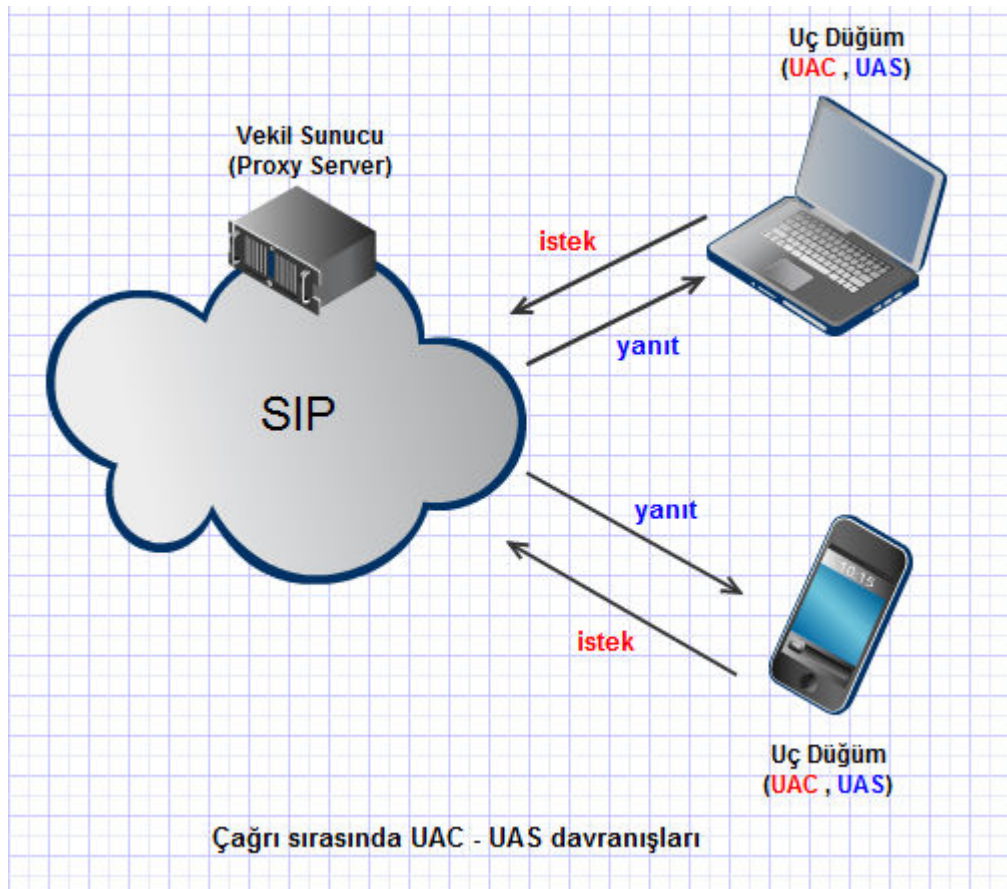
Registrar Server

User Agent: Kullanıcı aracısı (user agent), bir SIP oturumunda çağrı başlatabilen, karşılayabilen, sonlandırabilen bir uygulamadır. Bir kullanıcı aracısı bir çağrı başlatılmasında INVITE isteğini göndererek bir istemci gibi ya da istemciden BYE mesajını beklerken bir sunucu gibi davranabilir. Bir SIP oturumunda istemci tarafı bir cep telefonu, bir PDA ya da bir bilgisayar gibi bir son kullanıcı aracısı olabilir. Bir kullanıcı aracısı iki türlü davranabilir. Bu türler;

User Agent Client : Bir kullanıcı aracısı istemcisi yeni bir SIP isteği başlatan bir istemci uygulamasıdır. Bu uygulama SIP isteği gönderirken işlem süresince UAC gibi davranabilir. Eğer uygulama bir istekle karşılaşır, UAC işlemi yapabilmek için UAS gibi davranabilir. UAC tarafından işlenebilen istekler INVITE, ACK, OPTIONS, BYE, CANCEL ve REGISTER istekleridir.

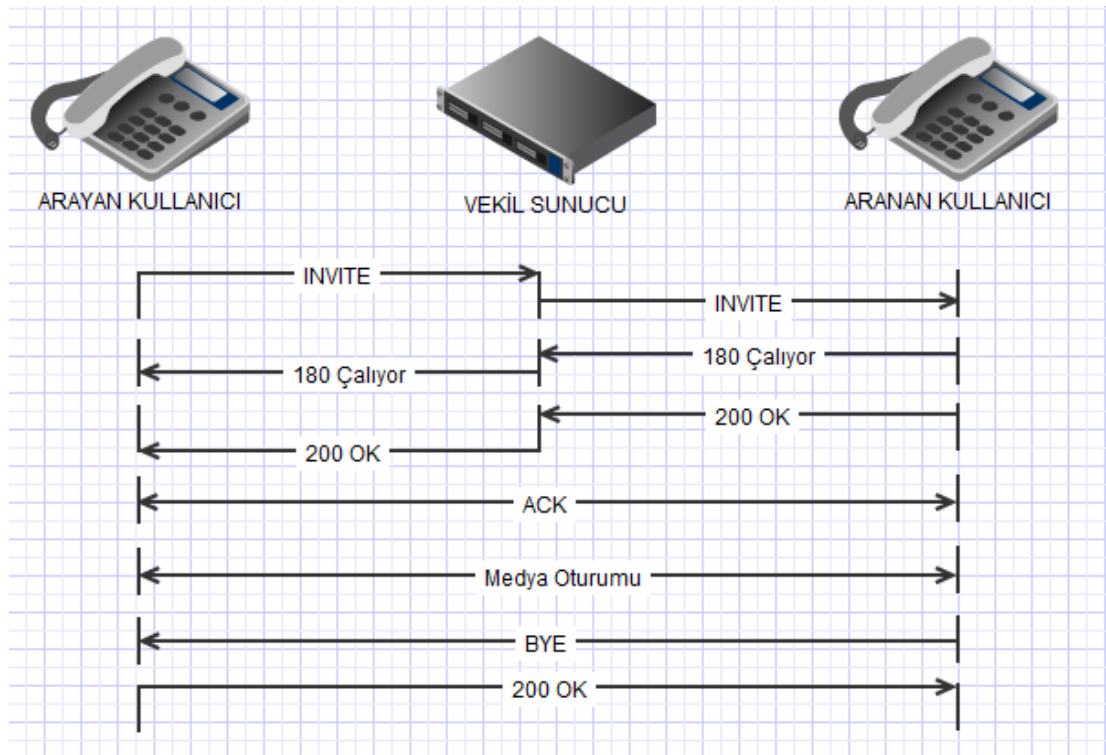
User Agent Server : Bir kullanıcı aracısı sunucusu istekleri karşılayıp işleme koyan ve işlem sonrasında yanıt döndüren bir sunucu uygulamasıdır. Bu uygulama sadece SIP yanıtı alma süresince UAS gibi davranır.

Bir SIP sonlandırıcısı hem UAC hem de UAS gibi davranabilir. Ancak bir işlem sırasında sadece bir tanesi gibi davranabilir. İsteği başlatan kullanıcı aracısına bağlı olarak ya UAC gibi ya da UAS gibi davranabilir.



Şekil 7.4.3 Çağrı sırasında UAC – UAS davranışları

Proxy Server: Bir SIP vekil sunucusu kullanıcı aracısından veya başka bir vekil sunucusundan gelen isteğin içeriğine bakarak isteğin yönlendirilmesini ya da yanıtlanmasını gerçekleştirir. SIP vekil sunucusu gelen SIP mesajlarını IP katmanına iletir. Bir vekil sunucusunun gelen isteği cevaplayabilmesi (örneğin bir sonraki aşamanın ne olduğunu belirleyebilmesi) için bir veritabanına ya da bir servise erişimi gerekmektedir. Kullanılacak olan servis ile vekil sunucusunda ki ara yüz SIP protokolü tarafından belirlenmez. Bir vekil sunucusu bir isteği işleyebilmek ve yanıtlayabilmek için birden fazla sayıda ve türde veritabanını kullanabilir. Bu veritabanları kullanıcının ve kullanıcıya dair bilgilerin (SIP kayıtları, durum bilgileri v.s.) barındırıldığı veritabanlarıdır. [Şekil 7.4.4] 'te görüldüğü gibi bir vekil sunucu arayan kullanıcıya hedef kullanıcının konum bilgilerini sağlayarak bir SIP mesajı takasçısı gibi davranır.



Şekil 7.4.4 SIP Oturum Başlatma

Bir vekil sunucusu taşımış olduğu SIP istek mesajını ulaştırmak için mesajın içeriğini anlamaya ihtiyaç duymaz. Anlamadığı her türlü mesajı NON-INVITE işlem modelini kullanarak değerlendirir. Bir vekil sunucusu gelen SIP mesajının başlık sıralamasını veya içeriklerini değiştiremez ya da silemez. Bir vekil sunucusuyla kullanıcı aracısı arasındaki ayırt edici farklar şu şekildedir:

- 1- Bir vekil sunucusu medya yeteneğine sahip değildir.
- 2- Bir vekil sunucusu gelen SIP istek mesajlarını kendi başına cevaplayamaz. Sadece kullanıcı aracısından gelen mesajları yanıtlayabilir.(CANCEL ve ACK istekleri istisna)
- 3- Bir vekil sunucusu mesaj gövdelerini ayrıştırıamaz. SIP başlıklarında belirtilen alanlara göre hareket eder.

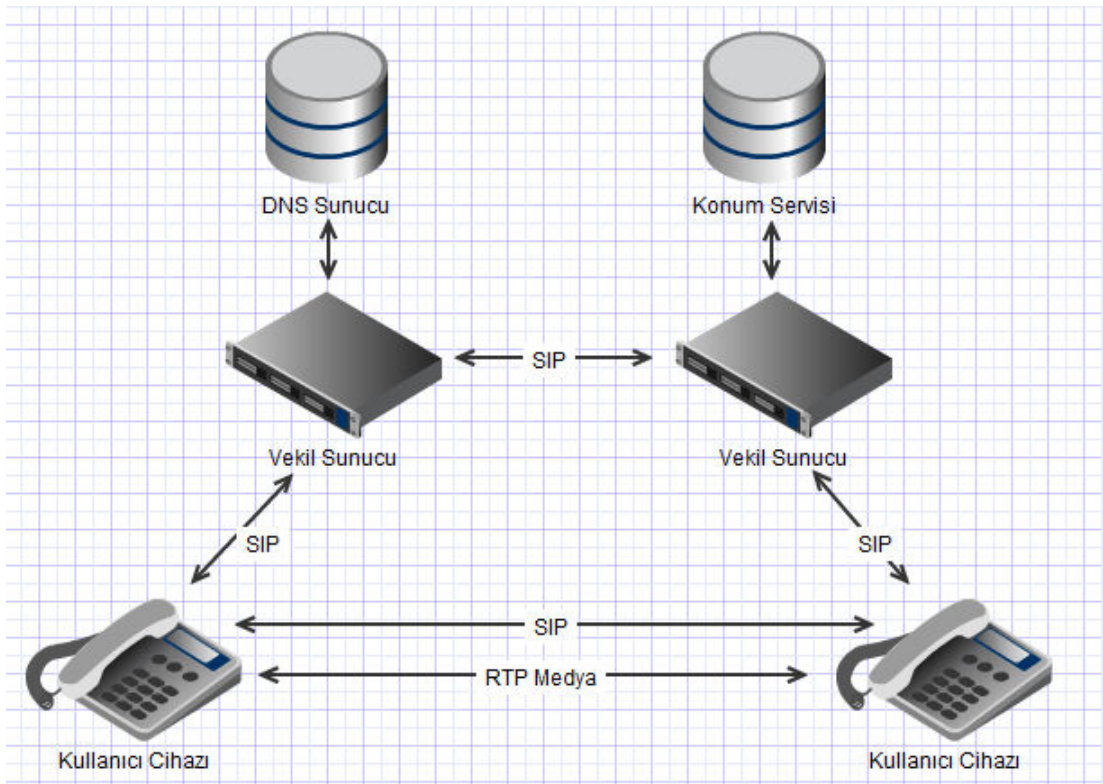
[Şekil 7.4.5] 'te görüldüğü gibi örnek bir SIP ağ topolojisi bulunmaktadır. Farklı etki alanında bulunan kullanıcı araçlarının kendi etki alanlarında bulunan vekil sunucuları kullanarak nasıl haberleştiklerini gösteren şekilde, her kullanıcı aracı mesajlarını yönlendirebilmek için kendi vekil sunucularını kullanacak şekilde tasarlanmışlardır. Bu vekil sunucular kullanıcıların oturum bilgilerini tutup, oturum doğrulamalarını yapıp, mesajlarının yönlendirilmelerini sağlayacak şekilde servis vermektedirler. Farklı etki alanında bulunan vekil sunucularının birbirlerinin konumlarını tespit edebilmeleri için Dinamik İsim Sunucusu (DNS) konulmuştur. Vekil sunucular arayan kullanıcıya bağlı olarak kendi içlerinde yönlendirme yapabilirler. Bu tip yönlendirmelerde bulunan vekil sunuculara dâhili vekil sunucu denilmektedir. Bu vekil sunucunun aynı zamanda kullanıcının aktif kayıtlı bilgilerine erişimi var ve arayan kullanıcıya isteği yönlendirebilir. Genellikle bir vekil sunucunun veya her iki sunucunun da RECORD-ROUTE başlık etiketi eklemedikleri isteklerde, SIP mesajları direk olarak iki kullanıcı arasında gönderilir.

Vekil sunucular iki konumda olabilirler. Bunlar:

- 1- Durumsuz (Stateless) : Durumsuz vekil sunucuları her bir isteği veya yanıtı sadece mesaj içeriğini baz alarak işleyebilirler. Bir mesaj ayrıştırıldığında, işlendiğinde, iletildiğinde veya yanıtlandığında mesaj ile ilgili herhangi bir bilgi depolanmaz. Durumsuz vekil sunucuları aynı mesajı tekrar iletmezler ve SIP zamanlayıcıları kullanmazlar.
- 2- Hazır (Stateful) : Hazır vekil sunucuları geçmişte yapılan isteklerin ve yanıtların geçmiş izlerini tutar ve ileride yapılacak isteklerde veya yanıtlarda kullanabilir. Örneğin hazır bir vekil sunucusu bir mesaj iletildiğinde

zamanlayıcıyı başlatır ve gönderilen isteğe belirlenen zaman diliminde yanıt gelmezse aynı isteği tekrar gönderir. Aynı zamanda hazır vekil sunucuları kullanıcı aracısı için güvenlik doğrulamaları kullanabilir.

Sıklıkla kullanılan vekil sunucu tipi Hazır (Stateful) vekil sunucu tipleridir. Hazır vekil sunucuları isteğe cevabın gelmesi halinde zaman dilimini görmezden gelerek devam eder. Örneğin hazır bir işlem vekil sunucusu yapmış olduğu INVITE isteğine 200 OK veya 404 Not Found cevabı alması durumunda zamanlayıcıyı sonlandırır ve böylece minimum masraflarla etkili bir şekilde çalışır. Hazır vekil sunucularının durumsuz vekil sunucularına göre diğer bir avantajlı yanı da birden fazla kullanıcıya gönderilecek aynı isteğin olması durumudur.



Şekil 7.4.5 Farklı etki alanında bulunan bir cihazla oturum başlatma

Redirect Server: Yönlendirme sunucuları SIP Kullanıcı Aracılarının ulaşılabilir alternatif konumlarının bulunmasını sağlarlar. Örneğin Mehmet kullanıcısı Ayşe kullanıcısını aramak istesin (Mehmet = SIP:mehmet@genel.com, Ayşe = SIP:ayse@genel2.com). Mehmet Ayşe'yi aramak istediğinde ilk olarak Mehmet kullanıcısının Kullanıcı Aracısı (UA) Ayşe kullanıcısının adresini dener fakat Ayşe kullanıcısının bulunduğu etki alanında gelen istekleri karşılayan bir SIP yönlendirme sunucusu mevcuttur. Bunun yerine Mehmet'in Kullanıcı Aracısı (UA) yönlendirme sunucusuyla irtibat kurar. Ayşe kullanıcısının etki alanında bulunan yönlendirme sunucusu Ayşe kullanıcısının iş yerindeyken SIP:ayse@isyeri.com evdeyken SIP:ayse@ev.com olduğunu bilir. Böylece yönlendirme sunucusu Mehmet'in Kullanıcı Aracısı'na (UA) SIP:ayse@genel2.com yerine bu iki adresi denemesini söyler. Yönlendirme sunucusu önceliklendirme yeteneklerine sahiptirler bu sayede önce işyerini sonra evi denemesini de söyleyebilir. Mehmet'in Kullanıcı Aracısı yeterli bilgiyi edindikten sonra tavsiye edilen iki adresi denemeye başlar. (Bir yönlendirme sunucusu her zaman aranan kullanıcının o anki konumunu dönebilir. Bunun yerine Ayşe'nin konumu hakkında daha fazla bilgi içeren başka bir sunucunun adresini dönebilir.)

Yukarıda ki örnekten de anlaşılacağı üzere bir yönlendirme sunucusu kullanıcının konumunu tespit etmek için bir işlem başlatamıyor, bunun yerine kullanıcının olabileceği yerlerin listesini döner. Kullanıcı aracısı da aranan kişiye ulaşabilmek için ulaşana kadar tüm listeyi dener. Vekil sunucularla yönlendirme sunucular arasında ki ana farkta budur. Vekil sunucular kullanıcı için yeni kontak bilgileri göndermek yerine denemelerde bulunur.

Yönlendirme sunucuları tek kullanıcıya yönlendirme yapabildiği gibi bir gruba da yönlendirme yapabilmektedirler. Örneğin bir firmanın teknik departmanında bulunan Ayşe kullanıcısı 08:00-16:00 saatleri arasında, Ali kullanıcısı 16:00-00:00 saatleri arasında ve Ahmet kullanıcısı da 00:00-08:00 saatleri arasında çalışıyor olsun. Böyle bir durumda yönlendirme sunucusuna gelen çağrı eğer **SIP:teknikdestek@isyeri.com** adresine geliyorsa, gelen isteğin saatine bakılarak arayan kullanıcıya farklı adresler döndürülebilir. Örneğin kullanıcı saat 17:00 'da arıyorsa, arayan kullanıcıya Ali kullanıcısının adresi (**SIP:ali@isyeri.com**) , saat

01:00 'da arıyorsa arayan kullanıcıya Ahmet kullanıcısının adresi (SIP:ahmet@isyeri.com) adresi döndürülür.

Registrar Server: SIP Kullanıcı Aracısı İstemcileri kendilerine atanmış olan benzersiz SIP adreslerini kullanarak buldukları konumlarını kayıt sunucusuna (registrar server) kaydederler. Böylelikle arayan kullanıcı hedef kullanıcıyı aradığında buradan okunan konum bilgilerine bağlı olarak arada ki iletişim gerçekleştirilir. Kayıt esnasında kullanılan bu SIP adresi Tam Nitelikli Alan Adları (FQDNs) veya E.164 numarası veya her ikisi birden olabilir.

Bu SIP adresleri örnek olarak aşağıdaki gibidir:

- sip: "Mehmet Karabayır" <mehmet.karabayir@info-line.com.tr>
- sip: 17705512345@gateway.com; user=phone
- sip:17705512345@gateway.com; user=phone;
sip:mehmet.karabayir@10.0.0.1

İstek yapıldığında kayıt sunucusu SIP kullanıcısının adres bilgilerini konum sunucusuna sağlar. Örneğin bir SIP kullanıcısı başka bir konuma taşınırsa konum sunucusu İnternet tabanlı bir protokol kullanarak (örneğin LDAP) kullanıcının konumunu tespit edebilir.

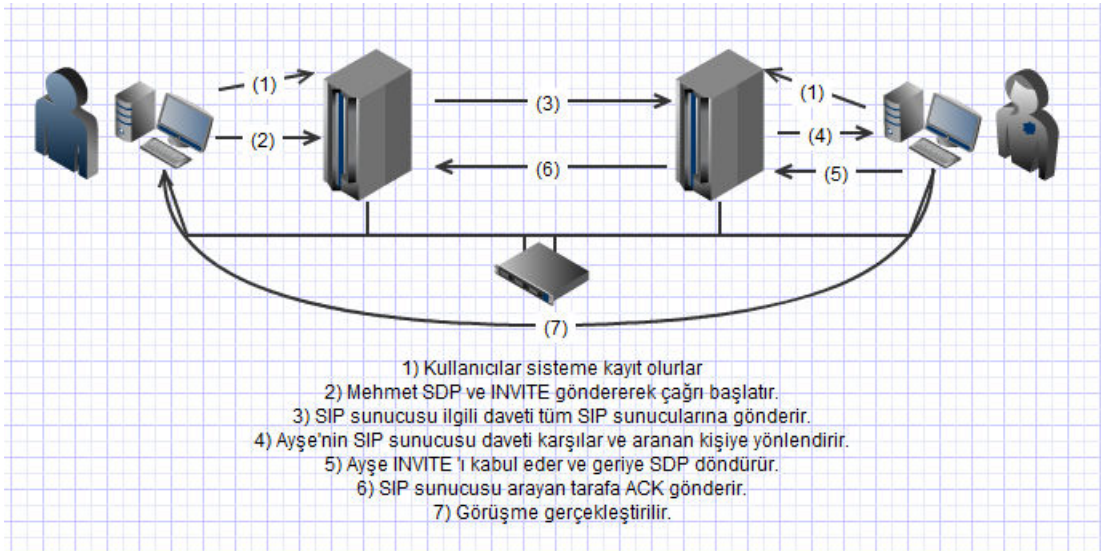
SIP kullanarak Çağrı Kurulumu:

Her biri farklı yerel ağlarda olan ve iki farklı yönlendiriciyle (router) birbirlerine bağlanan iki kullanıcı ele alınsın. Her ağ kendisine has bir SIP sunucusuna sahip. Böyle bir durumda çağrının nasıl gerçekleştiğini aşağıda adım adım görebiliriz: [Şekil 7.4.6]

(Kullanıcı 1: Mehmet , Kullanıcı 2: Ayşe)

- 1- Başlangıç olarak görüşmeyi sağlayacak olan kullanıcıların kendi yerel sunucularına kayıt olmaları gerekmektedir.

- 2- Mehmet, Ayşe'ye doğru bir çağrı başlatır. Öncelikle Mehmet'in Kullanıcı Aracısı (UA) kendi yerel ağında bulunan SIP sunucusuna bir davet (INVITE) gönderir. Bu davette oturumun içeriğiyle ilgili bilgiler bulunur.
- 3- Ayşe kullanıcısının kendi yerel ağında ki SIP sunucusuna kayıt olduğundan Mehmet kullanıcısının sunucusunun haberi yoktur ve nasıl ulaşacağına dair henüz bir bilgiye sahip değildir. Mehmet kullanıcısının sunucusu Ayşe kullanıcısına nasıl ulaşabileceğini bilen tüm SIP sunucularına daveti iletir. (Bu örnekte Ayşe'nin SIP sunucusuna bu davet iletilir)
- 4- Ayşe SIP sunucusuna kayıt olduğundan dolayı Ayşe'nin sunucusu Ayşe'ye ilgili daveti iletir.
- 5- Eğer Ayşe daveti kabul eder ve Mehmet'le konuşmayı kabul ederse, çağrıyı cevaplar ve davetin geldiği yolu takip ederek geriye doğrulama bilgisi (ACK) döner. Ayşe'nin oturum bilgileri de bu geri dönen doğrulama bilgisinin içerisinde yer alır.
- 6- Her iki tarafta karşılıklı olarak oturum bilgilerini paylaştıktan sonra birbirleri ile direk iletişim kurabilmek için gerekli olan IP ve PORT bilgilerine sahip olurlar. Şimdi birbirlerine RTP içeren medyalarını iletebilirler. SIP sunucusunun daha fazla çağrı oturumunda kalmasına ihtiyaç kalmaz.



Şekil 7.4.6 SIP Çağrı Kurulumu

Bir çağrı kuruluşu görüldüğü gibi H.323 protokolüyle çağrı kurulumundan çok daha basittir. TCP protokolüne ihtiyaç duyulmaz böylece daha fazla performans sağlanabilir.

7.5 SIP Protokol Güvenliği

Medya Güvenliği:

SIP protokolünde medya güvenliği SDP tarafından sağlanmaktadır. SDP güvenlik algoritmasını k parametresinde saklamaktadır. RFC2327 'de tanımlanan k tanımlamasının biçimi şu şekildedir:

k=clear:<encription key>

Bu şifreleme algoritması RFC 1890 'da tanımlanmıştır. RFC 1890 ilk olarak standart yoldan bir geçiş deyiminden anahtarın nasıl çıkarılacağını tanımlar. Geçiş deyimini ilk olarak kabul edilmiş formata çevrilir (boş karakterler kaldırılır ve tüm karakterler küçük yapılır) ve daha sonrasında MD5 algoritmasıyla 16sekizliklere dönüştürülerek tekrar sunulur.128 bitten küçük anahtarlar MD5 derleme tarafından kırılarak istenilen biçime sokulur.

Kullanılacak olan algoritma anahtardan önce eklenir ve tek bir bölü (/) işareti ile ayrılır. Sıklıkla kullanılan algoritmalar için standart tanımlayıcılar RFC 1423 'de açıklanmıştır. DES-CBC kullanılarak şifrelenmiş örnek bir anahtar:

k=clear:DES-CBC/mK58hlc1/34sK011a

Eğer medya şifreleme anahtarı korunacaksa, SDP istekleri ve yanıtları şifrelenmek zorundadır. SIP mesajlarını korumanın bunun yanında birçok sebebi olabilir, örneğin çağrıyı başlatan kullanıcının ve hedef kullanıcının bilgileri ve bu görüşme esnasında ki oturum ile ilgili bilgiler saklanmak istenebilir. SIP mesajları IPSEC kullanılarak sekmeden sekmeyle şifrelenebilir. Bunun yanı sıra SIP 'te kaynak kullanıcı ve hedef kullanıcının paylaşmış olduğu bir şifreleme anahtarı sayesinde kaynaktan hedefe kadar da bir şifreleme yapılabilir. Kaynak ve hedef kullanıcılar arasında anlaşılan bu şifre sayesinde kaynaktan gelen şifreli mesajları hedef kullanıcının çözümleyerek anlamlandırması sağlanır. Ortak şifreleme kabul edildiyse

kaynak kullanıcı hedef kullanıcının anlayabileceği şekilde mesajları şifreler. Bu şifreleme işlemi kaynak kullanıcı tarafından ya da aracı bir vekil sunucu tarafından yapılabilir.

İstekler :

İstekler gönderilirken ilk önce istek satırı ve şifreleme metodunun bilgisini içeren şifrenmemiş başlıklar kullanılır. Örnek şifrenmemiş başlık:

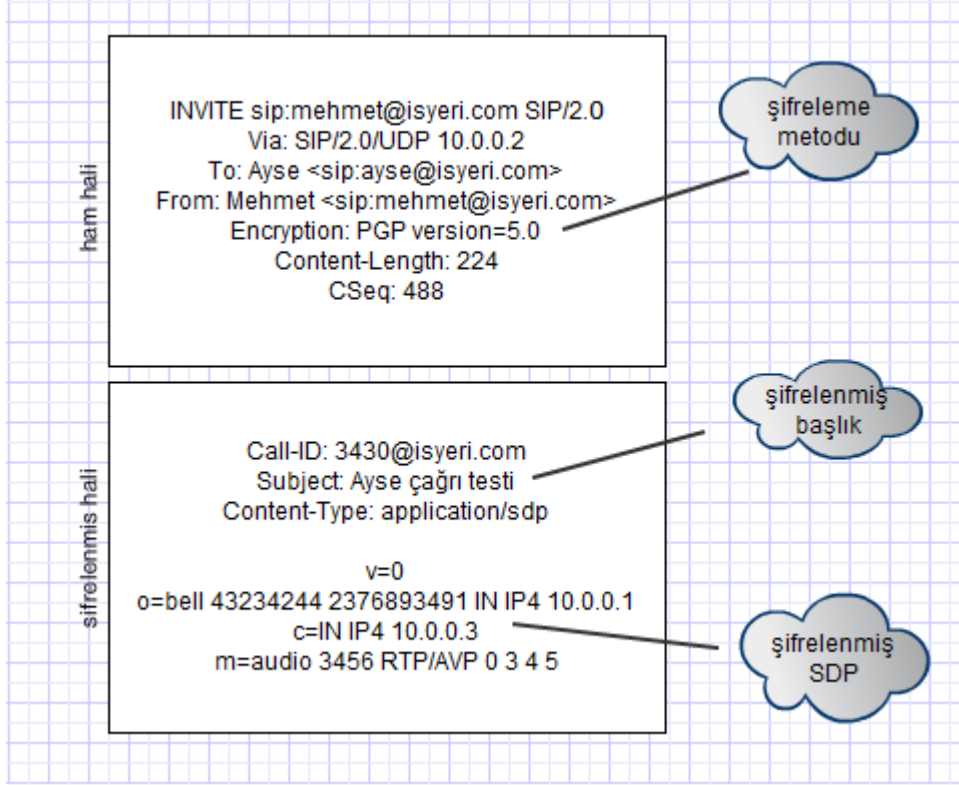
```
PGP version=2.6.2,encoding=ascii
```

Şifrenmiş bölüm ise ilk boş satırdan (bir önceki satırı takip eden ilk boş satır (CRLF)) başlar.

Eğer sadece mesaj gövdesi şifrelenecekse, alıcı kullanıcının şifrenmiş başlıkla mesaj gövdesini ayırt edebilmesi için (karıştırmaması) şifrenmiş satırdan önce bir tane daha ekstra boş satır eklenmek zorundadır.

Yanıtlar:

İsteği gönderen kaynak kullanıcı gelecek olan yanıtı anlamlandırabilmesi için şifreleme anahtarını bilmek zorundadır.

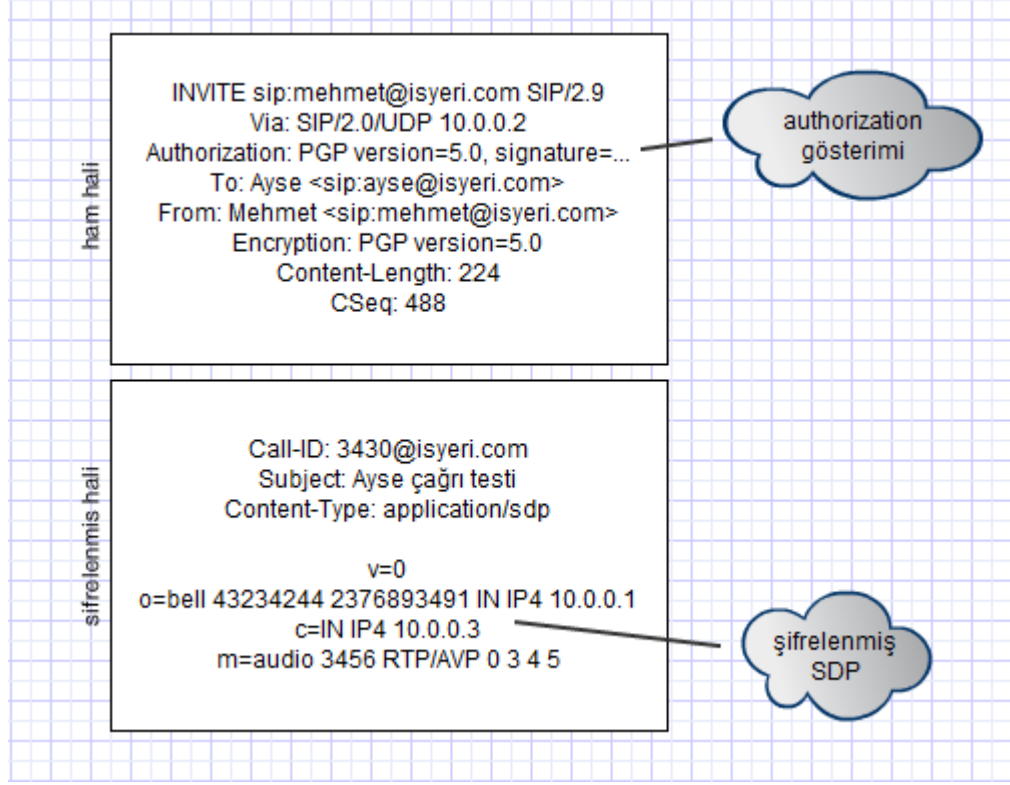


Şekil 7.4.7 SIP Yanıt Mesajı Detayı

Kimlik Denetimi:

SIP istekleri ve yanıtları dijital bir imza kullanılarak şifre doğrulamalı bir yapıya kavuşturulabilir. Authorization başlığı bu amaç için kullanılmaktadır. [Şekil 7.4.8] Bu başlık şu bilgileri içerir:

- İlk satır (istek veya durum satırı)
- Authorization başlığının yanında diğer tüm başlıklar
- Mesaj gövdesi



Şekil 7.4.8 SIP Yanıt Mesajı Detayı (Authorization)

Sesi ve Veri'yi ayırmak için Sanal Yerel Ağ (VLAN)

SIP protokolünü kullanan birçok büyük sistemlerde ses ve veri, sanal yerel ağlar aracılığıyla ayrı tutulmaktadır. Ses ve verinin ayrı tutulmasının performansa yararı olduğu kadar güvenlik açısından da oldukça yararlıdır. Ses blokları farklı sanal yerel ağlara taşınarak veri grubunda bulunan bilgisayarların ataklarından (Ör: DoS saldırıları) korunabilir. MAC filtreleme ile güvenlik konusunda bir adım daha ileriye gidilebilir. Ancak bilgisayarlarda kullanılan yazılımsal telefonlar (softphone) alınmış olan farklı sanal yerel ağ önlemini aşabilir. Böyle bir durumda da yazılımsal telefonlar için güvenli doğrulama yapabilmek amacıyla TLS kullanılmalıdır.

Kullanıcı Doğrulaması

Kullanıcı doğrulaması farklı tipteki isteklere (INVITE, REGISTER, OPTIONS, BYE) konularak güvenlik önlemi artırılabilir. Örneğin kimlik

doğrulaması etkinleştirildiğinde, eğer bir SIP telefonu REGISTER isteğinde bulunursa, SIP vekil sunucusu 401 Kimlik doğrulanamadı mesajı döner.

Sıklıkla Kullanılan Soketlerin Kullanılmaması

Varsayılan olarak kullanılan 5060 ve 5061 dışında ki soketler kullanılarak saldırganın SIP saldırısını 5060'dan yapıp bu soketin açık olmadığı uyarısını almasına neden olacaktır. Böyle bir durumda saldırganın ilgili sunucunun SIP sunucusu olmadığını düşünmesi sağlanabilir. Oldukça basit görünen bu önlem az da olsa güvenliğe katkıda bulunmaktadır.

SIP Güvenlik Duvarları (SIP Firewall)

SIP Güvenlik Duvarları kullanılarak bilinen saldırı tipleri durdurulabilir. Güvenlik duvarı kullanılmaması durumunda saldırılar direk olarak sunucuya yapılır. Bu da sunucunun gereksiz işlemler yapıp asıl işlerine vakit ayıramamasına neden olur. Örneğin bir UDP veya INVITE çoklu denemesinde ki istekler sunucuya gelirse sunucu da çok fazla sayıda işleme neden olabilir. Ancak bu istekler bir güvenlik duvarına gelirse, güvenlik duvarı bunun bir saldırı olduğunu anlayabilir ve saldırıyı yapan kullanıcıyı engelleyebilir.

7.6. SIP ve H.323 Protokollerinin Karşılaştırılması

Tez projesinde SIP protokolü seçildiğinden dolayı SIP hakkında detaylı bilgi önceki konularda verilmiştir. Bu sebepten dolayı da kısaca H.323 ile ilgili bilgilendirme yapmak gerekirse, H.323, ITU (International Telecommunication Union) tarafından geliştirilmiş bir telekomünikasyon standartları protokolüdür ve bir çok versiyonu geliştirilmiştir. [6] H.323, SIP protokolünden daha önce çıkmıştır.

Entegrasyon: H.323 daha çok telekomünikasyon yönelimli yaklaşıma, SIP ise internet yönelimli yaklaşıma sahiptirler. H323'ün entegrasyonu, sinyalleşme komutlarının ikili sistemde olmasından dolayı SIP'e nazaran daha zordur. SIP protokolünün mesaj yapısının HTTP protokolüne benzer olmasından dolayı entegrasyonu kolay bir protokoldür ama H.323 protokolünün daha karmaşık yapıda olması entegrasyonunu da zorlaştırmaktadır. SIP mesajları HTTP protokolünün mesajları gibi metin tabanlıdır, ancak H323 mesajları ikili kodlamada yapılmaktadır. H.323 çok fazla bileşen tanımlayabilir ancak SIP benzer yapıda farklı başlıklar tanımlayarak ilerler. Bu da SIP mesajlarının kolay ayrıştırılmasını ve anlamlandırılarak işlem yapılmasını sağlamaktadır.

Uyumluluk: SIP genişletilebilir ve uyumlu zengin bir set metoda sahiptir. HTTP protokolüyle benzerliklerinden dolayı HTTP genişletilebilirliği için kullanılan metotlar SIP içinde kullanılabilir. SIP bir oturumda hedef kullanıcıyla olacak iletişimde hedef kullanıcının destekleyeceği kodekleri iletmek için SDP (Session Description Protocol) protokolünü kullanır. [7]

H323 'te SIP protokolü gibi genişletilebilir mekanizmalara sahiptir. Bu mekanizmalar ASN.1 'in çeşitli konumlarında yer alan standart olmayan "param" alanlarıdır. Bu Param alanları kendisini takip eden ilgili sağlayıcıdan başkası için anlamsız bir değer içeren, sağlayıcı kodu olan alanlardır. Bu farklı sağlayıcılara kendi uzantılarını geliştirme imkanı sağlar. [6] H.323'te her bir kodek merkezi olarak kayıtlı ve standart olmalıdır. H.323 sistemlerinde kullanılan sub-28.8 kb/s kodeklerde ücretsiz bir kodek olmamasından dolayı küçük geliştiriciler için H.323 protokolü kullanımını bir engel olabilmektedir. [7]

Ölçeklenebilirlik : SIP, H.323'e nazaran daha az işlem yapmaya ihtiyaç duyarlar. Aynı gücü vererek aynı zaman diliminde SIP protokolü ile H.323 protokolüne göre daha fazla çağrı karşılanabilmektedir. [7]

Tablo 7.4.1 SIP vs. H323 Karşılaştırma

Kriterler	H.323	SIP
İletim Protokolleri	TCP ve UDP protokollerini destekler. Güvenilir bir iletim ortamına ihtiyaç duyar.	TCP ve UDP protokollerini destekler. Güvenilmeyen bir iletim ortamında da işlem yapabilir.
Adresleme Biçimi	Adreslerin direk barındırılmasına izin verir. Takma adlar bir çözümleyici (gatekeeper) tarafından çözümlenir.	E-mail, telefon numarası, H.323 ve HTTP içeren Adres bağımsız URL. Telefon numarası benzeri isimler herhangi bir ağ aygıtını işaret edebilir.
Medya İletimi	RTP/RTCP, SRTP.	RTP/RTCP, SRTP.
Güvenlik	H.235 sayesinde güvenlik mekanizması sağlar. İletim katmanı güvenliği için de SSL kullanabilir.	Arayan ve aranan kişinin güvenliklerini HTTP doğrulama mekanizmaları sayesinde sağlayabilir. SSL/TSL sayesinde her aşamada güvenli doğrulama ve şifreleme sağlar.
Mesaj Kodlama	H.323 mesajları sıkıştırılmış ikili kodlama şeklinde taşır. Bu da dar-bant veya geniş-bant genişliğine sahip bağlantılar için elverişlidir.	SIP mesajları ASCII karakter kodlamasına göre taşınır. Mesajlar bir insan tarafından bakıldığında anlaşılabilir.
Video ve Veri Konferansı	H.323 tamamen video ve veri konferansını destekler.	SIP video için limitli destek vermekte, T.120 gibi veri konferansları içinse hiç destek vermemektedir.
Yetenek Takası	Medya için iyi – sinyalleşme için kötü genişletilebilirlik.	Medya için kötü – sinyalleşme için iyi genişletilebilirlik.
Ayrıştırma	Tanımlayıcı alanların ayrıştırılması oldukça karmaşık.	Tanımlayıcı alanların ayrıştırılması çok basit.
Hata Ayıklama	Hata ayıklama oldukça zor	Hata ayıklama gayet basit.
Entegrasyon	Entegrasyon çok zaman alır.	Entegrasyon fazla zaman almaz.
İşlemci Kullanımı	Daha fazla işlem gerektirir.	Daha az işlem gerektirir.

8. SES SINIFLARI

Ses sınıfları; görüşme esnasında görüşmeci tarafların karşılıklı olarak birbirlerine seslerinin iletilebilmesi için kullanılan sınıflardır. Bant genişliğinin önemli bir etken olduğu çağrı merkezlerinde ses sınıfları değişiklikleri sayesinde aynı anda daha fazla kişiye hizmet verebilme gibi olanaklar vardır. Genellikle bant genişliğinden dolayı değişiklik yapılan ses sınıfları aynı zamanda ses kalitesinin ve anlaşılabilirliğinin de en önemli etkenlerindedir. Bir iletişim esnasında hangi ses sınıfının kullanılması gerektiği, bulunan ortamın kaynaklarına ve izin verilen kaynak kullanımına bağlı olarak belirlenebilir. Örneğin şirket içi görüşmelerde genellikle çağrı merkezleri en kaliteli ses sınıflarını tercih ederken, internet üzerinden görüşmelerde ses sınıfını değiştirerek anlaşılabilir seviyede bırakabilir ve daha fazla kişiye hizmet verecek şekilde davranabilirler. Yaygın olarak kullanılan ses sınıfları şu şekildedir;

- 1- **G.711 uLaw** : ITU tarafından geliştirilmiş 64 Kbps boyuta sahip bir ses sınıfıdır. G.711 Voip'lerde en iyi ses kalitesine sahip sınıf olarak kullanılmaktadır. Bu ses sınıfında herhangi bir sıkıştırma yapılmadığından dolayı ve PSTN ağlarda ve ISDN hatlarda kullanılan ses sınıfı olduğundan dolayı ISDN hatlarda olduğu gibi düzenli bir ses akışına sahiptirler. Aynı zamanda herhangi bir sıkıştırma olmadığından dolayı en az gecikmeye sahiptirler bu da daha az işlemci gücü kullanımı demektir. Konuşma esnasında 100Kbps boyutlara ulaşabilmektedirler. Eğer bant genişliğiyle ilgili herhangi bir sıkıntı yoksa Voip operasyonlar için kullanılacak en iyi ses sınıfı çözümdür. (Amerika'da kullanılmaktadır.)
- 2- **G.711 aLaw** : G.711 uLaw ile aynı özelliklere sahiptir. (Avrupa'da kullanılmaktadır.)
- 3- **G.722** : ITU-T tarafından geliştirilmiş ses sınıfı standartlarından birisidir. Yüksek bit hızına sahiptir. 48, 56 ve 64KBit/s bit hızlarına sahiptir.
- 4- **G.723.1** : İki tür iletim hızına sahiptir. Bunlardan birincisi 6.3 kbit/s (24 byte çerçeveler kullanır), ikincisi ise 5.3 kbit/s (20 byte çerçeveler kullanır). Özellikle düşük bant genişliğine sahip Voip uygulamalarında tercih edilir. Ancak bu ses sınıfıyla müzik veya DTMF tonlaması, Faks tonlaması gibi

tonlamalar iletilemez. Bu sebepten dolayı da interaktif sesli yanıt sistemine sahip sistemlerde kullanılamamaktadır.

- 5- **G.726** : İlk olarak uluslar arası hat gruplarında bant genişliğini korumak amacıyla kullanıldı. 32 kbps bit hızıyla neredeyse G.711 'in 64kbps bit hızıyla yakaladığı ses kalitesini yakaladı. Bu ses sınıfı da bir ITU standarttır. 32kbps dışında desteklemiş olduğu 3 adet bit hızı daha vardır. Bunlar 16kbps, 24kbps ve 40kbps'dir. Ancak standart olarak 32kbps bit hızı tercih edilmektedir.
- 6- **G.729** : Dijital sesleri sıkıştırarak 10ms 'lik paketler haline getirir. 8kbit/s bit hızında çalışmaktadır. Görüşme esnasında bu değer 64kbit/s hıza kadar çıkabilmektedir. Düşük bant genişliği ihtiyacı sebebiyle Voip uygulamalarda sıkça tercih edilmektedir. G.729 'un dört türü mevcuttur. Bunlar G.729, G.729a, G.729b, G.729ab türleridir. [9]
- 7- **GSM** : Mobil iletişim için kullanılan ses sınıfı türüdür. Konuşma sinyalleri 20ms. 'lik bloklara bölünmüştür. Bit hızı olarak 13kbps/s hıza sahiptir.
- 8- **iLBC** : Voip uygulamalarda kullanılmak üzere geliştirilmiş ücretsiz bir ses sınıfıdır. Düşük bant genişlikli düşünülerek geliştirilmiş olan bu ses sınıfında iki farklı bit hızı mevcuttur. Bunlardan birincisi 13.33kbit/s bit hızıyla 30ms'lik çerçevelerden oluşmaktadır, ikincisi ise 15.20kbit/s bit hızıyla 20ms'lik çerçevelerden oluşmaktadır.
- 9- **Speex** : Speex 4-48kbps arasında bit hızıyla konfigüre edilebilen açık kaynak kodlu sıkıştırılmış bir ses sınıfıdır. Oldukça esnek bir ses sınıfıdır fakat G729,G726 ve GSM ses sınıflarından daha fazla işlemci gücüne ihtiyaç duymaktadır. Speex'te ses çerçevelerinin boyutları ihtiyaçlara bağlı olarak kullanıcı tarafından ayarlanabilmektedir.

9. UYGULAMA

9.1 Asterisk Genel Bakış, Kurulum ve Konfigürasyon

Asterisk PBX Genel Bakış

Asterisk sıradan bir bilgisayarı ses iletişim sunucusuna dönüştürebilen bir yazılımdır. Genellikle PBX sunucu yazılımı olarak bilinen Asterisk, sadece bir PBX yazılımı değil bunun yanı sıra birçok işlevi yapabilen komple bir ses iletişim sunucu yazılımıdır. 2.000.000 üzerinde aktif kullanıcıya sahip Asterisk, geniş telefon protokolü desteğine sahiptir. Analog hatlar, ISDN/BRI hatlar ve dijital T1/E1 hat grupları gibi bir çok ses iletişim sağlayan ortamların algılanıp işlenebilmesine olanak sağlamaktadır. Bunun yanı sıra SIP,H.323 ve IAX gibi bir çok VOIP protokolüne de destek vermektedir.

Asterisk ile neler yapılabilir?

- Değişik türde bir çok VOIP protokolü için sürücü desteği
- PSTN arayüz kartları ve aygıtları için sürücü desteği
- Gelen çağrılar için karşılama ve yönlendirme olanakları
- Dış arama için çağrı başlatma ve yönlendirme
- Medya yönetim fonksiyonları (kaydetme, oynatma...)
- Hesaplaşma ve ödeme için Çağrı Detay Bilgilerini tutma
- Medya dönüştürmesi (bir medya biçiminden başka bir medya biçimine dönüştürme)
- Protokol dönüştürmesi (bir protokolden başka bir protokole dönüştürme)
- İlişkisel veritabanlarında bulunan bilgilere erişim için veritabanı entegrasyonu.
- Standart internet protokollerini kullanarak web servislerine erişim.
- Dizin sistemlerine erişim için LDAP entegrasyonu.
- Tekil veya Çoğul çağrı köprüleme.
- Çağrı kayıt veya izleme fonksiyonları.
- Çağrıyı işleyebilmek için çağrı akışı planlayabilme.
- AGI (Asterisk Ağ Geçidi Arayüzü) sayesinde herhangi bir programlama dilinde dış çağrı yönetimi.

- AMI (Asterisk Yönetim Arayüzü) sayesinde CTI (Bilgisayar-Telefon Entegrasyonu) entegrasyonu ve olay bildirimi.
- Üçüncü parti araçlar sayesinde değişik dillerde konuşma sentezleyebilme.

Asterisk Kurulumu

Asterisk Sunucu yazılımı kurulurken dikkat edilmesi gereken en önemli nokta seçilen bilgisayarın konfigürasyonudur. Asterisk bir PBX sunucu yazılımı olduğundan dolayı üzerinde çalıştırılacak işlemlerde maksimum performansı alabilmek için seçilen donanımlar bir hayli önem arz etmektedir. Niçin bu kadar önemli olduğu aşağıdaki gibi bir testle görülebilir. Asterisk projesi geliştiricilerinden Joshua Colp 'un yapmış olduğu bu test açıkça sistem konfigürasyonunun bizim için önemini göstermektedir. Test senaryosu şu şekildedir; Basit bir çağrı kurulur. Bir ses dosyası oynatılır ve bir süre beklenir.

Bu testte kullanılan konfigürasyon şu şekildedir:

İşlemci : AMD Athlon 64 X2 4200

Ram : 1 GB

Disk: 80 GB SATA

Tablo 9.1.1 Asterisk Çağrı Performans Testi

Basit bir SIP senaryosu: Senaryo da Wait() ve Playback() komutları kullanılmıştır ve sonuçlar aşağıdaki gibidir			
Eşzamanlı Çağrı Sayısı	330	330	550
CPU Kullanımı	149%	14.8%	57.6%
Yük Ortalaması	49	25	60
Depolama	Sabit disk	Ram disk	Ram disk

Asterisk yazılımının kurulumu için sunucu seçimi aslında oldukça basittir. Çünkü Asterisk herhangi bir x86 platformda çalışabilir. Ancak eğer ilgili sunucu da yüklü bir çağrı akışı, ya da Asterisk fonksiyonlarının kullanımı gerçekleşecekse donanım seçimi zorlaşmaktadır.

- **İşlemci Seçimi:** Sıradan bir bilgisayara kurulan Asterisk yazılımında ses işlemleri DSP'ler tarafından değil de işlemci tarafından yapıldığından dolayı FPU gücü oldukça önem arz etmektedir. FPU gücü ne kadar yüksek olursa eş zamanlı çalıştırılacak DSP görevleri o kadar fazla sayıda olabilecektir.
- Birden fazla sayıda işlemci kullanımı ya da çok çekirdekli işlemci kullanımı da, sistemin aynı anda birden fazla görevi yerine getirmesini sağlayacaktır.
- **Anakart Seçimi:** Anakart sistem yolu minimum gecikmeyi sağlamalıdır. Eğer bir PSTN bağlantı kurulacaksa, sistemde bulunan Zaptel kartlar ortalama 1000 kesme isteği/saniye gerçekleştirecektir. Eğer anakart sistem yolu bu istek hızıyla başa çıkamazsa, görüşmelerde ses kalitesinin düşmesine neden olacaktır.
- **Disk Seçimi:** İşlemcinin verileri RAM diskten okuması sabit diskten okumasından daha hızlı olacağından dolayı yapılacak olan operasyona göre RAM disk seçilmesi daha uygun olabilir. RAM disklerin fiyat olarak daha pahalı olmasından dolayı yapılacak olan projeye göre fiyat/performans değerlendirmeleri ölçümlenmeli ve buna göre karar verilmelidir.

Kurulması gereken Asterisk Kütüphaneleri

Şuanda aktif olarak kullanılmakta olan Asterisk PBX sunucularında yaygın olarak tercih edilen 3 adet paket bulunmaktadır.[4] Bunlar;

- Asterisk : Asterisk'in ana programıdır. Eğer sadece basit bir VOIP ağı düşünülüyorsa, sadece bu paketin kurulması yeterli olacaktır.
- Zaptel : Eğer analog veya dijital donanım kullanımı olacaksa, zaptel sürücülerinin yüklenmesi gerekmektedir.

- Libpri : PRI kütüphanelerini içerir. Eğer kurulacak olan sistemde ISDN PRI arayüzler kullanılacaksa bu kütüphanenin kurulması gerekmektedir.

Bu paketler dışında kullanılan başka Asterisk paketleri de mevcuttur. Kurulacak olan sistemin ihtiyaçlarına bağlı olarak bu paketlerinde sisteme entegre edilmeleri gerekebilir. Örneğin kullanılacak olan sistemde ses sınıflarının da yüklü olması gerekir ki, farklı ses sınıfı kullanan uç noktalarla ses iletişimde herhangi bir sıkıntı yaşanmasın.

Asterisk'in Kurulacağı İşletim Sisteminde Bulunması Gereken Kütüphaneler

Tez uygulamasında kullanılacak işletim sistemi Linux geliştirmesi olan Centos 5.2 işletim sistemi olacağından dolayı gerekli olan kütüphaneleri Centos işletim sistemine göre aşağıdaki şekildedir.

Tablo 9.1.2 Asterisk için Gereken Kütüphaneler

Paket Adı	Yükleme Komutu	Kullanan Sistem
GCC 3.x	yum install gcc	libpri, zaptel, asterisk
Ncurses-devel	yum install ncurses-devel	menuselect
Libtermcap-devel	yum install libtermcap-devel	asterisk
Kernel Development Headers	yum install kernel-devel	zaptel
Kernel Development Headers (SMP)	yum install kernel-smp-devel	zaptel
GCC C++ 3.x	yum install gcc-c++	asterisk
Libtool	yum install libtool	asterisk
GNU make	yum install make	asterisk

Asterisk Tarafından Kullanılacak Olan Klasörler

Asterisk; fonksiyonlarını gerçekleştirmek üzere Centos işletim sistemi üzerinde birçok klasör kullanmaktadır. Tez uygulamasında kullanılacak bu klasörler şu şekildedir:

/etc/asterisk : Asterisk'in konfigürasyon dosyalarının bulunduğu klasördür.

/usr/lib/asterisk/modules : Asterisk'in yüklenebilir modüllerinin barındığı klasördür. (Ses sınıfları, biçimler, kanallar gibi)

/var/lib/asterisk : Asterisk veritabanının tutulduğu klasördür. Veritabanının yanı sıra kendi içinde bulunan alt klasörlerde farklı yapılarda tutulmaktadır.

/var/lib/asterisk/agi-bin : Asterisk AGI senaryolarının tutulduğu klasördür.

/var/lib/asterisk/mohmp3 : Eğer çağrı senaryosunda bekletme sırasında kullanıcıya müzik dinletme özelliği kullanılacaksa, bekletme sırasında kullanılacak olan bu müziklerin bu klasörde barındırılması tercih edilmektedir.

/var/lib/asterisk/sounds : Çağrı senaryosunda kullanılacak olan her türlü anonsun bu klasörde tutulması tercih edilmektedir.

/var/log/asterisk : Asterisk'le ilgili her türlü geçmiş bilgisinin tutulduğu klasördür. (Örnek: /cdr-csv klasörü görüşmelere ait çağrı detay bilgilerini tutmaktadır.)

Yukarıda bahsi geçen klasörler tez uygulamasında kullanılacak klasörlerdir. Bunun yanı sıra Asterisk'in kullanmış olduğu diğer klasörlerde şu şekildedir:

/var/lib/asterisk/firmware : Çeşitli Asterisk uyumlu aygıtlar için sürücü yazılımlarını içeren klasör.

/var/lib/asterisk/images : Grafik destekleyen kanallar üzerinden iletişim yapan uygulamaların bakmış olduğu klasör.

/var/lib/asterisk/keys : Asterisk; uç kullanıcıların RSA dijital imza ile sisteme kimlik doğrulamasından geçerek giriş yapabilmelerini sağlamak amacıyla genel veya özel anahtarlama sistemi kullanabilir. Bu anahtarların barındırıldığı klasördür.

/var/spool/asterisk/outgoing : Asterisk; bu klasörü izleyerek çağrı isteği içeren metin dosyalarını yakalar. Bu klasörde dış arama yapısına uygun oluşturulan metin dosyaları sayesinde dış aramalar gerçekleştirilebilir.

/var/spool/asterisk/dictate : Dikte edilen görüşmelerin bulunduğu klasör. (Kaydedilen dosyalar ham biçimde tutulmaktadır.)

/var/spool/asterisk/system : *System()* uygulaması tarafından geçici dosyalarının tutulduğu alan olarak kullanılmaktadır.

/var/spool/asterisk/tmp : Asterisk uygulamalarının geçici dosyalarını yazması veya okuması için kullandıkları alan.

/var/spool/asterisk/voicemail : Tüm sesli postaların ve kullanıcı karşılımlarının tutulduğu alan.

/var/run : Asteriskte dahil olmak üzere çalışan tüm uygulamaların işlem numaralarının (PID) tutulduğu alan.

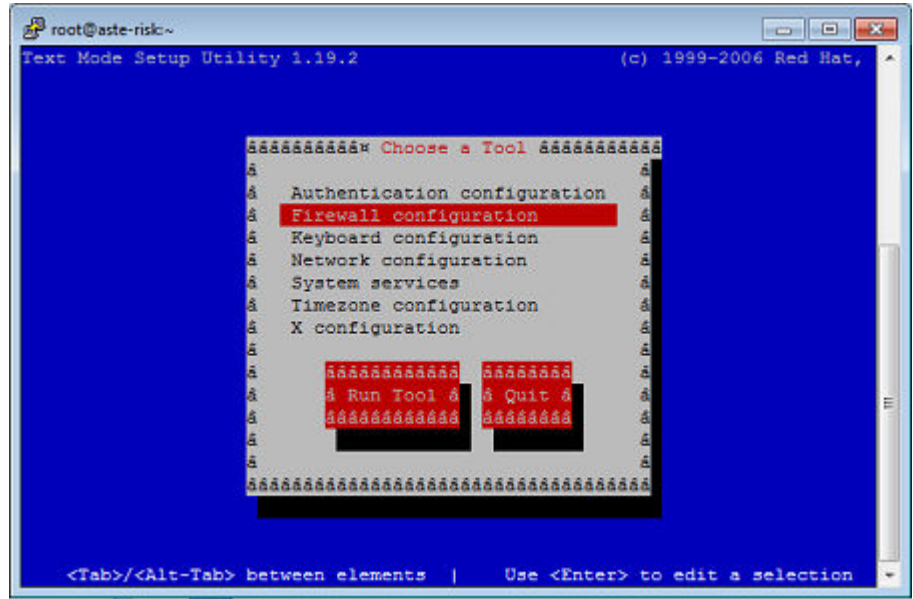
Asterisk Konfigürasyonu

Tez uygulamasında Asterisk VOIP bağlantılar için kullanılacak. Çağrılar internet üzerinden gerçekleştirileceği için Asterisk sunucunun yanı sıra Centos işletim sisteminde de bazı düzenlemeler yapılması gerekecektir. Bu düzenlemeler şu şekildedir;

- 1- İlk olarak bağlanacak kullanıcıların işletim sistemi tarafından engellenmemeleri gerekmektedir. Bunu sağlamanın iki yolu vardır. Bunlardan birisi işletim sisteminin kendi içerisinde bulunan yazılımsal güvenlik duvarının devre dışı bırakılmasıdır. Güvenlik duvarının devre dışı bırakılması belki çağrı gerçekleştirimini sağlayacaktır ancak daha başka sorunlara yol açabilecektir. Örneğin kötü niyetli bilgisayar kullanıcıların sisteme erişimlerine olanak sağlayacaktır. Bu yüzden bu metot pek tercih edilmemektedir. İkinci yöntem ise ilgili güvenlik duvarı üzerinden bağlanacak olan tüm kullanıcılara tek tek izin verilmesi işlemidir. Çağrı merkezleri gibi sıklıkla ara bağlantı yapan firmalar için bu yöntem yaygın bir şekilde kullanılmaktadır. Sırasıyla yöntemler şu şekilde gerçekleştirilmektedir;

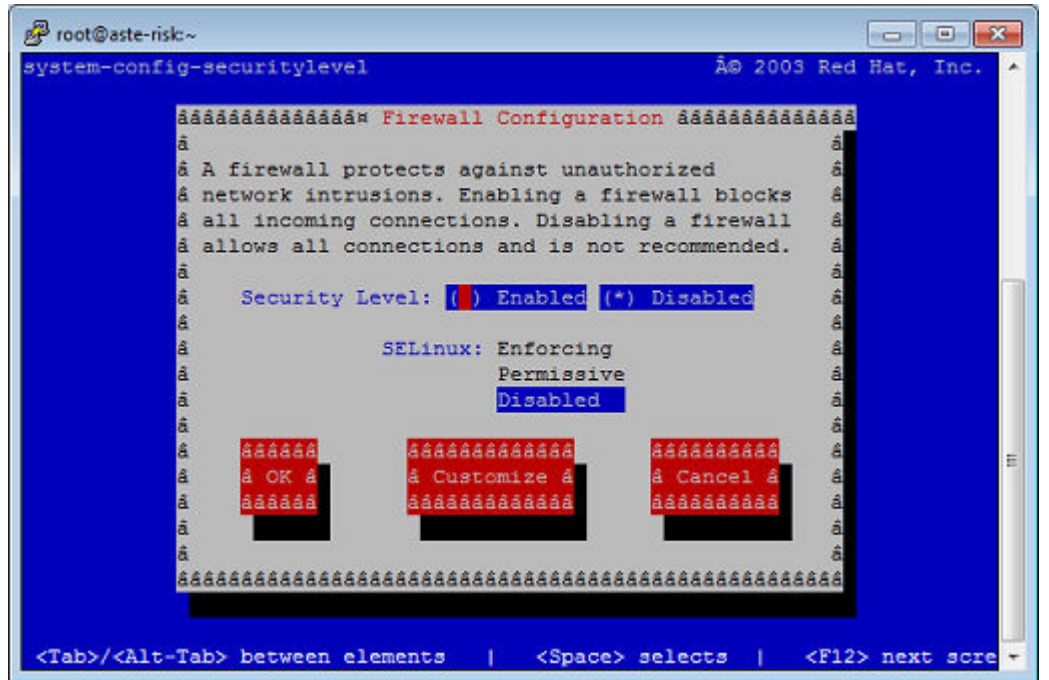
- a. **Yazılımsal Güvenlik Duvarının devre dışı bırakılması:** Bunun için ilgili Centos sunucusunun kurulum ayarlarına girerek Güvenlik Duvarı devre dışı bırakılabilir. Öncelikle *setup* komutu ile kurulum ayarlarına girilir. Burada karşımıza aşağıda ki gibi bir ekran gelir.

Şekil 9.1.1 Güvenlik Duvarı Konfigürasyonu Ana Ekran



Daha sonrasında *Firewall Configuration* alanına girilir ve gelen ekranda bulunan *Security Level* tanımlaması *Disabled* konuma getirilir. Aşağıdaki şekilde görülebilir.

Şekil 9.1.2 Güvenlik Duvarı Konfigürasyonu Detay Ekranı



b. İlgili IP adresine/bloğuna Güvenlik Duvarından izin verilmesi:

İzin verilecek IP adresi 192.168.0.1 olsun. O zaman işlem şu şekilde gerçekleşecektir;

- Eğer sadece tek bir IP adresine izin vereceksek;

```
iptables -A INPUT -s 192.168.0.1 -j ACCEPT
```

- Eğer ilgili IP bloğuna izin vereceksek;

```
iptables -A INPUT -s 192.168.0.0/24 -j ACCEPT
```

veya

```
iptables -A INPUT -s 192.168.0.0/255.255.255.0 -j ACCEPT
```

bu komutlardan birini yaptıktan sonra *iptables* servisinin yeniden başlatılması gerekir. Bu komutlar sayesinde çağrı senaryosunda kullanılacak 192.168.0.1 ip adresi için gerekli izinleri verilmiş olunur.

2- İşletim sisteminde sadece yukarıda belirtilen işlemin yapılması yeterli olacağından dolayı öncelikli olarak bu işlemler yapıldı. Bundan sonra ki işlemler ise tamamen Asterisk tarafında yapılması gereken konfigürasyonlardır. Asterisk 'te VOIP bağlantıların gerçekleştirilebilmesi için 2 adet konfigürasyon dosyasında değişiklik yapılması gerekir. Asterisk üzerinden gerek H.323 protokolüyle gerekse de SIP protokolüyle aynı anda ya da ayrı ayrı çağrı taşınması gerçekleştirilebilir. Tez projesinde SIP protokolü kullanılacağından dolayı SIP protokolü için yapılması gereken konfigürasyonları işlenecek. Asterisk yazılımı konfigürasyon dosyalarını */etc/asterisk* klasörü içerisinde barındırmaktadır. Bu klasör içerisinde bulunan *sip.conf* ve *extensions.conf* dosyaları, konfigüre edilmesi gereken dosyalardır.

- **Sip.conf** : Adından da anlaşılacağı üzere SIP protokolüyle ilgili parametrelerin ayarlandığı konfigürasyon dosyasıdır. İçerisinde bulunan tanımlayıcı alanlar ise şu şekildedir; [10]

context = Gelen tüm çağrılarının varsayılan akış içeriği.

Allowguest = Misafir çağrılara izin verilip verilmeyeceği.

Allowoverlap = Arama aşma desteğinin izin durumu.

Allowtransfer = Çağrı transferlerinin izin durumu.

Realm = Kullanıcı doğrulaması derlemesi için. Varsayılan olarak “asterisk” olarak ayarlanmıştır. Eğer bir realm değeri atanacaksa; RFC 3261 ‘e göre Realm değeri global olarak tekil olmalıdır.

Bindport = Yayının yapılacağı UDP portu. (Varsayılan SIP Portu: 5060)

bindaddr = Yayının yapılacağı sunucu makinesinin IP adresi.

Srvlookup = Dış aramalarda DNS ‘ten ilgili kullanıcıların bakılması.

Pedantic = Başlıklarda bulunan etiketlerin kontrolü.

Tos_sip = SIP paketleri için servis tipini seçimi.

Tos_audio = Ses paketleri için servis tipini seçimi.

Tos_video = Video paketleri için servis tipini seçimi.

Maxexpiry = Gelen katılımların ve kayıtların maksimum oturum zaman aşımı izin süreleri.

Minexpiry = Gelen katılımların ve kayıtların minimum oturum zaman aşımı izin süreleri.

Defaultexpiry = Gelen/Giden katılımların ve kayıtların oturum zaman aşımı izin süreleri.

T1min = İzlenen sunuculara mesajların minimum gidiş/dönüş süreleri.

Notifymimetype = Mime tipinin belirlenmesi.

Checkmwi = Kullanıcılar için posta kutusu kontrol süresi.

Vmexten = Posta kutularına erişimin sağlanabilmesi için bakılacak çağrı akış içeriği.

Disallow = Engellenecek ses sınıflarının belirlendiği tanımlayıcı.

Allow = İzin verilecek ses sınıflarının belirlendiği tanımlayıcı.

Mohinterpret= Eğer çağrı akış planında herhangi bir bekletme müziği belirtilmeden kullanıcı

bekletilmeye alınırsa, burada belirtilen içeriğin müziği dinletilir.

Mohsuggest= Arayan kullanıcı eğer bekletilmeye alınırsa hangi müziğin dinletileceğinin belirlendiği alan.

Language = Dil seçimi alanı.

Relaxdtmf = Eğer dtmf algılamasıyla ilgili bir sıkıntı yaşıyorsa, relaxdtmf ayarını aktif hale getirerek algılamının daha hassaslaşması sağlanabilir.

Useragent = User agent deęeri.

Promiscredir = Eęer bu deęer *yes* olarak ayarlanırsa, yerel olmayan SIP adresleri iin ynlendirmeye izin verilir. Eęer lokal bir SIP adresi iin gerekleřtirilirse dngye sebep olur.

Dtmfmode = Gnderilecek olan DTMF modunun belirlendięi alan.

Videosupport = SIP iin video desteęi.

Maxcallbitrate = Video akıřları iin maksimum bit hızı.

Alwaysauthreject = Herhangi bir INVITE ya da REGISTER isteęi reddedildięinde bunların sebebi olarak “401 Unauthorized” gsterilmesinin saęlanması.

Contactdeny = Belirtilen IP aralıęında ki kullanıcılardan gelen istekleri reddeder.

Contactpermit = Belirtilen IP aralıęında ki kullanıcılardan gelen istekleri kabul eder.

Rtptimeout = Eęer ses kanalında belirtilen sre ierisinde herhangi bir RTP ya da RTCP aktivitesi gerekleřmezse aęrıyı sonlandırır.

Rtpholdtimeout = Eęer bekleme konumundayken ses kanalında belirtilen sre ierisinde herhangi bir RTP ya da RTCP aktivitesi gerekleřmezse aęrıyı sonlandırır. Bu deęer her zaman iin rtptimeout deęerinden byk olmak zorundadır.

Rtpkeepalive = NAT’ı aık tutmak iin RTP akıřında canlı tutmalar (keepalives) gnderimi.

Sipdebug = SIP iřlemleri iin hata ayıklamanın devreye alınması.

Recordhistory = SIP gemiřinin kaydedilmesi.

Dumphistory = SIP diyalogunun sonunda gemiřin bořaltılması.

Registertimeout = Kayıt denemesinin ne srede tekrar gerekleřeceęi bilgisi.

Registerattempts = Kayıt denemesinin ka kere tekrar denenebilirlięinin bilgisi.

Externip =Eęer bir NAT arkasındaysak SIP mesajlarının gnderileceęi adresin ip bilgisi.

Externhost = IP bilgisine alternatif olarak eęer bir NAT arkasındaysak SIP mesajlarının gnderileceęi adresin sunucu bilgisi.

Externrefresh = exterhost’un ne sıklıkla yenileneceęi parametresi.

Localnet = Birden fazla yerel ađın eklenmesine olanak tanır.

Nat = Sunucunun NAT arkasında olup olmadığı bilgisinin ayarlandığı parametre.

Canreinvite = Varsayılan olarak Asterisk RTP medya akışını arayan kişiden aranan kişiye doğru direk olarak gerçekleştirmektedir. Bazı aygıtlar buna destek vermemektedir. (Özellikle NAT arkasında ki cihazlar). Böyle durumlarda bu parametre ilgili duruma göre ayarlanarak medya akışının gerçekleşmesi sağlanabilir.

Örnek bir sip.conf gösterimi;

```
[general]
context=default
allowoverlap=no
bindport=5060
bindaddr=192.168.0.1
srvlookup=yes
disallow=all
allow=g729
dtmfmode=rfc2833
rtptimeout=60
rtpholdtimeout=300
canreinvite=no
```

- **Extensions.conf** : Bu dosya sip.conf dosyasında tanımlanmış olan SIP kullanıcıların hangi içeriğe göre çağrı akışlarını işleyeceklerini belirten, bir veya daha çok çağrı akış içeriğinden oluşan işleyiş dosyasıdır. Burada hangi kullanıcının nasıl karşılanacağı, gelen/giden çağrılarının nereye nasıl yönlendirileceği, nereleri arayabileceği gibi çağrı akışını belirten işleyişler belirtilir. Varsayılan çağrı akış içerik işleyişi olarak *[general]* içeriği kullanılmaktadır. Eğer bir SIP kullanıcısı için herhangi özel bir içerik belirtilmemişse, Asterisk otomatik olarak *[general]* içeriği içerisindeki adımları gerçekleştirir. *[general]* içeriği her zaman için extensions.conf dosyasının en başında bulunmalıdır. Bu içerik içerisinde çağrı akış planına bağlı olarak şu seçenekler kullanılabilir;

- **static:** “*save dialplan*” komutunun çalıştırılıp çalıştırılmayacağını belirler.
- **writeprotect:** Eğer static=yes ve writeprotect=no olarak ayarlanırsa, CLI komut satırında “*save dialplan*” diyerek çağrı akış işleyişinin kaydedilmesi sağlanabilir.
- **autofallthrough:** Eğer bir dahilinin yapacağı işler bitmişse çağrıyı Asterisk’in o anki tahminine bağlı olarak BUSY, CONGESTION veya HANGUP ile sonlandırır.
- **clearglobalvars:** Eğer clearglobalvars “yes” olarak ayarlanırsa global değişkenler temizlenir.
- **priorityjumping:** Eğer priorityjumping “yes” olarak ayarlanırsa, uygulama şayet priorityjumping’i destekliyorsa belirtilen öncelikte ki noktaya giderek işleme devam edebilir.

Örnek bir extensions.conf gösterimi;

```
[general]
static=yes
writeprotect=yes
autofallthrough=yes
clearglobalvars=no
priorityjumping=no
```

Madde-2 'de belirtilen bu iki konfigürasyon dosyasında ki gerekli düzenlemeler de yapıldıktan sonra çağrı alabilmesi/gönderebilmesi için SIP kullanıcıların tanımlanması gerekir. İlgili kullanıcılar tanımlandıktan sonra da çağrı akış planlarının belirlenmesi gerekir. Basit bir çağrı örneğiyle bu şu şekilde işlenebilir;

Çağrı Başlatma / Karşılama Örneği;

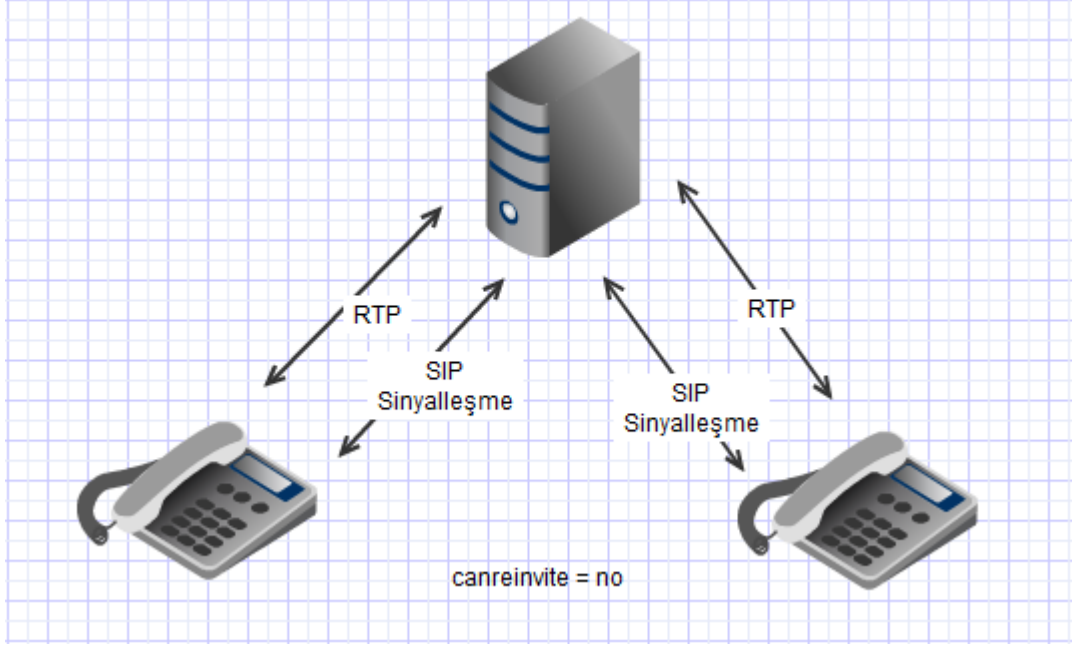
Senaryo şu şekilde olsun. 2 adet kullanıcı olsun. Bu kullanıcılardan 1.kullanıcı, 2.kullanıcıya doğru bir çağrı başlatsın. 2.kullanıcımızın çağrı akış planlamasında öncelikle arayan kullanıcıya bir karşılama anonsu dinletilsin daha sonra da çağrı 2.kullanıcıya gönderilsin ve görüşme gerçekleşsin. Bu senaryo da kullanılacak olan kullanıcıların bilgileri şu şekilde olsun;

Tablo 9.1.3 SIP Kullanıcı Hesap Bilgileri

Tanımlama/Kullanıcılar	1.Kullanıcı	2.Kullanıcı
Ip Adresi:	192.168.0.2	192.168.0.3
SIP Portu:	5060	5060
Kullanıcı Adı:	3401	3402
Şifre:	34011	34022
Görünen Ad:	3401	3402
Tip:	Friend	Friend
Nat:	Yes	Yes
Can reinvite?	No	No
Ses Sınıfları:	G729	G729
Çağrı Akış İçeriği:	out_caller	in_callee

Aşamalar şu şekilde gerçekleşecektir;

- 1- Kullanıcı tanımlamaları yapılır.
- 2- Çağrı Akış planları oluşturulur.



Şekil 9.1.3 SIP Çağrı Başlatma (canreinvite=no)

1- Kullanıcı tanımlamalarının yapılması;

a. 1.Kullanıcı için;

sip.conf dosyasına şu satırları ekleyerek 1.kullanıcı oluşturulabilir.

```
[3401] ; 1.Kullanıcı  
type=friend  
regexten=3401  
host=192.168.0.2  
callerid=3401  
nat=yes  
canreinvite=no  
disallow=all  
allow=g729  
context=out_caller  
secret=34011
```

b. 2.Kullanıcı için;

sip.conf dosyasına şu satırları ekleyerek 2.kullanıcı oluşturulabilir.

```
[3402] ; 2.Kullanıcı
type=friend
regexten=3402
host=192.168.0.3
callerid=3402
nat=yes
canreinvite=no
disallow=all
allow=g729
context=in_callee
secret=34022
```

2- Tanımlanmış olan kullanıcılar için Çağrı Akış Planlarının oluşturulması.

a. 1.Kullanıcı için;

extensions.conf dosyasına şu satırları ekleyerek 1.kullanıcı için çağrı akış planlaması oluşturulabilir.

```
[out_caller]
exten => _3402,1,Goto(in_callee,s,1)
exten => _3402,2,Hangup
```

b. 2.Kullanıcı için;

extensions.conf dosyasına şu satırları ekleyerek 1.kullanıcı için çağrı akış planlaması oluşturulabilir.

```
[in_callee]
exten => s,1,Playback(karsilama-anonsu)
exten => s,2,Dial(SIP/${EXTEN}@192.168.0.1)
exten => s,3,Hangup
```

Bu tanımlamalar tamamlandıktan sonra Asterisk sunucusunun yeni tanımlamaları görmesi için ilgili konfigürasyonların yüklenmesi gerekir. Bunun için öncelikle *asterisk -r* komutu ile Asterisk sunucusunun CLI komut satırı ekranına gelinir ve daha sonrasında *module reload* komutu sayesinde yapılmış olan değişiklikler Asterisk sunucusu tarafından görülebilir hale getirilir. Daha sonrasında SIP telefon cihazları aracılığıyla çağrı gerçekleştirilebilir.

En basit bir çağrı senaryosu gerçekleştirebilmek için gereken konfigürasyonlar/tanımlamalar bu şekildedir. Yapılmak istenen senaryoya bağlı olarak bu konfigürasyonlar değiştirilebilir. Birebir görüşme de bu şekilde bir senaryo yeterli idi. Ancak tez projesinde bir çağrı merkezi işleyişi yapılacağından dolayı bir çok farklı konfigürasyon yapılması gerekmektedir. Bu konfigürasyonlara geçmeden önce Çağrı Akış Planlaması nasıl yapılır, Çağrı Grupları nasıl oluşturulur ve Çağrı dağıtımını ne şekilde yapılır konularının işlenmesi gerekmektedir.

9.2 Çağrı Akış Planlaması ve Çağrı Grupları

Asterisk Çağrı Akış Planlaması (Dial-plan)

Oluşturulan her bir kullanıcı tanımlamasının yapacağı/alacağı çağrılarının nasıl yönlendirileceğinin belirlenmesi için bir çağrı akış içeriğine ihtiyacı vardır. Bu çağrı akış planlamasında çağrının nasıl yönlendirileceğinin belirlenmesi sağlanmaktadır. Bir çağrı geldiğinde önce anonsla mı karşılaştırılacak, çağrı direk başka bir yere mi transfer edilecek, bir kullanıcı mı aranacak, interaktif sesli yanıt sistemine mi gönderilecek, bir kullanıcıya mı yönlendirilecek ya da bir kullanıcı grubuna mı yönlendirilecek gibi her türlü ilerleme, çağrı akış planlamalarında gerçekleştirilmektedir. Çağrı akış planlamasının kendine özgür bir komut dizisi bulunmaktadır. Bu komut dizisi sayesinde çağrının akışı belirlenmektedir. Çağrı akış planlaması gerek dışarıdan gelen çağrılarının içeriye alınmasında, gerekse de içeriden dışarıya doğru yapılacak çağrılarının gerçekleştirilmesinde gerekli yönlendirmeleri yapabilmektedir.

Çağrı akış planlamasında önceliklendirme olanakları da mevcuttur. Böylece çağrı işleyişinde hangi komutların hangi sırayla çalıştırılacağı belirlenebilmektedir. Bu özellikle interaktif sesli yanıt sisteminde oldukça sık kullanılan bir özelliktir.

Asterisk'in bir çağrı akışı esnasında ne tür işlemlerin çalışması gerektiğiyle ilgili standart bir komut dizisi mevcuttur. Kullanıcılara ait kişisel dahililerin dışında Asterisk'in amaca hizmet eden kullanmış olduğu bazı dahili tanımlamaları da mevcuttur. Amaca özel kullanılan bu dahililer şu şekildedir;

- **i** : Çağrı akış planlamasında eğer kullanıcıdan gelen tuşlamaya ait herhangi bir tanımlama yoksa, çağrının yakalanacağı dahili tanımlama i tanımlamasıdır. Diğer bir deyişle "hatalı tuşlama" yakalayan dahili tanımlama.

[invalid-example]

exten => 1,1,Dial(SIP/3401@192.168.0.1)

exten => 2,1,Dial(SIP/3402@192.168.0.1)

exten => i,1,Answer

exten => i,2,Playback(hatali-tuslama)

exten => i,3,Hangup

- **s** : Eğer herhangi spesifik bir dahili numarası belirtilmemişse, içerik varsayılan olarak s tanımlamasından başlar. Başlangıç dahili tanımlaması olarak ta adlandırılabilir.

[start-example]

exten => s,1,Dial(SIP/3401@192.168.0.1)

exten => s,2,Hangup

- **h** : Bir çağrıyı sonlandırmak için kullanılır. Çağrı kapatılmadan önce bir uğurlama mesajı konulmasına da olanak sağlar. Özellikle telefon kartlarıyla görüşme yapan kullanıcıların kontörleri bittiğinde kontörlerinin bittiğine dair bilgilendirme mesajı verilmesinde kullanılmaktadır. Aşağıda ki örnekte yeni

bir çağrı geldiğinde global değişken olan sayac değişkeni +1 arttırılır, bir çağrı kapatıldığında da sayac değişkeni -1 azaltılır. Böylelikle aktif çağrılarının adedi tutulur.

```
[global]
sayac=0
```

```
[hangup-example]
exten => s,1,DialSet(sayac=${${sayac} + 1}|g)
exten => s,2,Dial(SIP/${EXTEN}@192.168.0.1)
exten => h,1,Set(sayac=${${sayac} - 1} |g)
```

- **t** : Belirtilen farklı türde ki zaman aşımı süreleri içerisinde sistemin kullanıcıdan beklemiş olduğu olaylar yerine getirilmediğinde devreye girer. Zaman aşımı durumlarını yakalar.

```
[timeout-example]
exten=> s,1,Background(islem-turu-seciniz)
exten => s,2,WaitExten(5)
exten => 1,3,Playback(islem-turu-1)
exten => 2,4,Playback(islem-turu-2)
exten => t,1,Playback(herhangi-tuslama-yok)
exten => t,2,Playback(tekrar-deneyiniz)
exten => t,3,Goto(timeout-example,s,1)
```

- **T** : Bir görüşmenin maksimum ne kadar süre olacağını belirtildiği tanımlama.

```
[Timeout-Absolute-Example]
exten => 3401,1,Answer()
exten => 3401,2,Set(TIMEOUT(absolute)=10)
exten => 3401,3,Playback(hosgeldiniz)
exten => 3401,4,Wait(1)
exten => 3401,5,Goto(3)
```

exten => T,1,Wait(1)

exten => T,2,Playback(tesekkurler)

exten => T,3,Hangup()

- **a** : Sesli posta sırasında kullanıcı eğer * tuşuna basarsa, çağrılan tanımlama.
- **o** : Operatör tanımlaması.
- **failed** : Otomatik arama işlemi gerçekleşmemesi durumunda devreye giren tanımlama.

exten => failed,1,Set(NumberDialed=\${CUT(PassedInfo,,1)})

exten => failed,n,SetCDRUserField(\${NumberDialed})

- **fax** : Zap kanallarında faks algılaması amacıyla kullanılmakta olan tanımlama.
- **talk** : BackgroundDetect komutuyla birlikte kullanılan tanımlama. BackgroundDetect sayesinde bir arama gerçekleştiğinde karşı tarafın bir insan mı yoksa bir robot mu olduğunun anlaşılması sağlanmaktadır. Eğer insan sesiye ilgili çağrı akışı *talk* tanımlamasından devam etmektedir.

Asterisk Çağrı Grupları Nasıl Oluşturulur?

Çağrı grupları; bir şirketin departmanlarının ayrıştırılmasında, çağrı merkezlerinin projelerini ayrıştırmasında ya da bankaların operasyonlarını ayrıştırmasında sıklıkla kullanılan bir özelliktir. Bu özellik sayesinde gelen çağrıların doğru gruplara yönlendirilmesi sağlanarak daha çabuk çözümlenmeye gidilebilir. Örneğin bir çağrı merkezi düşünölsün; bu çağrı merkezinde bir kredi kartları bölümü bir de vadeli hesaplar bölümüne bakan iki grup olsun. Eğer bu gruplar birbirlerinden ayrıştırılmazlarsa gelen çağrı her iki gruptan birinin kullanıcıasına düşebilir. Böyle

bir durumda da kredi kartlarıyla ilgilenmeyen vadeli hesaplar kullanıcısına kredi kartlarıyla ilgili soru sorulmasına neden olur ve cevabın verilememesi ya da yanlış verilmesi durumu söz konusu olabilir. Çağrı gruplarının oluşturulmasıyla beraber ilgili çağrının konuya özgün uzman kullanıcılar tarafından karşılanması sağlanabilmektedir.

Çağrı gruplarının çağrı dağıtım yöntemlerine bağlı olarak ta gruba ait üyelerin maksimum verimde efektif bir şekilde kullanılabilmesi sağlanabilmektedir. Çağrı grupları oluşturmada ve yönetmede çağrı dağıtım yöntemleri ve önceliklendirme işlemleri oldukça önemlidir. Etkili çağrı dağıtım yöntemleri sayesinde daha az masrafla daha fazla çağrı karşılanabilir, önceliklendirme sayesinde de çağrının öncelikli olarak belirlenen kişi ya da gruplara düşmesi sağlanabilir.

Çağrı gruplarının sıklıkla kullanıldığı diğer bir alan da proje bazlı çalışmalardır. Türkiye’de bulunan bazı büyük çağrı merkezi kuruluşları dışarıdan almış oldukları farklı projeler için çağrı merkezi hizmeti vermektedirler. Böyle bir durumda hizmet vermiş oldukları projeye özel müşteri temsilcilerine çağrılarının düşürülmesi gerekmektedir. Örneğin bir çağrı merkezi düşünölsün. Bu çağrı merkezi aynı anda hem A Bankasına hem de B bankasına hizmet versin. Eğer gruplandırma olmasaydı çağrılar hem A bankasının müşteri temsilcilerine hem de B bankasının müşteri temsilcilerine düşebilirdi. Çağrı grupları oluşturarak çağrının bizim belirlemiş olduğumuz yönergelerle ilerlemesi sağlanabilmektedir.

Reklamın her alanda yayılmasıyla beraber çağrı grupları içerisinde de reklam uygulamalarının var olması durumu meydana gelmiştir. Çağrı gruplarına özel konulan bekletme müzikleri sayesinde kurumlar müşterinin müşteri temsilcisine düşene kadar ki zamanda da kurumla ilgili kampanya duyurularını yapabilmektedirler. Özellikle tatil hizmeti veren kuruluşlar bunu sıkça kullanmaktadırlar. Örneğin A tatil firması aranılır. Tüm müşteri temsilcileri dolu olduğundan dolayı müşteri kuyrukta sıraya alınır ve bu arada da bekletme müziği dinletilir. Bu bekleme süresi içerisinde A tatil firması koymuş olduğu reklam

sayesinde müşterisinin kafasında tatil ile ilgili bazı işaretler oluşturabilir. Bu da ilgili tatil firması için müşteri çekme konusunda önemli bir reklam mecrası haline gelir.

Çağrı grupları oluştururken iki dosya da yapılması gereken bazı konfigürasyonlar söz konusudur. Bu dosyalar `agents.conf` ve `queues.conf` dosyalarıdır. `Agents.conf` dosyası; hangi kullanıcının hangi grupta bulunduğu bilgisinin yazıldığı/okunduğu dosyadır. `Queues.conf` dosyası da gruba çağrı geldiğinde çağrının kuyrukta nasıl bir algoritmayla bekleyeceğinin belirlendiği dosyadır.

`Agents.conf` tanımlamaları:

- **autologoff:** Müşteri temsilcisinin oturumu kapatılmadan önce arama yapıldığında cevapsız bir şekilde maksimum ne kadar süre çalacağını belirtildiği değer.
- **ackall:** Eğer bu seçenek “yes” olarak ayarlanırsa, Müşteri temsilcisinin *AgentCallBackLogin* uygulaması ile oturum açması sağlanır.
- **wrapuptime:** Bir çağrı kapatıldıktan sonra ne kadar süre boyunca ikinci bir çağrının alınmayacağını belirlendiği değer.
- **musiconhold:** Müşteri temsilcisine özel bekletme müziğinin koyulabilmesini sağlar. Varsayılan olarak *default* müziği bekletme müziği olarak kullanılmaktadır. Eğer bu değiştirilmek istenirse, *musiconhold.conf* dosyasında bekletme müziği oluşturulur ve buradan sadece ilgili bekletme müziğinin ismi yazılarak çağırılabilir.
- **updatecdr:** Bu seçenek sayesinde görüşme sonrasında oluşacak çağrı detay bilgilerinin kaydedilip kaydedilmeyeceği belirlenebilir.
- **group:** Müşteri temsilcilerinin kolay yönetilebilmelerini sağlayabilmek amacıyla müşteri temsilcilerini ayrı ayrı gruplara bölüştürebilmeyi sağlar.
- **recordagentcalls:** Müşteri temsilcisinin konuşmalarının kaydedilmesini sağlar.
- **recordformat:** Kaydedilecek olan görüşmelerin hangi biçimde kaydedileceğini belirler. Varsayılan olarak `.gsm`, `.wav` ve `.wav49` biçimlerine destek vermektedir.

- **createlink:** Çağrı detay bilgisi kullanıcı alanına oluşturulan ses kaydının isminin eklenmesini sağlar.
- **urlprefix:** Kaydedilen ses kaydına metin eklenmesine izin verir. Bu metin bir URL 'de olabilir.
- **savecallsin:** Kaydedilecek olan ses kayıtlarının hangi klasöre kaydedileceğinin belirtildiği alan. Eğer herhangi bir değer verilmezse varsayılan olarak */var/spool/asterisk/monitor* klasörüne kaydedilir.
- **custom_beep:** Kişiye özel bip sesi ayarlanabilmesine olanak tanır.

Bu tanımlamalar işlendikten sonra bir müşteri temsilcisinin nasıl oluşturulacağı aşağıda görülmektedir;

Müşteri temsilcileri **agent => temsilci_numarasi,temsilci_sifresi,gorunen_ad** biçiminde oluşturulur.

- **temsilci_numarasi:** Müşteri temsilcisinin sisteme giriş numarası.
- **temsilci_sifresi:** Müşteri temsilcisinin sisteme giriş şifresi.
- **gorunen_ad:** Bir arama gerçekleştirdiğinde karşı tarafta müşteri temsilcisinin adının ne olarak gözükeceğinin belirlendiği alan.

Grupları da müşteri temsilcilerini de içeren örnek bir agents.conf dosyası şu şekilde olmalıdır;

```
[general]
persistentagents=yes

[agents]
autologoff=15
ackcall=no
wrapuptime=5000
musiconhold => default
recordagentcalls=yes
recordformat=gsm
```

```
group=1
agent => 101,101,temsilci1
agent => 102,102,temsilci2
agent => 103,103,temsilci3
agent => 104,104,temsilci4

group=2
agent => 1001,1001,operatör
agent => 1002,1002,mehmet

group=3
agent => 9000,9000,operatör2
```

Queues.conf tanımlamaları:

Bu dosyada iki tip içerik bulunmaktadır. Bunlardan birincisi *[general]* içeriğidir. *[general]* içeriğinin ismi değiştirilemez. Eğer herhangi başka bir içerik belirtilmediyse kuyruğa giren her akış *[general]* içeriğinin akışına göre ilerler. İkinci içerik türü ise kişisel isimlendirmelerden oluşan kuyruk tipleridir. İçerik tanımlama da dikkat edilmesi gereken en önemli husus “[“ ve “]” işaretlerinin unutulmamasıdır. Aksi takdirde içerik olarak algılanamazlar. Bir içeriğin akışı diğer bir içeriğin başlamasıyla sınırlıdır.

***[general]* içeriğinin tanımlamaları;**

- **persistentmembers:** Eğer bu seçenek aktif edilirse, Asterisk her bir ayrı kuyruktan dinamik olarak oturum açmış müşteri temsilcilerini veritabanında tutar. Böylece eğer Asterisk yeniden başlatılırsa müşteri temsilcileri otomatik olarak kaydedildikleri kuyruklarından okunurlar.
- **autofill:** Kuyrukların eski işleyiş türünde seri tipte dağıtım yapılırdı. Yani kuyrukta bekleyen müşteriler sistemde birden fazla müşteri temsilcisi olmadığı sürece bekletilirdi ve en başta bulunan müşteri bağlanana kadar diğer bekleyenler yine beklerlerdi. Yeni davranış tipinde ise bekleyen müşteriler müşteri temsilcilerine paralel bir şekilde bağlanmaktadır ta ki

bekleyen müşteri veya boşta müşteri temsilcisi kalmayana kadar. Bu özellik sayesinde kuyrukta kaybedilen gereksiz zaman kaybının önüne geçilmesi sağlanmıştır.

- **monitor-type:** Kuyruk esnasında gerçekleşen konuşmaların kaydedilmesini sağlar.

Kişisel isimlendirmeden oluşan içeriklerin tanımlamaları;

- **musicclass:** Bu içerik için hangi müziğin kuyrukta bekleme müziği olarak belirlendiği alan.
- **announce:** Müşteri temsilcileri çağrıyı karşıladıklarında duymalarının istendiği anonsun belirtildiği alan. Genellikle bu özellik birden fazla projede bulunan müşteri temsilcilerinin projeleri birbirine karıştırmamaları için hangi projeye ait çağrı geldiyse o projeye ait kısa anonsların yapılması, bilgilendirme amacıyla kullanılır.
- **strategy:** Gelen çağrıların çağrı dağıtımlarının nasıl yapılacağı belirlendiği parametre. Burada birden fazla çağrı dağıtım yöntemi bulunmaktadır. Bunlar:
 - o **ringall:** Gruptaki tüm müşteri temsilcileri çaldırılır, ta ki bir müşteri temsilcisi telefonu açana kadar. Varsayılan olarak kullanılan çağrı dağıtım yöntemidir.
 - o **roundrobin:** Her bir uygun müşteri temsilcisi birebir çaldırılır.
 - o **leastrecent:** Grupta ki en az aranan müşteri temsilcisine çağrıyı yönlendirir.
 - o **fewestcalls:** Grupta ki en az çağrı karşılayan müşteri temsilcisine çağrıyı yönlendirir.
 - o **random:** Gruptan rastgele bir müşteri temsilcisine çağrıyı yönlendirir.
 - o **rrmemory:** Roundrobin mantığıyla çalışır ancak bir önceki seferde en son nerede kaldıysa işleme oradan devam eder.
- **servicelevel :** Servis verilebilirliğinin istatistiksel bilgileri için kullanılır. Belirlenen zaman diliminde cevaplanan çağrı ile cevaplanamayan çağrıya göre hesaplanır.

- **context** : Bu özellik sayesinde tek bir dijital tuşlayan müşteri temsilcisi bulunmuş olduğu kuyruktan alınarak basmış olduğu tuşa ait içeriğe alınması sağlanır.
- **timeout** : Bir çağrının cevaplanana kadar maksimum ne kadar çalacağını belirlediği alan.
- **retry** : Tüm kullanıcıları tekrar denemeye başlamadan önce ne kadar süre bekleneyeceğinin belirlendiği parametre.
- **weight** : Bir kanal birden fazla kuyrukla ilişkilendirildiğinde, öncelikli olarak yüksek ağırlığa sahip olan kuyruğun bu kanal üzerinden çağrıyı karşılaması sağlanır.
- **wrapuptime** : Bir çağrı bittikten sonra diğer çağrının ne kadar süre sonra alınacağını belirlediği alan.
- **autopause** : Eğer kullanıcılar çağrıyı cevaplayamıyorsa otomatik olarak kuyruğun bekletilmesinin sağlandığı parametre.
- **maxlen** : Kuyrukta bekleyebilecek maksimum arayan sayısının belirlendiği alan.
- **announce-frequency** : Arayan kişinin kuyrukta kaçınıcı sırada olduğunun anonsunun ne sıklıkla yapılacağını belirlediği parametre.
- **announce-holdtime** : Pozisyon anonsu sırasında tahmini bekleme süresinin de söylenmesini sağlayan parametre.
- **monitor-format** : Bu parametre iki özelliği bir arada içermektedir. Bunlardan birincisi görüşmenin kaydedilip kaydedilmeyeceği bilgisinin ayarlanması, ikincisi ise görüşmelerin hangi dosya biçiminde saklanacağını bilgisinin ayarlanmasıdır.
- **monitor-join** : Eğer bu özellik aktif edilirse görüşmede bulunan iki ses dosyası (birisi dışarıya giden ses, birisi içeriye gelen ses) otomatik olarak birleştirilerek kaydedilir.
- **joinempty** : Arayan kullanıcının müşteri temsilcisi olmayan bir grubun kuyruğunda bekleyip bekleyemeyeceğini belirleyen parametre. Üç farklı değer alabilmektedir. Bunlar;
 - o **yes** : Arayan kişinin, içinde müşteri temsilcisi olmayan veya müşteri temsilcisi olan fakat o an müsait müşteri temsilcisi olmayan grupların kuyruğunda beklemesine olanak tanır.

- **no** : Arayan kişinin, içinde müşteri temsilcisi olmayan gruba ait kuyrukta beklemesine izin verilmez.
- **strict** : Arayan kişinin, içinde müşteri temsilcisi olmaya gruba ait kuyrukta beklemesine izin vermez ama o an müsait müşteri temsilcisi olmayan grubun kuyruğunda beklemesine olanak tanır.
- **leavewhenempty** : Grubun kuyruğunda yer kalmadığı zaman eski bekleyenlerin kuyruktan kaldırılmalarına olanak tanıyarak yeni gelenlerin kuyruğa alınmalarının sağlar.
- **eventwhencalled** : Herhangi bir olay gerçekleştiğinde **AgentCalled**, **AgentConnect** gibi olay yöneticilerinin çalıştırılmasını sağlar.
- **eventmemberstatus** : Eğer bu parametre “yes” olarak ayarlanırsa **QueryMemberStatus** olay yöneticisinin çalıştırılması sağlanır.
- **reporholdtime** : Bu özellik aktif edildiğinde kuyrukta bekleyen müşteri cevaplandığı zaman cevaplayan müşteri temsilcisine ilgili müşterinin ne kadar süredir kuyrukta olduğunu bilgisini söyler.
- **memberdelay** : Kuyrukta bekleyen müşteri hatta bağlanmadan önce müşteri temsilcisinin eğer bir süre sessizlik duyması isteniyorsa bu parametre de ilgili süre belirtilerek bahsi geçen işlem gerçekleştirilebilir.
- **ringinuse** : Eğer kuyruğun o anda aktif görüşmede olan müşteri temsilcilerine çağrı göndermesi istenmiyorsa bu özelliğin ayarlanması gerekir. (Sadece SIP kanalları kullanımda olup olmadığının bilgisini raporlayabilir)
- **timeoutrestart** : Eğer hattın BUSY veya CONGESTION sinyalleri alınırsa, ilgili sinyale sebep olan müşteri temsilcisinin cevaplama zaman aşımı sıfırlanır. Çağrı reddetme hakkı olan müşteri temsilcilerinde kullanılacak önemli bir özellik.

Bu tanımlamalar incelendikten sonra örnek bir kuyruğun nasıl oluşturulduğu aşağıdaki şekilde görülebilir;

[general]

persistentmembers = yes

```
[queue_test1]
reporholdtime=yes
member => Agent/101
member => Agent/102
member => Agent/103
member => Agent/104
```

```
[queue_test2]
monitor-format=wav
monitor-join
strategy = ringall
```

```
member => Agent/101
member => Agent/102
member => Agent/103
member => Agent/104
```

```
[test]
context=qtest
announce-frequency=5
periodic-announce-frequency=5
announce-holdtime=yes
announce-round-seconds=10
reporholdtime=yes
member => SIP/operator
```

9.3 Asterisk Ağ Geçidi Arayüz Programlaması (AGI)

Asterisk Ağ Geçidi Arayüz Programlaması; Asterisk'in farklı programlama dilleriyle etkileşimde bulunmasını sağlamak amacıyla geliştirilmiş bir arayüzdür. Özellikle AGI programlama da PHP, Perl dilleri yaygın bir şekilde tercih edilmektedir. Bunun yanı sıra C, Pascal gibi bir çok programlama diline de etkileşim

olanağı tanınmıştır. Asterisk 1.6 sürümüyle beraber 5 tip AGI ortaya çıkmıştır. Bunlar;

- **AGI** : Sistemle, veritabanıyla, farklı API 'lerle çeşitli işlemler yapabilmeyi ve çağrı akışını kontrol etmeyi sağlar.
- **EAGI** : Çağrı sırasında kurulan konuşma kanallarıyla etkileşime olanak tanır. Bu sayede ses kaydı yapabilmek yada ses kaydını metine çevirmek (speech-to-text) için gerekli olan ses akışlarının elde edilmesini ve üzerinde işlem yapılabilmesini sağlar.
- **DeadAGI** : Bir çağrı kapatıldığı zaman kapatılan kanala erişim olanağı tanır. Özellikle temizleme işlemlerinde kullanılır. Çağrı sonlandığı zaman kapatılması gereken öğelerin kapatılması gibi işlemlerde kullanılmaktadır.
- **Async AGI** : Eş zamanlı olmayan AGI programlamalarına imkan sağlar.
- **FastAGI** : Ağ bağlantısı sayesinde uzak makineden işlem yapabilme imkanı sağlar.

Tez proje uygulamasında AGI kullanıldığından dolayı diğer türler kısaca tanımlanmaktadır. Asterisk, bir AGI betiğini çalıştırdığı zaman tanımlayıcılarını kendi ön ekini ekleyerek değerlerden kolonla ve boşlukla ayırır. AGI tarafından kullanılan değişkenler şunlardır;

- **agi_request** : Kullanılan AGI Betik'in dosya adı.
- **agi_channel** : Başlatan kanal.
- **agi_language** : Dil kodu (Ör: en)
- **agi_type** : Başlatan kanal tipi (Ör: SIP)
- **agi_uniqueid** : Çağrı için tekil bir kimlik.
- **agi_version** : Asterisk versiyonu
- **agi_callerid** : Arayan kişinin numarası
- **agi_calleridname** : Arayan kişinin görünen adı
- **agi_dnid** : Aranılan kişinin numarası.
- **agi_context** : AGI'nin extensions.conf dosyasından çağrıldığı içerik adı
- **agi_extension** : Aranılan numara

- **agi_priority** : Akış planında belirtilen öncelik.
- **agi_accountcode** : Çağrı başlatan kanalın hesap kodu
- **agi_threadid** : AGI Betik'inin işlem parçacığı (thread) numarası

Burada belirtilen değerler Voip uygulamasında kullanılacak değerlerdir. Bunlar dışında PRI ve Analog uygulamalarda kullanılacak farklı değişkenler de bulunmaktadır.

AGI Betikleri hazırlayabilmek için AGI komutları kullanılmaktadır. Bu komutlar sayesinde Asterisk sunucusuna yapmak istenilen işlemler iletilebilir. Özellikle interaktif sesli yanıt sistemi uygulamalarında AGI betikleri sıkça kullanılmaktadır. [8] Durumlara bakılarak farklı yönergeler oluşturup, akışın belirlenmiş olan şekilde devamlılığı sağlanabilir. AGI komut listesi şu şekildedir;

- **answer** : Çağrıyı cevaplar.
- **asyncagi break** : Eşzamansız AGI döngüsünü durdurur.
- **channel status** : Bağlı olan kanalların durumunu döndürür.
- **control stream file** : Belirtilen dosyanın kontrolüne olanak sağlar.
- **database del** : Veritabanından anahtar/değer tanımlamalarını kaldırır.
- **database deltree** : Veritabanından anahtar ağacı/değer tanımlamalarını kaldırır.
- **database get** : Veritabanı değerini alır.
- **database put** : Veritabanına değer ekler veya varolan değeri günceller.
- **exec** : Belirtilen uygulamanın çalıştırılmasını sağlar. (Uygulama diye extensions.conf dosyasında çağrı planı oluşturmak için kullanılan metotlardan bahsedilir)
- **get data** : Bir kanalda ki veriyi alır.
- **get full variable** : Bir kanalda ki değişkeni alır, karmaşık değer adlarını ve yerleşik değer adlarını anlayabilir.
- **get option** : Kanala zaman aşımı seçeneğini de ekleyerek ses dosyası gönderir.
- **get variable** : Bir kanalda ki değişkeni alır.
- **hangup** : Aktif kanalı sonlandırır.
- **noop** : Hiç bir şey yapmaz.

- **receive char** : Kanallardan bir karakter alır.
- **receive text** : Kanallardan bir metin alır.
- **record file** : Belirtilen dosyaya kaydeder.
- **say alpha** : Verilen karakter katarını söyler.
- **say date** : Tarihi söyler.
- **say datetime** : Biçimlendirilmiş tarih zamanı söyler.
- **say digits** : Verilen dijital katarını söyler.
- **say number** : Verilen numarayı söyler.
- **say phonetic** : Verilen karakter katarını söyler.
- **say time** : Verilen zamanı söyler.
- **send image** : Eğer kanalda desteklerse, kanala resim gönderir.
- **send text** : Eğer kanal desteklerse, kanala metin gönderir.
- **set autohangup** : Belirlenen zaman diliminde kanalı otomatik olarak kapatır.
- **set callerid** : O an ki aktif kanal için arayan kimliğini ayarlar.
- **set context** : Kanal içeriğini ayarlar.
- **set extension** : Kanal uzantılarını ayarlar.
- **set music** : Bekletme müziğini ayarlar, belirlenen müziği oynatır, bekletme müziğini açar/kapatır.
- **set priority** : Kanal önceliklendirmelerini ayarlar.
- **set variable** : Bir kanal değişkenini ayarlar.
- **stream file** : Kanaldan ses dosyası gönderilmesini sağlar.
- **tdd mode** : TDD'lerle iletişimi sağlayabilmek için eğer kanal desteklerse TDD modunu açar/kapatır.
- **verbose** : Bir mesajı Asterisk ayrıntı geçmişinde tutar.
- **wait for digit** : Bir dijital tuşlanmasını bekler.

Bunlar dışında Asterisk 1.6 ile gelen bazı komutlar da vardır. Onlar da şu şekildedir;

- **speech activate grammar** : Dilbilgisini aktif hale getirir.
- **speech create** : Bir konuşma nesnesi oluşturur.
- **speech deactivate grammar** : Dilbilgisini pasif hale getirir.
- **speech destroy** : Bir konuşma nesnesini yok eder.
- **speech load grammar** : Bir dilbilgisi yükler.
- **speech recognize** : Konuşmayı teşhis eder.

- **speech set** : Bir konuşma motoru ayarlar.
- **speech unload grammar** : Dilbilgisini kaldırır.

AGI Betiklerinin yazılabilmesi için geriye sadece programlanacak dilin kütüphanelerinin yüklenmesi kalır. Daha önce de AGI yapan kişiler AGI Betiklerini genellikle PHP ve Perl dillerinde kodlamayı tercih etmektedirler.

Perl kütüphanelerini sisteme kurulduktan sonra AGI Betiklerine örnek olarak kısa bir AGI betiği örneği verilebilir. Örnek senaryo şu şekilde olsun;

Bir güvenlik kontrol uygulaması olsun. Kullanıcıdan alınmış olan parametreyi www.biteknolojist.com adresinde çalışan servise göndersin ve dönen cevaba bağlı olarak AGI betiğinde yönergeler oluşturulsun. Eğer servisten dönen cevap “OK” ise *giris* değişkeninin değeri 1 olarak ayarlanır ve kullanıcıya girişinin başarılı olduğuna dair anons gönderilir. Eğer servisten cevap olarak “ERROR” dönerse de kullanıcının *giris* değişkeninin değeri 0 olarak ayarlanır ve kullanıcıya *hatalı_giris* anonsu gönderilir.

Uygulama kodu şu şekildedir;

```
#!/usr/bin/perl
use Asterisk::AGI;
use LWP::UserAgent;
my $ua = LWP::UserAgent->new;
$ua->timeout(30);
$ua->env_proxy;
$AGI = new Asterisk::AGI;
```

```

my $telNo=$AGI->get_variable('secCode');
my $response = $ua-
>get("http://www.biteknojist.com/gkontrol/Default.aspx?secCode=$telNo");if
($response->is_success) {
    $AGI->set_priority(2);
    my @values = split(';', $response->decoded_content);

    if($values[0] eq "OK")
    {
        $AGI->set_variable("giris",$values[1]);
        $AGI->exec('Playback', 'giris_onaylandi');

    }

    else($response->decoded_content eq "ERROR")
    {
        $AGI-> set_variable("giris",$values[0]);
        $AGI->exec('Playback', 'hatali_giris');
    }
}

else {
    die $response->status_line;
}

exit 1

```

9.4 Internet Telefonu Uygulamaları

Proje de esas olarak SIP protokolü seçildiğinden dolayı hazırlanmış olan telefon uygulamaları da SIP protokolü ile haberleşme yapacak şekilde tasarlanmıştır. Telefon uygulamalarında izlenen işleyiş sırası şu şekildedir;

- 1- Kullanıcı telefon uygulamasını açar ve giriş bilgilerini girer.
- 2- Girmiş olduğu giriş bilgileri sisteme kayıt işlemi gönderilir ve dönecek olan cevap beklenir.
- 3- Eğer başarılı bir işlem olmuşsa kullanıcının sistemde oturum açmasına müsaade edilir. Eğer işlem başarısız olursa, kullanıcının sisteme dahil olamadığına dair bilgi mesajı döner.

Tez projesinin uygulamasında iki adet telefon programı geliştirilmiştir. Bunlardan bir tanesi müşteri temsilcisi tarafından kullanılacak, diğeri ise müşteri tarafından kullanılacak şekildedir. Her iki uygulamada da sistemde oturum açma zorunluluğu bulunmaktadır. Müşteri temsilcisi telefonu sadece gelecek olan çağrıları karşılayacağından dolayı sadece açma ve kapama tuşundan oluşmaktadır. Müşteri telefonunun da çağrı başlatma senaryosunu gerçekleştirebilmesi için dış arama yapabilmesini sağlayacak gereksinimler programda gerçekleştirilmiştir.

9.5 Sesli Yanıt Sistemi Web Servisleri

Web servisleri; firmanın vermiş olduğu servislerini internet üzerinden tüm dünya ile paylaşmasında oldukça yaygın olarak kullanılmakta olan yazılımlardır. Bu yazılımlar aracılığıyla firmalar yapmış oldukları hizmetlerini dış dünyaya kendi belirledikleri şekilde açabilmektedirler. Özellikle web uygulamaları geliştiren firmalarda yaygın olarak kullanılan web servisleri bu tez uygulamasında da kullanılmaktadır. Tez uygulamasında, müşterinin telefondan tuşlamış olduğu tuşların yakalanarak bir web servise gönderilmesi ve web servisten gelen cevaba göre iş akışına devam edilmesi şeklinde bir senaryo planlanmıştır. Kullanıcının sisteme giriş yapabilmesi için gerekli olan güvenlik doğrulamasında, ders seçimi yapabilmesi için derslerin müsaitliğinin kontrolünde, randevu alma işlemi esnasında ilgili öğretim görevlisinin müsaitliğinin kontrolünde ve ders notlarının telefon üzerinden sesli olarak aktarılmasında olmak üzere bir çok alanda bu tez uygulamasında web servisleri kullanılmıştır.

Web servisleri ile telefon tuşlarının etkileşimi sağlanarak kullanıcıların kendi işlemlerini kendi inisiyatifleri doğrultusunda belirleyebilmesi sağlanmış ve böylece gereksiz insan gücü kaybının önüne geçilmiştir. İstenilen biçime göre geliştirilip firmalara uyarlanabilecek bu web servis yapısı sayesinde aynı anda birden fazla çağrı merkezine aynı servisler üzerinden sorgulama yapma izinlerinin verilmesi de sağlanabilir. Bunun yanı sıra özellikle birden fazla kuruma aynı servisi vermekte olan kuruluşların bu tür servislere yönelmesi sıkça görülmektedir.

10. SONUÇ

Bu tezde hedeflenen amaçların nasıl yapılabileceği uygulamalarıyla beraber anlatılmıştır. Teorik anlatılan özellikler aynı zamanda uygulamaya da dökülerek gerçekleştirilebilirliği gösterilmiştir. Çağrı merkezlerinde önemli derecede sorun olan ancak önüne geçilmesi oldukça zor olan önemli konular çözümleriyle beraber ele alınmıştır.

İlk olarak tüm sistemlere mobilite özelliği kazandırılarak gerek personelin gerekse de sistemin konum bağımlılığı minimuma indirgenmiştir. Bu sayede personel giderlerinde önemli oranda tasarrufa gidilmiştir. Çağrı merkezlerinin personel taşımacılığında sorun yaşamalarından dolayı personelleri şirketin konumunun bulunduğu yere yakın yerlerden seçme zorunluluğu da ortadan kaldırılmıştır. Personel taşımacılığına harcanan bedel ve diğer personel giderlerinde bu sayede ciddi oranda tasarruf kazanımı sağlanmıştır.

Bu önemli özellikte yapılan iyileştirmeden sonra giderler de sadece personelden yana değil başka alanlarda da tasarruf kazanımı sağlanmıştır. Çağrı merkezlerinde kullanılmakta olan pbx sunucu yazılımlarında bulunan kullanıcı başına ödenen lisans sorunu da tez uygulamasında kullanılmış olan ücretsiz açık kaynak kodlu pbx sunucu yazılımları sayesinde ciddi oranda azalmıştır. Bunun yanı sıra çağrı merkezlerinde kullanılan donanımsal telefon cihazları yerine tez uygulamasında hazırlanan yazılımsal telefon uygulamaları sayesinde personel başına ödenen telefon cihazı masrafı da ortadan kaldırılmıştır.

Personelin yolda kaybetmiş olduğu zamanın geri kazanımı da sağlanarak etkin zaman kullanımı başarılmıştır. Ayrıca trafik süresi boyunca müşteri temsilcisinde oluşabilecek sinir, stres sorunları da müşteri temsilcisinin sisteme bulunduğu konumdan bağlanması sağlanarak ortadan kaldırılmıştır. Bu özelliğin iyi yanı olduğu kadar zor olan bir yanı da kontrol etme kısmıdır. Bu personeller şirket bünyesinde hizmet vermeyeceklerinden dolayı görüşmelerinin sık sık kontrol edilmesi gerekmektedir.

Altyapı kurulumu çağrı merkezlerinin en büyük sorunları arasındadır. Bu kurulum esnasında kaybedilen zaman oldukça fazladır. Özellikle operatörlerle çağrı merkezleri arasında kurulacak olan hat gruplarının bağlanması 2 hafta gibi bir süreye kadar sürebilmektedir. Bu da çağrı merkezlerinin 2 hafta süresince işlerinin beklemesine neden olabilir. Tez uygulamasında çağrıların internet üzerinden taşınması gerçekleştirilmiştir. Bu sayede sistemin bir hat tahsisi beklemesi zorunluluğu ortadan kalkmıştır ve internet olan herhangi bir konumdan sistem ayaklandırılabilir hale getirilmiştir. Çağrı merkezi personelinin bilgisayar, telefon cihazı gibi ekstra kurulumları da bu süreye daha fazla süre ekleyebilmektedir. Oysa bu tez çalışmasında personelin kendi bilgisayarlarından da sisteme bağlanabilmelerini sağlayan telefon yazılımları sayesinde, sadece bir program kurmasıyla sisteme dahil olabilmeleri sağlanmaktadır. Bu da önemli ölçüde zaman ve para tasarrufu sağlamaktadır.

Çağrıların internet üzerinden taşınması sayesinde diğer bir büyük sorun olan bağımlılık sorunu da kısmen çözülmüştür. Şirketlerin taşınması durumunda en büyük sıkıntıları hatların yeni konumda tekrardan çalışır hale getirmeleri olmaktadır. Oysa bu sistemde çağrıların taşındığı ortamın hat grupları değil de internet olması bu bekleyişin ortadan kaldırılmasını sağlamaktadır.

Kurulan açık kaynak kodlu pbx sunucu yazılımı sayesinde istenildiği zaman sunucu üzerinde şirket planlarına göre geliştirmeler oldukça esnek bir şekilde yapılabilmektedir. Bir çok pbx sunucusunda esneklik yeteri kadar kazandırılmamış ve sisteme müdahale etmenize izin verilmemektedir. Tez uygulamasında kullanılan Asterisk PBX Sunucu yazılımı sayesinde dilendiği takdirde PBX sunucusunun kendisinde de geliştirmeler yapılabilmektedir.

Çağrıların hat gruplarından alınıp internet üzerine taşınmasının bir diğer önemli kazancı da servis verilebilirlik alanının genişletilmesinin sağlanmasıdır. Bir telekom firmasıyla çağrı merkezi arasında çekilen ara bağlantıdan sadece ilgili telekom operatörünün müşteri kitlesi yararlanabilmektedir. Oysa tez uygulamasında çağrıların internet üzerinden taşınması sayesinde istenildiği takdirde dünyanın herhangi bir yerinde bulunan internet kullanıcılarına bile hizmet verilebilmektedir.

Dünyanın en büyük açık ağı olan internet sayesinde servis verilebilirlik kitlesi çok daha büyüktür.

Personellerin farklı konumlardan sisteme bağlanabilmelerinin bir diğer artısı da risk bölüşümünün sağlanmasıdır. Bir afet durumunda personellerden birinin devre dışı kalmasında farklı konumda bulunan personelin sistemde aktif olarak kalabilmesi sağlanabilir ve böylece servis verilebilirlik zamanı arttırılır ve risk bölüşürülebilir.

Sistemde çağruların internet üzerinden taşınmasının sağlanması ve fiziksel bağlantıların oldukça az olması aynı zamanda sistemde afet durumunda çok rahat bir şekilde farklı bir konumda tekrar canlandırılabilmesi sağlanabilmektedir.

Sonuç olarak çağrı merkezi kurulumunun nasıl yapılabileceği, çağrı merkezlerinde sıkça rastlanan sıkıntıların nasıl giderilebileceği, sorunların belirlenmesi ve çözümlerinin nasıl geliştirilmesi gerektiği belirlenmiştir.

11. Kaynaklar

[1] Alan B. Johnston, "Understanding the Session Initiation Protocol", Artech House Telecommunications , 2009

[2] Henry Sinnreich, "Internet Communications Using SIP: Delivering VoIP and Multimedia Services with Session Initiation Protocol", Networking Council, 2006

[3] Gonzalo Camarillo, "SIP Demystified", Paperback, 2001

[4] Jim Van Meggelen & Leif Madsen & Jared Smith, "Asterisk : The future of Telephony, Second Edition", O'Reilly Media Inc. , August 2007

[5] Ted Wallingford, "Switching to VOIP", O'Reilly Media Inc. , June 2005

İnternet:

Elektronik Makale ve Yayınlar

[6] H323 vs SIP : http://www.packetizer.com/ipmc/h323_vs_sip/ (Erişim Tarihi: 21 Mart 2010)

[7] SIP and H323 Comparison : <http://www.sipcenter.com/sip.nsf/html/SIP+and+H.323> (Erişim Tarihi: 21 Mart 2010)

[8] Asterisk IVR Tips : <http://www.voip-info.org/wiki/view/Asterisk+tips+IVR+menu> (Erişim Tarihi: 28 Mart 2010)

[9] G.729 Discussion Platform : <http://groups.google.com/group/asterisk-g729> (Erişim Tarihi: 28 Mart 2010)

[10] Asterisk SIP Conf : <http://www.voip-info.org/wiki/view/Asterisk+config+sip.conf> (Erişim Tarihi: 03 Nisan 2010)

12. Özgeçmiş

Ad Soyad : Mehmet Karabayır
Doğum Tarihi : 11.03.1985
Doğum Yeri : Sivas / Türkiye
Lise : Özel Asfa Ahmet Mithat Lisesi
Üniversite / Lisans : Haliç Üniversitesi Bilgisayar Mühendisliği