T.C.
NECMETTİN ERBAKAN NİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

MOBİL HABERLEŞME GÜVENLİĞİNİN STEGANOGRAFİ İLE ARTTIRILMASI

Celalettin MİSMAN

YÜKSEK LİSANS TEZİ

Endüstri Mühendisliği Anabilim Dalı

Temmuz-2018
KONYA

TEZ KABUL VE ONAYI

Celalettin MİSMAN tarafından hazırlanan "MOBİL HABERLEŞME GÜVENLİĞİNİN STEGANOGRAFİ İLE ARTTIRILMASI" adlı tez çalışması 20/07/2018 tarihinde aşağıdaki jüri tarafından oy birliği ile Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalı'nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Jüri Üyeleri	**İmza**
Başkan Dr. Öğr. Üyesi Barış KOÇER	
Danışman Dr. Öğr. Üyesi Mehmet HACIBEYOĞLU	
Üye Dr. Öğr. Üyesi Şaban GÜLCÜ	

Yukarıdaki sonucu onaylarım.

Prof. Dr. Mehmet KARALI
FBE Müdürü

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Celalettin MİSMAN

Tarih: 20.07.2018

ÖZET

YÜKSEK LİSANS TEZİ

MOBİL HABERLEŞME GÜVENLİĞİNİN STEGANOGRAFİ İLE ARTTIRILMASI

Celalettin MİSMAN

Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü
Endüstri Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Mehmet HACIBEYOĞLU

2018, 61 Sayfa

Jüri
Dr. Öğr. Üyesi Mehmet HACIBEYOĞLU
Dr. Öğr. Üyesi Barış KOÇER
Dr. Öğr. Üyesi Şaban GÜLCÜ

Bilginin gizliliği ve güvenliği insanlık tarihi boyunca temel bir ihtiyaç olmuştur. İnsanlar bu ihtiyacı karşılamak için kendi bedenlerini, dillerini, yazılarını ve buna benzer araçları kullanmışlardır. Günümüzde, steganografi bilginin gizlenme yöntemlerini araştıran bir bilim ve sanat olarak kabul edilmektedir. Temel olarak steganografi örtü nesnesi ve gizli bilgi olmak üzere iki elemandan oluşur. İnsanlar geçmişte bedenlerini, şiirlerini, günlüklerini ve dövmeleri örtü nesnesi amacıyla kullanmışlardır. Günümüzde steganografi teknolojideki gelişmeler sayesinde, yöntemlerini ve çalışma alanlarını genişletmiştir. Bilgisayar biliminde steganografi örtü nesnesi tipine göre yazı, ses, video, resim olmak üzere dört yöntemle sınıflandırılmıştır. Her yöntemin gizli bilgiyi saklama konusunda kendine has, farklı metotları bulunmaktadır. Dijital steganografi yöntemlerinde en popüler ve en çok tercih edilen yöntem en önemsiz bit kodlamasıdır. En önemsiz bit örtü verisinin baytlarının belirli sayıdaki bitlerinin gizli bilginin ikilik gösterimdeki değerleri ile değiştirilmesi olarak tanımlanabilir. Bu tez çalışmamızda kullanıcının herhangi bir dijital resim içerisine anlık gizli bilgi saklamasına olanak sağlayan bir android mobil uygulama sunmaktayız. Resim kullanıcı galerisinden eklenebilir ya da anlık olarak çekilen fotoğraf kullanılabilir. Gizli bilginin resim içerisine saklanmasında resim steganografi yöntemlerinden en önemsiz bit kodlama kullanılır. Bunun yanında, gizli bilgi kodlama aşamasından önce AES şifreleme algoritması ve alıcının kullanıcı adı kullanılarak oluşturulmuş SHA-256 özet değeri anahtarı ile şifrelenir. Kullanıcı oluşturulmuş sonuç resmini alıcı ile sağladığımız Openfire sunucusu üzerinden veya diğer iletişim uygulamaları ile paylaşır. Paylaşılan resim iletilen kişi tarafından uygulaya giriş yapılarak çözümlenir ve gizli bilgiye ulaşılır. Sonuç olarak yalnızca istenilen kişinin görebileceği gizli bilgi iletimi sağlanmış olur.

Anahtar Kelimeler: Android, bilgi güvenliği, mobil iletişim, steganografi

ABSTRACT

MS THESIS

INCREASING SECURITY OF MOBILE COMMUNICATION WITH STEGANOGRAPHY

Celalettin MİSMAN

The Graduate School of Natural and Applied Science of Necmettin Erbakan University
The Degree of Master of Science in Industrial Engineering

Advisor: Asst. Prof. Dr. Mehmet HACIBEYOĞLU

2018, 61 Pages

Jury
Asst. Prof. Dr. Mehmet HACIBEYOĞLU
Asst. Prof. Dr. Barış KOÇER
Asst. Prof. Dr. Şaban GÜLCÜ

Hiding and securing information is a basic demand throughout humanity. People have used their own bodies, languages, writings, etc. to provide information security needs. At the present time, steganography is acknowledged by science and art that researches hiding information methods. Steganography consists of two elements basically that cover and secret information. In the past, people used their bodies, poems, diaries, tattoos for cover. Today, thanks to the developments of technology, steganography has widened its methods and study areas. There are four steganography methods which are text, image, audio and video based on the cover in computer science. All types of steganography methods have distinctive different ways to hide information. Least Significant Bit (LSB) method is the most used and popular method in digital steganography. The LSB is a way that we overwrite the LSB of each byte of the cover (image, video, audio) with binary representation of secret information. In our project, we are developing an android mobile application that allows user to hide a secret information inside any image. The image can be captured instantly or selected from user gallery. We are using LSB image steganography method in order to hide secret information in image. Beside this, we encrypt the secret information with AES encryption algorithm using secret key generated with SHA-256 checksum value of the receiver's username before inserting it in image. The user can share the result image over our Openfire server or other communication applications. The result image is decrypted and the secret message reached by the received user that is logged in the application. At the end, the transmission of the secret information is provided with just desired person.

Keywords: Android, information security, mobile communication, steganography.

ÖNSÖZ

Çalışmalarımın her aşamasında beni sevk eden, destek ve bilgilerini benimle paylaşan Sayın Dr. Öğr. Üyesi Mehmet Hacıbeyoğlu'na, iş yerimdeki IT grup yöneticim Suat Küçükyılmaz'a ve bünyesinde çalıştığım Mepsan Petrol Cihazları San.Tic. A.Ş. ye teşekkürlerimi sunarım.

Ayrıca, destekleriyle her zaman yanımda olan aileme sevgi ve saygılarımı iletirim.

Celalettin MİSMAN
KONYA-2018

İÇİNDEKİLER

SİMGELER VE KISALTMALAR

Kısaltmalar

AES: Gelişmiş Şifreleme Standardı

API: Uygulama Programlama Arayüzü

DCT: Ayrık Kosinüs Dönüşümü

DFT: Ayrık Fourier Dönüşümü

DWT: Ayrık Dalgacık Dönüşümü

JPEG: Birleşik Fotoğraf Uzmanları Grubu

LSB: En önemsiz bit

NIST: Ulusal Standartlar ve Teknoloji Enstitüsü

PVD: Piksel Değer Farkı

SSL: Güvenli Soket Katmanı

TPVD: Üç Yönlü Piksel Değer Farkı

TLS: Taşıma Katmanı Güvenliği

XMPP: Genişletilebilir Mesajlaşma ve Varlık Protokolü

1. GİRİŞ

Bu tez çalışmasında, bir bilgi gizleme sanatı olarak kullanılan steganografinin ne olduğundan, çalışma şeklinden, metotlarından ve günümüzdeki kullanım alanlarından kısaca bahsedilmiştir. Resim steganografisi olarak adlandırılan ve steganografi metotlarından biri olan, dijital resimler veya fotoğraflar içerisine bilgi gizleme yöntemi kullanılarak bir mobil uygulama geliştirilmiştir.

Teknolojinin mobil cihazlar üzerine yaptığı genişleme, mobil cihazları günlük hayatta vazgeçilmez bir araç yapmıştır. İnsanlara yer ve mekândan bağımsız olarak iletişim kurabilme, bilgiye erişebilme, resim, dosya, veri paylaşabilme, video izleyebilme imkânı sağlamıştır. Geliştirmiş olduğumuz STEGMIS isimli uygulama bir mobil işletim sistemi olan android platformunda çalışabilecek özelliklerde, Java programlama dili kullanılarak oluşturulmuştur.

Kullanıcı uygulamaya kullanıcı adı ve şifre ile kayıt olduktan sonra android cihazın galerisinden seçtiği bir resmi ya da anlık olarak çektiği fotoğrafı gizli bilgiyi saklamak için kullanabilir. Resim veya fotoğrafın seçilmesinden sonra kullanıcı resim içerisine saklanmak istediği mesajı girer. Girilen mesaj AES simetrik şifreleme algoritması ile şifrelenerek resim içerisine gömülür. Kullanıcı yerleştirme sürecinden sonra elde edilen resmi android cihazında saklayabilir, başka uygulamalar ile ya da kullanıcı adını bildiği başka bir STEGMIS kullanıcısı ile paylaşabilir. Böylece, resim olarak görülen verilerle aslında gizli mesajlar ve bilgiler barındıran güvenli bir depolama ve haberleşme kanalı oluşturulmuştur.

Gizli bilgi yerleştirilmiş resimlerle orijinal resimler arasındaki değişiklikler dışarıdan bakıldığında anlaşılmayacak şekildedir. Fakat dijital verilerin herhangi bir steganografi metodu kullanılarak bir işlemden geçirilip geçirilmediğini inceleyen steganaliz adında bir bilim dalı bulunmaktadır. Her ne kadar steganografi işlemi ile resim içerisine yerleştirilmiş gizli mesajı ortaya çıkarma steganalizin çalışma alanına girmese de resim içerisine saklanmış bir bilgilinin var olduğunun anlaşılması durumunda o bilginin nasıl yerleştirildiği öğrenilmeden gizli mesaja ulaşılamayacaktır. Herhangi bir şekilde gizli mesaj istenmeyen başka kişiler tarafından öğrenilmesi durumunda ise, ulaşılan veri şifrelenmiş bir veri olması hasebiyle, o kişilerin gizli mesaja ulaşabilmesi için bir de bu veri çözebilmesi gerekmektedir. Bu uygulama ile steganografinin sadece bilgi gizleme sanatı olarak kullanılmasının yanında bir iletişim ve depolama yöntemi olarak da kullanılabileceği gösterilmiş olacaktır.

2. KAYNAK ARAŞTIRMASI

Gizli iletişim sanatı olan steganografi, Yunancada örtü, taşıyıcı anlamına gelen "stegos" ve yazı manasında kullanılan "grafia" kelimelerinden türetilip "örtülü yazı" olarak tanımlanmaktadır. İletişim insanlığın varoluşundan beri temel bir ihtiyaçtır. İletişimin olduğu her alanda gizli ve güvenli iletişim de bir gereksinim halini alır. Bu noktadan sonra da her ne kadar steganografi ismiyle bilinmese de ve günümüzdeki araştırmalarla o kadar eskiye gitmek mümkün olmasa bile, iletişim var olduğundan beri steganografi de vardır.

Tarihte ulaşılabilen ilk örneği Yunan tarihçi Herodot'un bir asilzade olan Histaeus'un damadı ile haberleşmesinde görürüz. Antik Yunan'da bulunan Histaeus damadı ile iletişim kurarken güvendiği kölelerinin birinin saçını tıraş eder ve iletmek istediği bilgiyi kölesinin kafa derisine dövme yapar. Döve, yani gizli bilgi görünemeyecek derecede saçı uzayan köle o şekilde gönderilir (Aubrey de S´elincourt 1996).

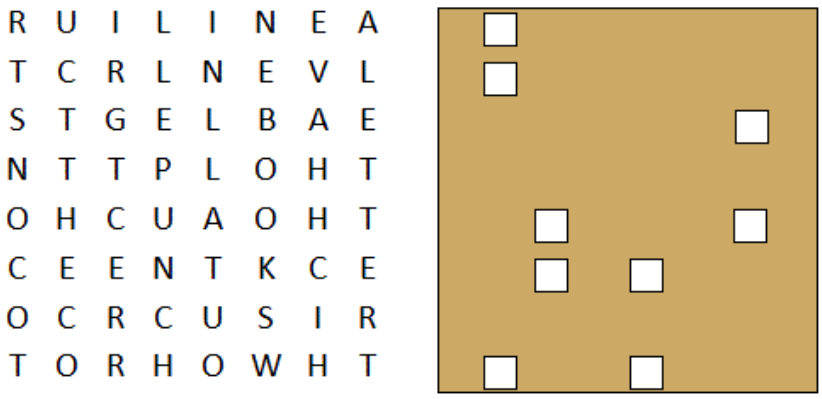
İlk Yunan yazarlarından olan ve askeri iletişimde güvenliğin nasıl sağlanması gerektiğini ortaya koyan Aeneas Tacticus, birçok steganografik teknik ortaya koymuştur. Fakat bu teknikler o zamanda bir sanat olarak algılanmıştır. Kadınların küpelerine gizli mesajlar saklama ve güvercinlerle gizli mesajların taşınması bunlardan bazılarıdır (Tacticius 1990).

Dil biliminde de steganografinin birçok alanda kullanıldığını görürüz. En bilinen yöntem ise akrostiş dediğimiz kelimelerinin ilk harfleri ya da cümlelerin ilk kelimeleri birleştirildiğinde gizli mesajın ortaya çıktığı yöntemdir. Bu yönteme genellikle şiirlerde rastlarız. Akrostiş yöntemini aynı zamanda birinci dünya savaşına katılan ülkeler de kullanmıştır (Wilkins 1954, Corliss 1993). Bunu bir Nazi ajanının notuyla örneklendirebiliriz;

"Apparently neutral's protest is thoroughly discounted and ignored. Isman hard hit. Blockade issue affects pretext for embargo on by-products, ejecting suets and vegetable oils."

Gönderilen mesajdaki her kelimenin ikinci harfini alıp birleştirdiğimizde **"Pershing sails from NY June 1"** gizli bilgisine ulaşırız (Johnson ve Jajodia 1998).

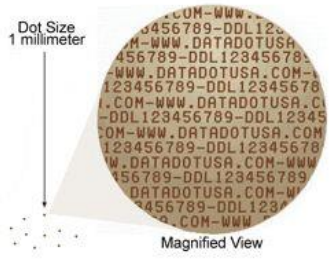
Dil biliminde bilinen diğer bir steganografi metodu ise Çin'de tasarlanmıştır. Temel olarak Çin'de tasarlanan bu metot diğer akrostiş metotlara göre daha gelişmiştir. Cardan tarafından yeniden incelemeye alınmış ve günümüzde Cardan Grille olarak bilinmektedir. Bu metotta harfler standart bir duruma göre oluşturulmak yerine rastgele gizlenir. Gizli mesajı okumak için, kişiler arasında daha önceden paylaşılmış anahtara ihtiyaç vardır.



Şekil 2.1. Cardan Grille ve Anahtarı

Modern steganografik metotların öncüsü olarak Francis Bacon kabul edilmektedir. Bacon çalışmalarında harflerin ikili tabandaki karşılıklarını kodlamak için italik veya normal yazı stillerini kullanmıştır. 16.yy'da kullanılan yazı tipi çeşitlerinin fazla olması bu metodun fark edilmeden kullanılabilmesine olanak sağlamıştır (F. Bacon 1640). Daha sonra ise Brassil ve arkadaşları yazıyı anlaşılamayacak şekilde 300 de 1 gibi bir oranla aşağı ya da yukarı kaydırarak mesajı gizleme tekniğini geliştirmişlerdir (Brassil ve ark. 1995).

19. ve 20. yüzyıldaki savaşlarda kullanılan ve Brewster (1857) tarafından önerilen diğer bir teknik de microdot tekniğidir. Bu teknikte gönderilmek istenilen bilgi küçültme veya parçalara bölünme işleminden geçerek örtü nesne içerisinde gizlenir ve bilgi bu şekilde periyotlar halinde aktarılır. Küçültülen resimler burun deliklerinde, kulaklarda veya tırnaklarda saklanabilir. 1. Dünya Savaşı ile birlikte Almanlar microdot tekniğini yoğun bir şekilde kullanmışlar ve resim gönderecek kadar geliştirmişlerdir.



Şekil 2.2. Microdot Tekniği

Yakın bir zamanda Vietnam'da esir düşen ABD'li Amiral Jeremiah Denton, haber kanalına konuşması sırasında gözlerini kırparak Mors alfabesi ile "T-O-R-T-U-R-E" kelimesini kodlamıştır.

Dijital dünyadaki gelişmelerle birlikte steganografinin de kullanım amacı ve yöntemlerinde değişme ve ilerlemeler olmuştur. Günümüzde steganografinin çalışma alanları dijital resim, ses, video, yazı gibi örtü verilerine kaymıştır. Gerçek dünyaya kıyasla sanal dünyadaki olanakların genişliği ve hayal gücü, steganografide kullanılan metotların gelişmesine imkân sağlamıştır.

Yazı steganografisinde Bender (1996), kelimelerin arasına bir veya iki boşluk koyarak bunlar ile bir ve sıfır bitlerini kodlamıştır. Fakat bu yöntem bakıldığında kolayca anlaşılabilecek bir yöntemdir. Nasereddin ve Abdallah (2016) bu yöntemi geliştirmek amacıyla, gizli mesajların her bir harfinin örtü nesnesindeki yerlerini bularak o yerlerin sırasını bir dizide tutmuşlardır. Örtü verisinde olmayan değerler veya büyüklük ve küçüklük açısından uyuşmayan harfler için, o harflerin ASCII kodlarının karşılıklarını diziye eklemişlerdir. Bu dizi elemanlarını sekizlik tabana çevirip dizi elemanları arasını 9, ASCII kodlarını 8 rakamlarıyla belirtip 1pt büyüklüğünde ve yazının arka plan renginde kelimelerin arasına kodlamışlardır. Bu şekilde, bakıldığında algılanması oldukça zor olan kelimeler arası boşluklar oluşmuştur.

Ses steganografisi alanında Khan ve arkadaşları (2011) gizli bilgiyi ilk olarak AES algoritması ile şifreleyerek bir ses dosyasının içerisine koymayı önermişlerdir. Bu yöntemde gizli bilgi şifrelenip stego nesnesi oluşturulduktan sonra, internet üzerinden gönderme işlemi yapılmadan önce stego-ses nesnesi Yayılma Spektrumu (Spread Spectrum) ile ikinci bir şifreleme işlemine tabi tutulur.

Resim steganografisinde en çok kullanılan yöntem en düşük bite veya bitlere veri saklama işlemidir. LSB yöntemi olarak bilinen bu yöntem sadece dijital resimlere veri saklama işleminde değil, aynı zamanda dijital ses ve video steganografilerinde de kullanılan bir metottur.

Juneja ve Sandhu (2013) bilgi güvenliği konusunda dirençli bir yöntem önermişlerdir. Bu yöntemde mavi ve resmin köşe bölümlerindeki rastgele seçilmiş yeşil rengin bir bölümü olmak üzere iki elemanlı LSB uygulamaktadır. Uyarlanan diğer LSB tabanlı steganografi yönteminde ise gizli bilgi yumuşak alanlardan rastgele seçilen piksellerin mavi, kırmızı, yeşil elemanlarının en önemli bitlerine kodlama işlemini önermektedir.

Yang ve arkadaşları (2009) resimler için LSB tabanlı, uyarlanabilir veri gizleme metodu geliştirmişlerdir. Stego-resimleri için daha kaliteli bir görsel sağlamak amacıyla, veri saklama işleminde alanların kirlilik oranı dikkate alınmaktadır. Önerilen metot köşeleri, parlaklıkları, örtü resminin doku maskesini analiz ederek gizlenecek *k*-bit sayısını hesaplar. *k* değeri resimdeki hassas olmayan bölgelerde yüksek, yüksek hassasiyet içeren resimlerde ortalama kalitedeki resimlere göre düşüktür. Bu LSB metodunda *k* değeri resmin yüksek sıralı bitlerine göre hesaplanır. Bununla birlikte resmin görsel kalitesini arttırmak amacıyla piksel düzeltme yöntemi de kullanılır.

Bir diğer LSB tabanlı resim steganografi yöntemi Channalli ve Jadhav tarafından 2009 yılında önerilmiştir. Veri saklamada temel şablon bitleri (stego-anahtar) kullanılmıştır. Resimlerin en önemsiz bitleri anahtar bitleri ve gizli mesaj bitlerine bağlı olarak değiştirilir. Şablon bitleri rastgele üretilen sayı ile satır ve sütunun oluşturduğu MxN genişliğinin birleşimidir. Gizleme işleminde her bir şablon gizli mesaj bitleri ile karşılaştırılır. Uygun olması durumunda, resmin en önemsiz ikinci seviyedeki biti değiştirilir, olmaması durumunda ise aynı kalır. Bu metot ile mesaj saklama kapasitesi MxN bloğundaki piksel yalnızca bir bit saklama yapılabildiği için oldukça düşüktür.

Laskar ve Hemachandran yönteminde, gizli veri resmin kırmızı düzeyine gömülürken, piksel rastgele üretilen bir sayı ile seçilir. Resimdeki değişiklikleri fark etmek neredeyse imkansızdır. Piksel konumlarını seçmek ve rastgele üretilen sayıları saklamak için bir stego anahtarı kullanılır. Bu çalışma ile mesaj güvenliğinin arttırılması ve bozulma oranının azaltılması hedeflenmiştir.

Sharmila ve arkadaşları renkli resimler (JPEG) üzerinde çalışan bir algoritma geliştirmişlerdir (2012). Bu yöntemde dirençliliği arttırmak amacıyla veri saklama yeri olarak köşeler seçilmiştir. Bu alanlar keskin geçişlerinin olduğu, daha karmaşık istatiksel özellikler sağlayan ve resim içeriği ile oldukça bağlantılı köşelerde konumlandırılmıştır. Keskin köşelerdeki değişiklikleri gözlemek yumuşak geçişlerden daha zordur. Bilgi saklama işleminde RGB elemanları ayrıştırılarak, bir simetrik anahtar ile bir veya daha fazlası seçilir. Örtü resmi tekrarsız bölümlere ayrılır. Her bölüm gizli

anahtar ile rastgele üretilen bir derece ile döndürülür. Sonuç resmi bir satır vektörü olarak taramalı arama yöntemi ile tekrar düzenlenir. Gizli mesaj şifrelenir ve LSBMR yöntemi ile her saklama alanına iki bit olacak şekil gizlenir. Gizli mesaj, mesaj saklama verimi eğri ile hesaplandıktan sonra gömülür.

2013 yılında Hemalatha ve arkadaşlarının çalışmasında, 128x128 boyutlarında gri tonlamasına sahip iki resim gizli resim olarak kullanılmış ve gizleme işlemi RGB ve YC_bC_r alanlarında yapılmıştır. Stego resimler PSNR değerleri ile karşılaştırıldığında RGB tabanında daha kalitelidir. Yazarlar gizli resimleri renkli örtü resimlerine saklamak için tamsayı dalga dönüşümü olan IWT kullanmışlardır. Gizleme işleminden sonra RGB ve YC_bC_r alanlarındaki PSNR değerleri ile resim kaliteleri karşılaştırılmıştır.

Diğer bir DCT tabanlı veri saklama metodunda, renk bilgisi sıkıştırılmış gri seviye resimde saklanır. Bunu renk nicelemesi, renk sıralaması ve gizli veri saklama işlemi takip ederek resim steganografi yöntemi tamamlanır. Bu yöntem, gri seviye resimlere erişimi geniş bir şekilde herkese verirken, renkli resimlere erişimi sadece stego-anahtarı olan kişilerle kısıtlamaktadır. Bu yöntem yüksek PSNR'ye sahip olmakla birlikte veri saklama mimarisi olarak dikkate değer bir araştırmadır (Chaumont ve Puech 2006).

Bir resim üzerinde LSB metodu kullanılarak herhangi bir bilgi saklama işlemi yapılıp yapılmadığının istatiksel steganaliz yöntemler ile anlaşılıyor olması insanları farklı veri saklama yöntemleri keşfine yöneltmiştir. 2003 yılında Wu ve Tsai resim steganografisi için yeni bir yöntem olan PVD metodunu önermişlerdir. Bu metotta resmin gri değerleri değiştirilip, değiştirme işlemine kullanıcının daha az odaklandığı resmin köşelerinden başlanılır. Örtü verisinin aynı olmayan ardışık piksel değerleri hesaplanır. Bu değerler arasındaki farklar bulunarak bir aralık dizisi oluşturulur. Bu oluşturulan diziden seçilen iki değer arasındaki farkın gizlenecek bitler dizisinin ondalık değerinden küçük olma şartı gözetilir. Önceki fark değeri olması gereken yeni farka değeri ile güncellenmelidir. Bunun sağlanabilmesi için de seçilen pikseller değerleri arasındaki farklar yeni farkı sağlayacak şekilde güncellenir.

Dijital steganografi uygulamalarının özelliklerinden bir tanesi de birden fazla tekniğin aynı örtü verisi üzerine uygulanabilir olması durumudur. 2005 yılında Raja ve arkadaşları herhangi bir resim üzerine LSB metodu ile veri saklama işlemi yapıldıktan sonra DCT metodu kullanarak resmin uzaysal alanını frekans alanına dönüştürmüşlerdir. Böylece aynı örtü verisi üzerinde iki farklı steganografi metodu birleştirilerek uygulamışlardır.

Video steganografi yöntemleri resim steganografisi ile karşılaştırıldığında, birçok saklama algoritması içermesine rağmen video steganografisi alanında daha az çalışma yapıldığı görülür. 2009 yılında Eltahir ve arkadaşları en önemsiz bit yöntemini temel alan yüksek oranlı video akış steganografi sistemi sundular. Anlık resimlere uygulanan bu metot ile birlikte resimlerin 33.3%'ne kadar bilgi saklanabilmesine olanak sağlanmıştır. Önerilen yöntemdeki fikir, 24 bitlik resimlerdeki RGB değerlerinin en önemsiz bitlerinin kullanılması olarak tanımlanan 3-3-2 yaklaşımıdır. Bu yaklaşımda kırmızı renk ve yeşil renk değerinin en önemsiz 3 biti, mavi rengin ise en önemsiz 2 biti kullanılarak mesaj saklama işlemi yapılır. Mavi renkten daha az bitin kullanılmasının sebebi insan görme algısının mavi renge kırmızı ve yeşile göre daha hassas olmasıdır. Sonuç olarak her piksele bir baytlık veri saklanmış olur.

2014 yılında R. Shanthakumari ve Dr.S. Malliga AVI formatında bir videoda LSB eşleme yöntemi ile yeniden tasarlanmış bir video steganografi yöntemi sundular. Bu çalışmada ilk olarak örtü videosu karelerine ayrıştırılır. Mesaj birden fazla kareye saklanabilir. Bu da mesaj uzunluğundan bağımsız bir yöntem özelliği kazandırmaktadır. Birden fazla kareye mesaj saklama işlemi yapıldıktan sonra, kareler stego-video olarak gruplanır. Alıcıya gönderilen stego-video tekrar karelerine bölünerek gizli bilgi ayrıştırılır. Önerilen bu metodun iki problemi bulunaktadır. Bunlar güvenlik zafiyetinin yüksek olması ve mesaj saklama kapasitesinin düşük olmasıdır.

Kapoor ve Mirza 2015 yılında LSB tabanlı geliştirilmiş video steganografi sistemi önermişlerdir. Bu yöntemde video karelerine ayrıştırılır, daha sonra piksellerin 24 bitlik RGB değerleri (0…255, 0…255, 0…255) formatına dönüştürülür. Video içerisine gömülecek olan yazı dosyası ZIP ile sıkıştırılarak bayt verisine dönüştürülür. Dönüştürülen bitler ile piksel değerleri kaynaştırılarak stego-video nesnesi oluşturulur ve alıcıya iletilir. Alıcı aldığı videoyu sırasıyla karelerine ve sıkıştırılan dosyanın baytlarına ayrıştırarak gizli bilgiye ulaşır. Önerilen metot MPEG video formatı için geliştirilmiş olup, bazı düzenlemelerle AVI, 3GP video formatlarında da uygulanabileceği söylenmektedir.

Steganografinin mobil uygulama üzerine uygulanma çalışmalardan biri AUCH'tur. Gerçek zamanlı sohbet uygulaması olan AUCH, istemci sunucu mimarisine sahiptir. "AUdio CHat" yani "Sesli Sohbet" kelimelerinden esinlenilerek isimlendirilen AUCH bir programlama dillerinden Java ile geliştirilmiştir. Kullanıcıların ses ve yazı ile iletişim kurmasına olanak sağlayan bu uygulama, ses kaydı sırasında gerçek zamanlı olarak sesin arkasına veri gizleme işlemi yaparak ses verisi karşı tarafa iletmektedir.

Böylece normal bir şekilde yazılı ve sesli sohbet izlenimi veren uygulama ile aslında iletilmek istenilen gizli bilgiler dikkat çekmeden ses steganografisi kullanılarak güvenli bir iletişim kanalı sağlamaktadır (Ali Tarık Gürkan 2016).

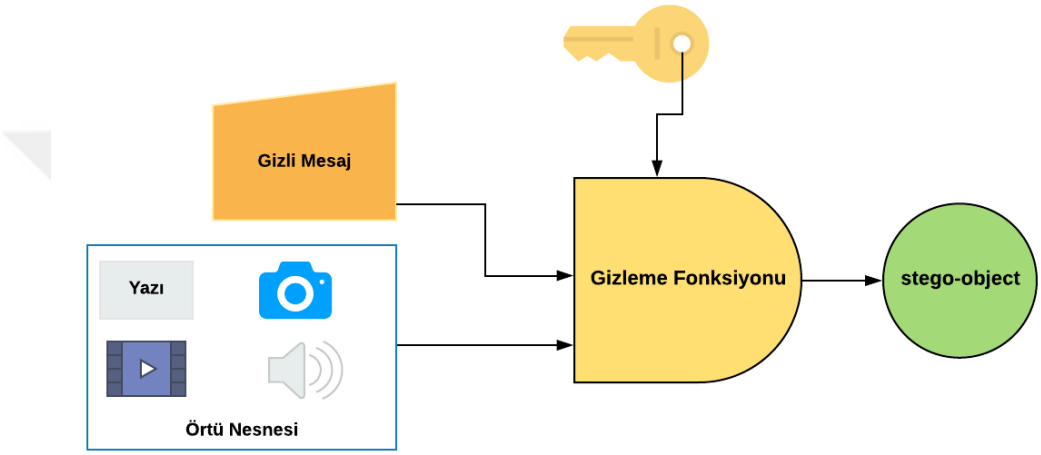
Steganografinin kullanım alanlarından biri de damgalama (digital watermarking) işlemidir. Ülkelerin paralarının sahte olup olmadığının kontrolleri ve bazı şirketler logolarında yetkisiz kullanımların takibini bu şekilde sağlamaktadır. Aynı zamanda; akıllı kimlik kartlarında kişilerin kimlik bilgilerinin resim içerisinde saklanması, sağlık alanında hastanın bilgilerinin resmi içerisine gizlenmesi gibi alanlarda da steganografinin kullanıldığı görülmektedir.

Örtü nesnelerinin içerisine gizli bilgilerin saklanması sanatının yanında, dijital örtü verilerine herhangi bir steganografi yöntemi ile gizli bilgi kodlama işlemi yapılıp yapılmadığını inceleyen steganaliz bilim dalı bulunmaktadır. Genel olarak steganaliz metotları, dijital örtü verisine LSB metodu ile bir bilgi gizlenip gizlenmediğini inceler. İnceleme işlemi sırasında ve sonuca ulaşma aşamasında; model, bit dizileri, yakın renk ikilileri ve benzer resimlerin karşılaştırılması sonucunda elde edilen denklemlerden, histogramlardan, istatiksel verilerden yararlanmaktadır.

3. MATERYAL VE YÖNTEM

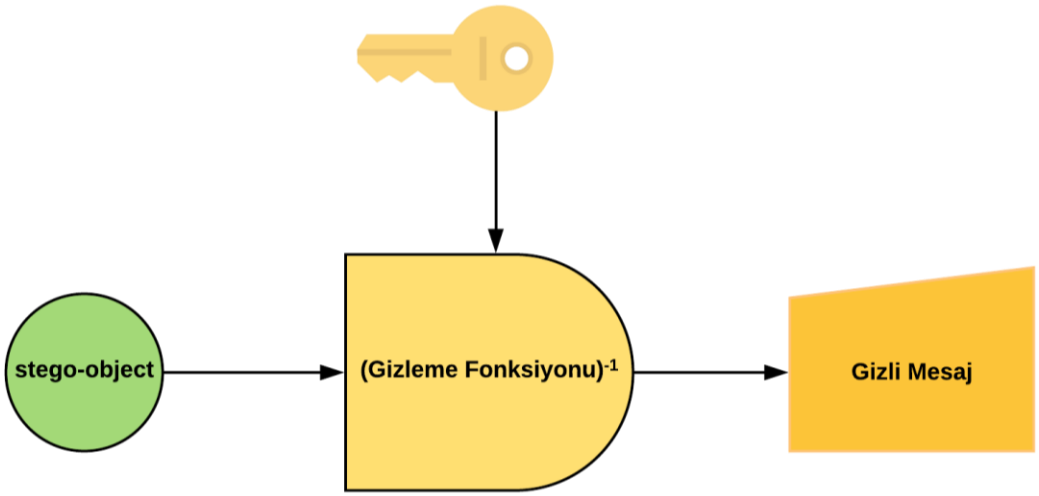
3.1. Steganografi Yöntemleri

Genel olarak bakıldığında dijital steganografi yöntemleri örtü verisine göre dört ana başlık altında sınıflandırılabilir. Bunlar yazı, resim, ses ve video steganografileridir. Steganografinin genel çalışma stratejisi olarak stego nesnesi(stego-object) oluşturulması Şekil 3.1.1'de gösterilmiştir.



Şekil 3.1.1. Steganografi genel çalışma stratejisi, stego nesnesi oluşturulması

Oluşan stego nesnesinden yeniden gizli mesajın elde edilmesi ve ortaya çıkarılması için gizli anahtar ile birlikte gizleme fonksiyonunun tersi uygulanır. Bu işlem Şekil 3.1.2 de gösterilmiştir.



Şekil 3.1.2. Steganografi genel çalışma stratejisi, stego nesnesinden gizli mesajın elde edilmesi

3.1.1. Yazı steganografisi

Yazının insanlık tarihinin başından beri mevcut olması hasebiyle, yazı steganografisi çok özgün ve gelişmiş yöntemlere sahiptir. Yazı steganografisi resim, ses ve video steganografi metotları ile kıyaslandığında algılanması daha kolaydır. Çünkü yeni bir harfin eklenmesi ya da harfe vurgu yapılması sıradan bir okuyucu tarafından dahi algılanabilir (W. Bender ve ark. 1996). Yazı dosyalarının saklanması diğer steganografi metotlarına binaen diskte daha az yer kaplar ve iletişimde daha kolay ve hızlıdır (M. H. S. Shahreza ve M. S. Shahreza 2006). Yazı steganografisi üç ana başlık altında incelenebilir (Monika Agarwal 2013).

3.1.1.1 Yazı formatında uygulanan yöntemler

Bilgiyi gizlemek için yazının fiziksel görünümünde yapılan değişikliklerdir. Stego dosyası kelime düzenleyici programlar (Word, Open Office, vs.) kullanılarak ile görüntülendiğinde harf hataları, kelimeler arasındaki fazladan konulan boşluklar ortaya çıkacaktır. Yazı boyutundaki değişiklikler de kaçınılmaz şekilde okuyucunun gözüne çarpabilir. Bu tarz problemli durumlar göz önüne alındığında yazı formatında uygulanan steganografi çalışmaları üzerinde yoğun şekilde düşünülmesi gerekir.

3.1.1.2 Rastgele ve istatiksel yöntemler

Yazı steganografisinde kullanılan örtü metinlerinin karşılaştırılmasını önlemek amacıyla, kullanılan örtü metinlerinin üretilmesidir. Bu üretim ise iki yolla olur. Birincisi rastgele oluşturulan karakter dizileridir. Gizli bilgi oluşturulan bu diziler içerisinde saklanır. Diğer yöntemde, kelime uzunlukları ve harf tekrarlarının istatiksel özellikleri kelime oluşturulmasında kullanılır. Aynı istatiksel özelliklere sahip kelimelerle asıl bilgiye ulaşılır (Monika Agarwal 2013).

3.1.1.3 Dilbilim yöntemler

Dilbilim yöntemleri yazının formatındaki değişiklikler yerine; yazının dil yapısı üzerindeki değişimler ile bilgi saklanmasını dikkate alır. Bu yöntemlerde örtü yazısında görsel değişiklikler yapılmadığı, dilin yapısı ve kelime seçimleri ön plana çıktığı için

fark edilmesi diğerler yöntemlere göre daha zordur. Bir yazı metnindeki ‘A’ harflerinin ‘0’, ‘B’ harflerinin ise ‘1’ olarak kodlanması bu yöntemlere örnek olarak verilebilir.

Satır kaydırma, kelime kaydırma, sözdizimi yöntemi, spam metin, kelime haritası, özellik kodlama akrostiş, günümüzde yazı steganografisinde kullanılan yöntemler arasındadır.

3.1.2. Ses steganografisi

Dijital seslere normal olarak dinleyen bireylerin anlayamayacağı şekilde gizli bilgilerin saklanması işlemidir. Örtü verisi olarak WAV, AU ve MP3 uzantılı dosyalar kullanılabilir. Ses steganografisinde kullanılan başlıca yöntemler;

- LSB kodlama
- Faz kodlama (Phase coding)
- Eş kodlaması (Parity coding)
- Yayılma spektrumu (Spread spectrum)
- Yankı gizleme (Echo hiding)

3.1.2.1 LSB kodlama

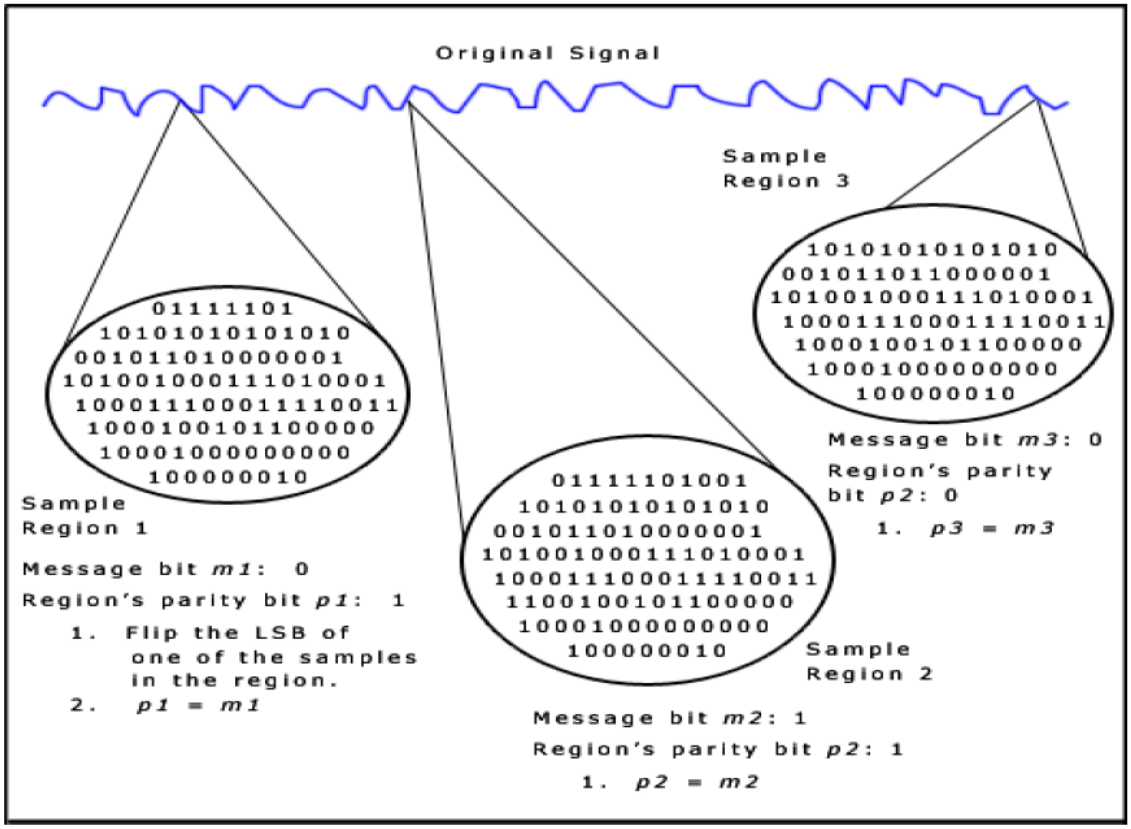
Bu metot veri gizleme konusunda bilinen en eski metottur. Geleneksel olarak, mesaja ait her bir bitin örtü ses verisinin en önemsiz bitine gömülmesi şeklinde tanımlanmıştır. Bu yüzden 16 kHz örnek ses örtü nesnesine 16 kbps veri saklanabilir. LSB metot yüksek oranda veri saklanmasına olanak sağlar ve diğer gizleme metotları ile beraber kullanılabilir. Buna karşın gürültü eklenmesi basit ataklara bile güvenlik zafiyeti oluşturmaktadır. Filtreleme, gürültü ekleme, genişletme ve sıkıştırma işlemlerinde veri kaybının yaşanması muhtemeldir. Bunun yanında veri gömme işlemi, belirli bir yöntemle sağlandığı için, saldırganın örtü verisinden gizli bilgiye ulaşması oldukça kolaydır. Bu metotla yüksek mesaj saklama oranı sağlansa da veri güvenliği ve sağlamlığı konusunda oluşturduğu tehlike için aynı şeyleri söylemek mümkün değildir.

3.1.2.2 Faz kodlama (Phase coding)

Faz kodlaması ses steganografisinin güçlü tekniklerinden bir tanesidir. Faz kodlaması insanın farklı spektral bileşenlerin ilgili fazlarına olan işitsel hassasiyet eksikliğinden yararlanır. Seçilen faz bileşenleri yerine veri saklanmış orijinal ses dalgası spektrumunun değiştirilmesi esasına dayanır. Dinleyenin işitmesinin önüne geçilmesi için faz bileşenlerinde yapılan değişikler küçük tutulmalıdır (Gang ve ark. 2001). Ses içerisine veri gizleme teknikleri düşünüldüğünde faz kodlama tekniği sinyal bozulmasını en iyi dirayet gösteren tekniktir (W. Bender ve ark. 1996).

3.1.2.3 Eş kodlama (Parity coding)

Eş kodlama tekniğinde ses sinyali özgün örneklere bölmek yerine sinyal farklı alanlara ayrıştırılır. Gizli mesajın her bir biti ayrılan alanın eşlik biti olarak kodlanır. Eğer seçilen alanın eşlik gizli mesajın biti ile uyuşmazsa, seçilen alandaki bir bit LSB yöntemi ile değiştirilir.



Şekil 3.1.3. Eş kodlama (Jayaram ve ark. 2011)

3.1.2.4 Yayılma spektrumu (Spread spectrum)

Yayılma spektrumu olarak geçen yöntemde, gizli bilgi spektrum frekansı boyunca yayılır. Yayılma spektrumu, veri iletişiminde uygun bir düzeltme sinyalinin gürültülü kanaldan gönderimi işlemidir. Bu işlem üretilen gereksiz veri sinyali kopyalarının bu kanaldan gönderilmesi ile sağlanır. Temel olarak veri gönderici ve alıcının bildiği M dizisi kadar çarpılır (Khan 1984).

3.1.2.5 Yankı gizleme (Echo hiding)

Yankı gizleme metodunda gizli bilgi temel ses sinyalinin içerisine kısa yankılar tanımlanarak oluşturulur. Yankı doğası gereği temel ses sinyaline eklenen bir tınlamadır. İnsan işitsel sistemindeki hassasiyetten ötürü tını yüksekliğinden kaçınılmalıdır. Yankı eklendikten sonra, stego sinyali istatistiksel ve algısal özelliklerini korur. Düşük bilgi gömülebilme oranı ve güvenlikten dolayı yakında zamanda yankı gizleme yöntemi ile hiçbir çalışma yapılmamıştır.

3.1.3. Video steganografisi

Video resim ve ses elemanlarının birleşimi ile oluşan bir veri türüdür. Bu nedenle dijital resim veya ses dosyalarına uygulanan steganografi yöntemlerinin çoğu video steganografisinde de kullanılabilir. Video steganografisi resim steganografisi ile karşılaştırıldığında üzerinde daha az araştırma yapıldığı bilinmekle beraber, video steganografisinin birçok yararı vardır (Al-Frajat ve ark. 2010). Dijital video steganografilerinde kare hızı yüksek olmasından dolayı, videonun herhangi bir steganografi işleminden geçip geçmediğinin ve gizli mesajın anlaşılması oldukça zordur. Gizli mesaj H.264, Mp4, MPEG, AVI ve benzeri video dosyalarına gizlenebilir. Video steganografisinde çeşitli teknikler bulunmakla beraber en iyi teknik, gizli bilgi video içerisine gömülmesi sağlandığında, çıplak gözle bakan kişi video kalitesinde herhangi bir değişiklik fark etmemelidir (Bhaumik ve ark. 2009). Video steganografisinde kullanılan bazı yöntemler aşağıda belirtilmiştir.

- LSB kodlama
- Düzensiz dikdörtgen bölümlemesi (Non - Uniform rectangular partition)

- Maskeleme ve filtreleme yöntemleri
- Sıkıştırma video steganografisi (Compressed video steganography)

3.1.3.1 LSB kodlama

LSB tekniğinin veri koruma işleminde basitliği ve genel kullanımı açısından en iyi metot olduğu söylenir. Veri gömme işleminin en basit ve performanslı yöntemidir. LSB de, örtü video nesnesinin pikseller baytlarına ayrıştırıldıktan sonra, baytların en önemsiz bitleri ile gizli bilginin bitleri yerleri değiştirilerek veri saklama işlemi gerçekleştirilir. Ana videonun sadece en önemsiz bitlerini değiştirildiğinden dolayı video neredeyse orijinalinden farksızdır.

3.1.3.2 Düzensiz dikdörtgen bölümlemesi (Non - Uniform rectangular partition)

Bu metot sıkıştırılmamış videolar içindir. Düzensiz dikdörtgen bölümlemesi tekniğinde, sıkıştırılmamış bir video ana video içerisine saklanır. Gizlenecek video ve örtü videosunun aynı boyutlarda olmasına dikkat edilmelidir. Bu teknikte örtü video ve gizlenecek videonun her bir karesine resim steganografisi yöntemlerinden bazıları uygulanır. Gizlenecek videonun her bir karesine düzensiz dikdörtgen ayrıştırması uygulanır. Ayrıştırılan kodlar şifrelenebilir. Bu kodlar örtü videosunun her bir karesinin en önemsiz dört bitine saklanarak metot uygulanmış olur (Dun Hu ve Tak U. 2011).

3.1.3.3 Maskeleme ve filtreleme yöntemleri (Masking and filtering methods)

Maskeleme ve filtreleme yöntemlerinde 24-bit renk kod yapısına sahip resimler kullanılır. Bu yöntem renkli veya gri tonlama yapısındaki resimlerde uygulanmakla beraber resmin niteliğini etkilememektedir. Diğer steganografi yöntemlerine göre veri maskeleme işleminde gizli mesaj neredeyse bir multimedya dosyası gibi işlenmektedir. Gizli bilginin maskeleme işlemi yöntemi günümüzdeki steganaliz yöntemleriyle kolay bir şekilde keşfedilememektedir.

3.1.3.4 Sıkıştırma video steganografisi

Bu yöntem tamamen sıkıştırılan alanda yapılır. Veriler maksimum kare değişimi ve P, B alanlarındaki maksimum hareket vektörleri ile birlikte I çerçevesi alanına gömülebilir. AVC kodlama tekniği ile oluşturulan videolarda en yüksek performansa ulaşılmaktadır. DCT (Shanableh ve Tamer 2012), TPVD (Sherly ve Amritha 2010), Macro Block (Kapotas ve ark. 2007) gibi steganografik yaklaşımlar bu alanda önerilmiştir.

3.1.4. Resim steganografisi

Dijital resimlere normal olarak bakan kişilerin anlayamayacağı şekilde gizli verinin saklanması işlemidir. Örtü verisi olarak JPEG, BMP, GIF gibi resim dosyaları kullanılır. Resim içerisine gizli bir mesaj ya da veri saklanırken o verinin şifrelenmesi ya da verinin dikkati daha az celbedecek alanlara saklanması verinin güvenliği açısından önem teşkil etmektedir. Resim steganografisinde kullanılan çok çeşitli yöntemler kullanılmakla birlikte bunlar dört ana başlık altında incelenecektir (Hussain ve Hussain 2013).

- Uzay alanı metotları (Spatial Domain Methods)
- Alan dönüştürme teknikleri (Transform Domain Techniques)
- Çarpıtma teknikleri (Distortion Techniques)
- Maskeleme ve filtreleme yöntemleri (Masking and Filtering)

3.1.4.1 Uzay alanı metotları (Spatial Domain Methods)

Uzay alanı metotlarının çok farklı versiyonları bulunmakla birlikte, veri saklama işlemi yapılırken bütün metotlar resim piksel değerlerinden bazılarını değiştirir. LSB steganografi temelinde kullanılan algoritmalar bu alandaki en basit yöntemler olup gizli mesaj herhangi bir bozukluk olmaksızın piksel değerlerinin en önemsiz bitlerine gömülür. En önemsiz bitlerde yapılan değişiklikler insan gözü tarafından algılanması mümkün değildir. Uzay alanı metotları;

- En önemsiz bit (LSB)
- Piksel değeri farkı (Pixel value differencing (PVD))
- Kenar tabanlı veri gömme metotu (Edges based data embedding method (EBE))
- Rastgele piksel veri gömme metodu (Random pixel embedding method (RPE))
- Eşleme piksel veri saklama metodu (Mapping pixel to hidden data method)
- İsimlendirme ya da bağlantı metodu (Labeling or connectivity method)
- Piksel yoğunluğu metodu (Pixel intensity based method)
- Doku tabanlı metot (Texture based method)
- Histogram değiştirme metotları (Histogram shifting methods)

Uzay tabanlı LSB yöntemlerinin genel avantajları;

- Orijinal resimde bozulmanın az olması
- Resim içerisine yüksek oranda veri saklanabilmesi

LSB yöntemlerinin dezavantajları;

- Daha az sağlam olmakla birlikte, gizli veriler örtü resmi üzerindeki tahriflerle kaybolabilir.
- Gizli bilgi basit bir saldırı ile kaybolabilir.

3.1.4.2 Alan dönüştürme teknikleri (Transform Domain Techniques)

Resim içerisinde veri saklama işleminde daha karmaşık yöntemler içeren bir tekniktir. Bu alanda çeşitli algoritma ve yöntemler tanımlanmıştır. Alan dönüştürme gömme tekniği, gömme teknikleri alanında önerilen algoritma serilerinin temel alanı olarak isimlendirilebilir (Katzenbeisser ve Petitcolas 2000). Bir sinyalin frekans alanındaki veri gizleme işlemi, zaman alanında veri saklama işlemindeki prensiplerinden daha güçlüdür. Günümüzdeki güçlü steganografik sistemlerden pek çoğu alan dönüştürme yöntemi ile geliştirilmiştir. Alan dönüştürme yönteminin uzay alanı tekniklerine tercih edilmesinin sebebi, alan dönüştürme yönteminde verinin saklanacağı alanın resim kesme, sıkıştırma gibi operasyonlara daha az maruz

kalmasıdır. Bazı alan dönüştürme teknikleri resim formatından bağımsız olarak çalışmaktadır. Alan dönüştürme metotları;

- Ayrık Fourier dönüşümü (Discrete Fourier transformation technique (DFT)).
- Ayrık kosinüs dönüşümü (Discrete cosine transformation technique (DCT)).
- Ayrık dalgacık dönüşümü (Discrete Wavelet transformation technique (DWT)).
- Kayıpsız ya da döndürülebilir (Lossless or reversible method (DCT))
- Katsayı bitlerine gömme (Embedding in coefficient bits) teknikleridir.

3.1.4.3 Çarpıtma teknikleri (Distortion Techniques)

Çarpıklık tekniğinde; gizli mesajın ortaya çıkarılma işlevlerinin çalışması, stego nesnesi ile karşılaştırma yapılabilmesi için orijinal örtü resmine de ihtiyaç vardır. Şifreleyen taraf değişiklerin listesini örtü resmine ekler. Bu şekilde gizli bilgi bozulma, çarpıklık olarak tanımlanır (Reddy ve Raja 2009). Stego nesnesi bu değişiklik dizisinin örtü verisine uygulanması ile elde edilir.

Bu değişiklik listesi gizli mesajı doğrulamak için kullanılır (Katzenbeisser ve Petitcolas 2000). Gizli mesaj sözde rastgele seçilmiş piksellere şifrelenir. Eğer stego resmi verilen piksel değeri ile örtü verisinin aynı piksel değeri farklıysa “1” aynı değilse “0” olarak kodlanır. Şifreleyen taraf resmin istatistik verilerini etkilemeyecek şekilde “1” değerli pikselleri değiştirebilir. Bu yöntemde örtü verisinin de karşı tarafa iletilme ihtiyacı yararı oldukça kısıtlamaktadır. Diğer steganografi yöntemlerinde örtü nesnesi birden fazla kullanılmamaktadır. Bir saldırganın stego resim üzerinde yaptığı kesme, ölçekleme ve çevirme gibi işlemler alıcı tarafından kolayca fark edilebilir. Bazı durumlarda, gizli mesaj hata düzeltmesi ile birlikte şifrelendiğinde, değişiklik tersine çevrilip gizli mesaj elde edilebilir (Kruus ve ark. 2003).

3.1.4.4 Maskeleme ve filtreleme yöntemleri (Masking and Filtering)

Bu yöntemlerde veri gizleme işlemi dosya damgalama ile aynı yöntem olan resmin işaretlenmesi ile sağlanır. Gizli bilgi karışık alanlar yerine daha önemli bölgelerde saklanır. Gizli veri örtü nesnesi ile daha bütünleyici ve birleşik yapıdadır. Damgalama yöntemleri resmin yapısını bozma korkusu olmaksızın uygulanabilir. Bu

yöntemler LSB yöntemlerinde daha sağlam olmakla birlikte, bu yöntemler 24 bitle sınırlı olup sadece gri ton paletindeki resimlere uygulanabilir.

3.2. LSB Yöntemi

İngilizce açılımı (Least Significant Bit) en değersiz bit olan yöntem, bir baytın 8 bitlik dizisindeki en sağdaki bit veya bitleri ifade eder. Örneğin, bir baytlık 10111001 ikili verisi düşünüldüğünde en önemsiz biti, en sağdaki 1'dir.

Geliştirdiğimiz mobil uygulamamızda LSB yöntemini resim steganografisinde kullandığımız için, bu metodu dijital resimler üzerinden anlatacağız. Diğer örtü verileri olan dijital video ve ses verilerine LSB uygulama işlemini, okuyucunun kıyas yeteneğine ve bu konuda yazılmış diğer makale ve kitaplara bırakıyoruz.

Dijital dünyadaki resimler genellikle 8-bit grayscale veya 24-bit RGB renk şemaları olarak tanımlanan piksellerden oluşur. 8-bit renk şemasında her piksel 1 bayt iken 24-bit renk şemasında her piksel 3 bayttır. Bu baytlar sırasıyla İngilizce terimleri red, green, blue olan (RGB) kırmızı, yeşil, mavi renklerini ifade eder. Bu durumun sonucunda bitler ile renkler arasında aşağıdaki gibi bir ilişki oluşur;

8-bit $= 2^8 = 256$ farklı renk

24-bit $= 2^{24} = 16777216$ farklı renk üretilebilir.

Aşağıdaki tabloda örnek renginin ikilik tabanda, sayısal, onaltılık tabandaki ifade şekilleri gösterilmiştir. Örtü verisi içerisine bilgi saklama işlemi bit düzeyinde yapılan bir muamele olmasından dolayı LSB metodu rengin ikilik tabandaki gösterimi ile ilgilenmektedir.

Çizelge 3.2.1. Bir pikselin renk şeması olarak gösterimi

	Kırmızı (Red)	Yeşil (Green)	Mavi (Blue)
İkilik Tabanda	11111000	11001001	00000011
Onluk Tabanda	248	201	3
Onaltılık Tabanda	F8	C6	03

Aşağıdaki örnekte bir resmin ilk üç pikselinin son bitleri kullanılarak 'C' karakterinin nasıl gizleneceğini göstereceğiz.

C karakterinin ikilik tabandaki gösterimi = **01000011**

Çizelge 3.2.2. Bir resmin ilk üç pikselinin ikilik tabanda gösterimi

	1.Piksel	2.Piksel	3.Piksel
R	00100111	11101001	11001000
G	00100111	11001000	11101001
B	11001000	00100111	11101001

Çizelge 3.2.3. C karakteri gizlendikten sonra resmin ilk üç pikseli

	1.Piksel	2.Piksel	3.Piksel
R	00100110	11101001	11001000
G	00100110	11001000	11101000
B	11001001	00100111	11101001

Çizelge 3.2.3'te 'C' karakteri gizlendikten sonra değişen bitler koyu renk ile gösterilmiştir. Bu durumda 9 bitten sadece 4 bit değişip diğer bitler aynı değerde kalmıştır.

Şekil 3.1.4. Örtü resmi

Şekil 3.1.5. Stego-object

Şekil 3.1.4 de gizli bilgi içermeyen orijinal örtü verisi gösterilmektedir. Şekil 3.1.5 de ise içerisine Necip Fazıl Kısakürek'in "Gelir" şiiri gizlenmiştir. Herhangi bir kişinin iki resim arasındaki farkı bakarak anlaması mümkün görülmemektedir.

LSB yöntemi temel alınarak üretilmiş bazı algoritmalar bulunmaktadır. Bunlar Jsteg, F3, F4, F5 algoritmalarıdır.

3.2.1 Jsteg

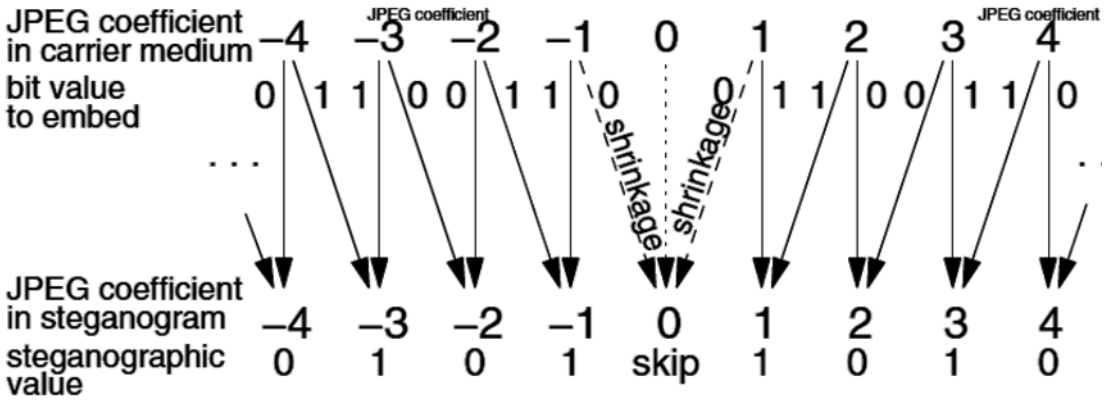
Bu algoritma Derek Upham tarafından stego-resimlere yapılan görsel saldırılara karşı geliştirilmiştir. Fakat sunduğu mesaj saklama kapasitesi bakımından tartışılmaktadır (stego nesnesinin 12.8%). Jsteg algoritması niceleme (quantisation) işleminden sonra, en düşük bitlerin sıklık katsayısına göre gizli mesaj bitlerini değiştirir. Jsteg niceleme işleminde 0 ve 1 katsayılarında herhangi bir düzenleme yapmadan bir sonraki değere geçer (Upham 1997). Bunun temel sebepleri, LSB işleminden 1 değerlerin 0'a dönüşümü ve gizli bilginin kaybolma durumlarının önüne geçmektir. Jsteg algoritması niceleme işleminde DCT katsayılarında ilerlerken herhangi bir steganografi anahtarı kullanmaz.

3.2.2 F3

F3 algoritmasının Jsteg algoritması ile iki önemli farkı bulunmaktadır.

1- Niceleme katsayı biti ile gizli mesaj bitlerinin eşleşmemesi durumunda Jsteg algoritmasında yapılan yerine koyma işlemi yerine, F3 algoritmasında niceleme katsayı değerini düşürme işlemi yapılmaktadır. 0'dan daha küçük değer olamayacağı için 0 katsayısı kullanılmamaktadır.

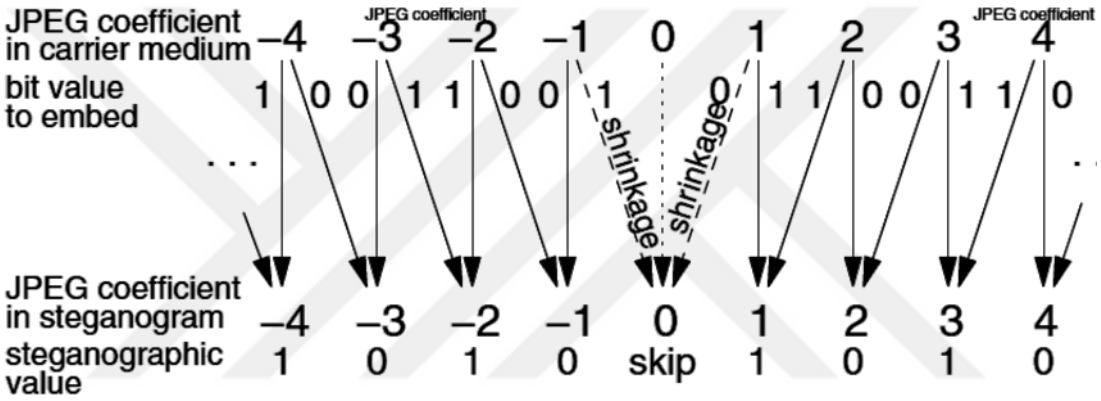
2- Gizleme işleminin sıkıştırma aşamasında DCT katsayı değeri 1 veya -1 olanlara 0 biti eklenmesidir. Bu durumun ortaya çıkmasındaki temel sebep ise alıcıya 0 bitinin geçilen 0 biti mi olduğu yoksa sıkıştırmadan dolayı mı üretilen 0 bitinin olduğunun söylenememesidir.



Şekil 3.2.1 F3 algoritması çalışma yöntemi

3.2.3 F4

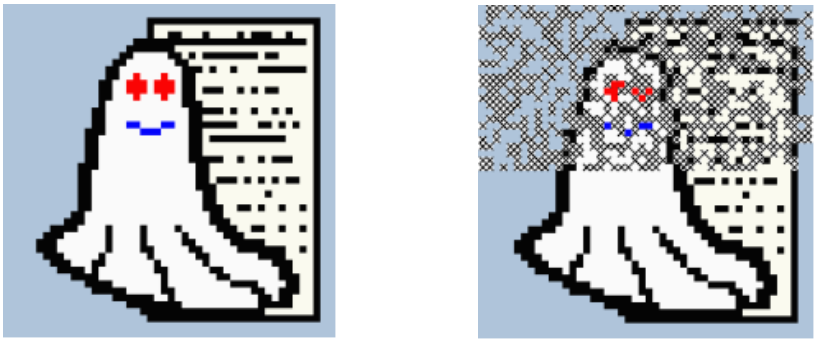
F3 algoritmasına çok benzer olmakla birlikte mesaja bit ekleme yönteminde AC katsayısını dikkate almasıdır. F3 algoritmasının zayıf noktalarından bir tanesi; sıkıştırma işleminde 0 bitlerinin göz ardı edilmesi zorunluluğundan dolayı mesajda 1 bitlerinden daha çok 0 bitlerinin olmasıydı. Bu da histogramlardan anlaşılabilecek bir durumdu. F4 algoritması negatif katsayı değerlerini steganografik değerlere dönüştürerek bu zayıflığı ortadan kaldırmayı amaçlamıştır. F4 algoritmasının kodlaması; negatif çift katsayı değerleri için steganografik 1, negatif tek değerler 0; çift pozitif değerler 0 ve tek pozitif değerler ise 1 şeklindedir.



Şekil 3.2.2 F4 algoritması çalışma yöntemi

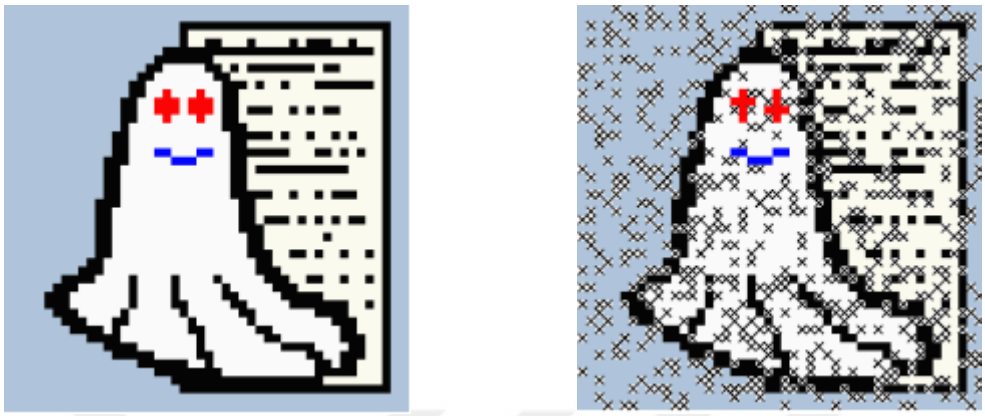
3.2.4 F5

F4 algoritmasına çok benzer olmakla birlikte F4 algoritmasının istatistiksel saldırılara karşı 2 yeni özellik eklenmiştir. Bunlar permütasyon saçılması ve matris kodlamasıdır. Birçok saklama yönteminde gizli mesajlar örtü verisinin tamamını kapsayacak kadar büyük veriler değildir. Bu da gizli bilginin örtü verisi içerisinde saklanırken Şekil 3.2.3 de bir yerde toplanmasına (continuous embedding) yol açar. Bu durum istatistik ve histogram saldırılarına karşı zafiyet oluşturur.



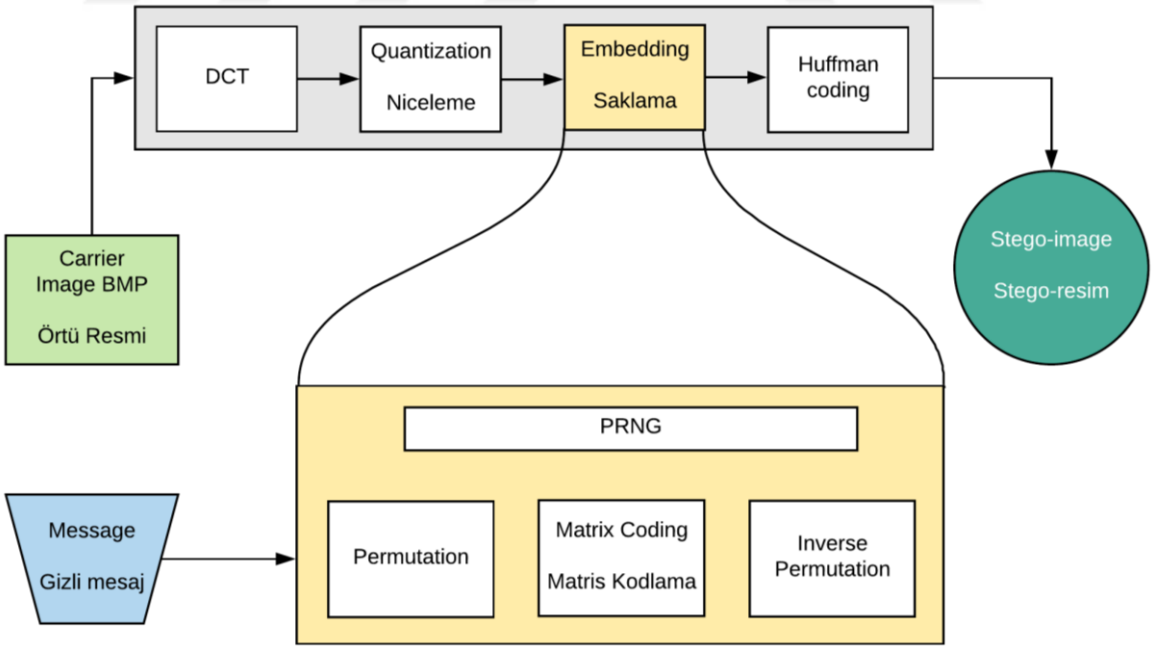
Şekil 3.2.3. Saklanan verinin tek yerde toplanması, (x) değişiklikleri göstermektedir.

Bu sebeple permütasyon saçılması işlemiyle Şekil 3.2.4 de görüldüğü üzere gizli mesajın örtü verisi içerisine homojen olarak saklanması amaçlanır.



Şekil 3.2.4. Permütasyon saçılması işleminden sonra, (x) değişiklikleri göstermektedir.

Matris kodlaması işleminde değişiklik yapılacak bit sayısının azaltılarak F5 algoritma performansı arttırılmıştır (Westfeld 2001). F5 algoritmasının çalışma yordamı Şekil 3.2.5 de gösterilmiştir.



Şekil 3.2.5. F5 algoritması işlem akışı

3.2.5 Ayrık Logaritma

Ayrık logaritma algoritmasının temel amacı rastgele tekrarsız sayılar üretmektir. Uzak bir geçmişe sahip olan ayrık logaritma, başlangıçta sınırlı hesaplama alanlarında kullanılmıştır. Algoritma alanındaki sorularla yirminci yüzyılda tekrar gelişmeye başlamıştır.

Ayrık logaritma işleminde öncelikle bir asal sayı olan p değeri belirlenir. Daha sonra 1'den (p-1)'e kadar sayılar taranarak bir ilkel kökü olan a değer seçilir. Bu durumda eğer a değeri p asal sayısının ilkel bir kökü ise,

a mod p, a^2 mod p, …, a^{p-1} mod p değerleri birbirinden faklı ve 1 ile p-1 arasında sayılardır.

a, a^2, …, a^{p-1} sayıları ise p ile aralarında asal ve farklı sayılardır. Bu durumda herhangi bir y sayısı ve ilkel kökü a olan p asal sayısı için,

$$y = a^i \bmod p \tag{1}$$

$0 \leq i \leq (p\text{-}1)$ olmak kaydıyla (1) numaralı denklemden eşsiz bir i değeri bulunabilir (Mohamed Amin ve ark. 2003). Bu i değeri y'nin a tabanında mod p ye göre ayrık logaritmasıdır.

Şaban Gülçü ve Sevda Gülcü'nün önerdiği ayrık logaritma ile LSB kodlama yönteminde (2013), ayrık logaritma örtü resminin boyutundan küçük olan en büyük p asal değerini seçer. (1) numaralı denkleme göre tekrarsız rastgele sayılar üretilir. Üretilen bu sayılar LSB kodlama yapılacak pikselleri belirtmektedir. Bu yöntemde 8-bitlik örtü resmi kullanılmış ve belirlenen piksellerin LSB yöntemi ile üç biti değiştirilmiştir. Örtü resminin son iki pikseli gizli mesaj saklamak amacıyla kullanılmaz. Bu iki piksel alanına gizli mesaj saklama işleminde kaç tane piksel kullanıldığı bilgisi saklanır.

3.3. Android Studio

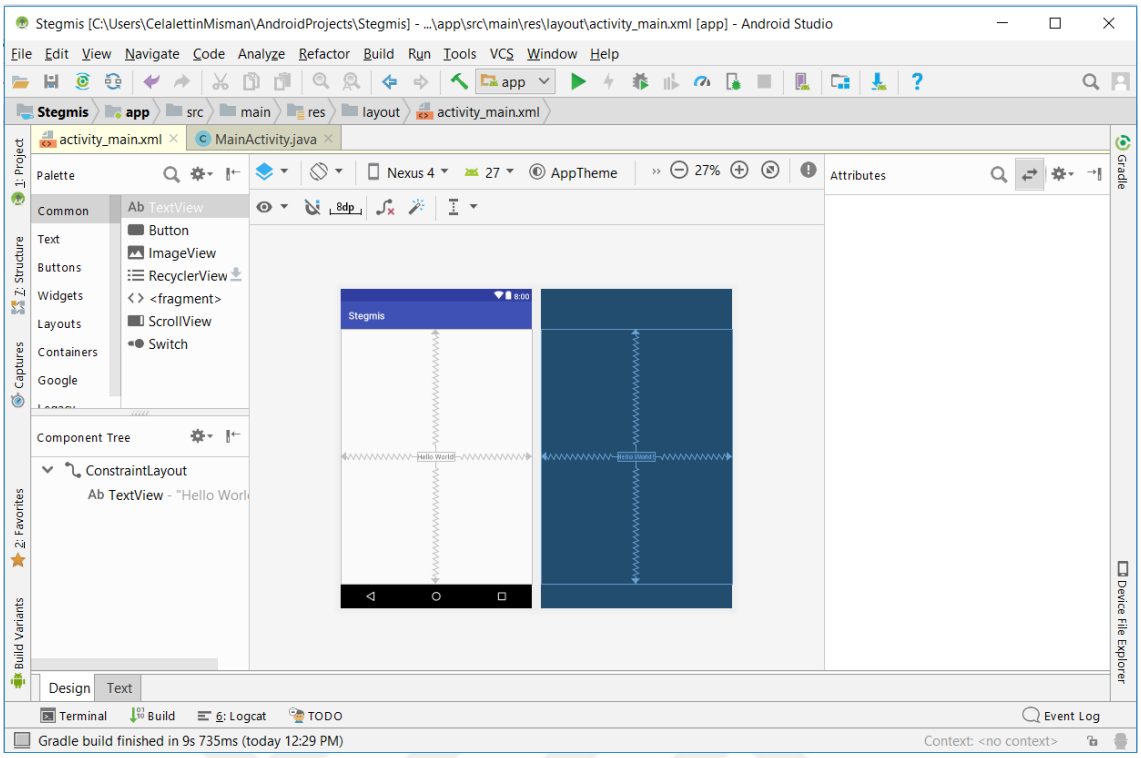
Android Studio; geliştiricilere her android cihaz için uygulama geliştirebilmesine olanak sağlayan, gelişmiş ve modern bir araçtır. Android Studio; proje ve dosya yönetimindeki kolaylıkların yanı sıra, zekice tasarlanmış kod düzenleme sistemi, hata ayıklama araçları, uygulamanın performans durumunu analiz etme özellikleri ile eşsiz ve özgün bir geliştirme ortamıdır.

Yeni geliştirilecek uygulamaların hangi API seviyesindeki cihazlar tarafından kullanıp kullanılmayacağına karar verilmelidir. Android Studio güncel API seviyelerini ve oranlarına ait bilgiyi vermektedir. Günümüzdeki güncel kullanım oranları API seviye ve isimleri ile birlikte aşağıdaki çizelgede yer almaktadır (Haziran 2018, https://developer.android.com/about/dashboards/ adresinden alınmıştır.).

Çizelge 3.3.1. Güncel Android versiyonları ve kullanım oranları

Versiyon	İsmi	API Seviyesi	Kullanım Oranı
2.3.3 – 2.3.7	Gingerbread	10	0.3 %
4.0.3 - 4.0.4	Ice Cream Sandwich	15	0.4 %
4.1.x	Jelly Bean	16	1.5 %
4.2.x		17	2.2 %
4.3		18	0.6 %
4.4	Kitkat	19	10.3 %
5.0	Lolipop	21	4.8 %
5.1		22	17.6 %
6.0	Marshmallow	23	25.5 %
7.0	Nougat	24	22.9 %
7.1		25	8.2 %
8.0	Oreo	26	4.9 %
8.1		27	0.8 %

Android uygulamaların bir diğer temel elemanı da kullanıcı arayüzünün tasarlanıp, etkileşimlerinin tanımlandığı “Layout” olarak isimlendirilen plan dosyalarıdır. Bunlar xml uzantılı dosyalardır.



Şekil 3.3.1. Layout dosyası çalışma ekranı

Bu plan (layout) dosyalarının çalışma ekranı şekilde 3.3.1'deki gibidir. Resmin sol alt köşesindeki "Text" sekmesinde bu dosyaları xml olarak görüntüleyip, düzenlemek mümkündür.

Günümüzde uygulamaların geliştirilmesi ve takip edilmesinde bir versiyon kontrol sistemi kullanmak büyük öneme sahiptir. Android Studio Git, Github, CVS, Google Cloud, Mercurial, Subversion versiyon/sürüm kontrol sistemleri ile uyumlanıp bütünleşebilir. Bu tarz versiyon kontrol sistemleri yazılımın kolayca versiyon işlemine tabi tutulmasını, gerektiğinde eski versiyonlara dönülebilmesini sağlamaktadır. Projeler uzak sunucu tarafında versiyon sistemi kullanılarak saklandığında veya takip edildiğinde, birden fazla geliştiriciye aynı proje üzerinde çalışabilme imkânı sağlanır. Yapılan değişikliklerin takip edilmesi, iptal edilmesi, karşılaştırılması gibi birçok olanağa ve faydaya sahip olan versiyon kontrol sistemleri proje yönetiminde ve yazılım geliştirmede olmazsa olmaz eşsiz bir araçtır.

3.4. MySQL

İlişkisel veritabanı yönetim sistemi olan MySQL, her işletim sisteminde rahatlıkla çalışmaktadır. Geniş bir kullanıcı kitlesine sahip olan MySQL Linux tabanlı işletim sistemlerinde daha verimli çalışmaktadır. Mantıksal modellemesini veri tabanları, tablolar, görüntüler, satırlar, sütunlar gibi elemanlarla sağlayan MySQL, esnek programlamaya olanak sağlamaktadır.

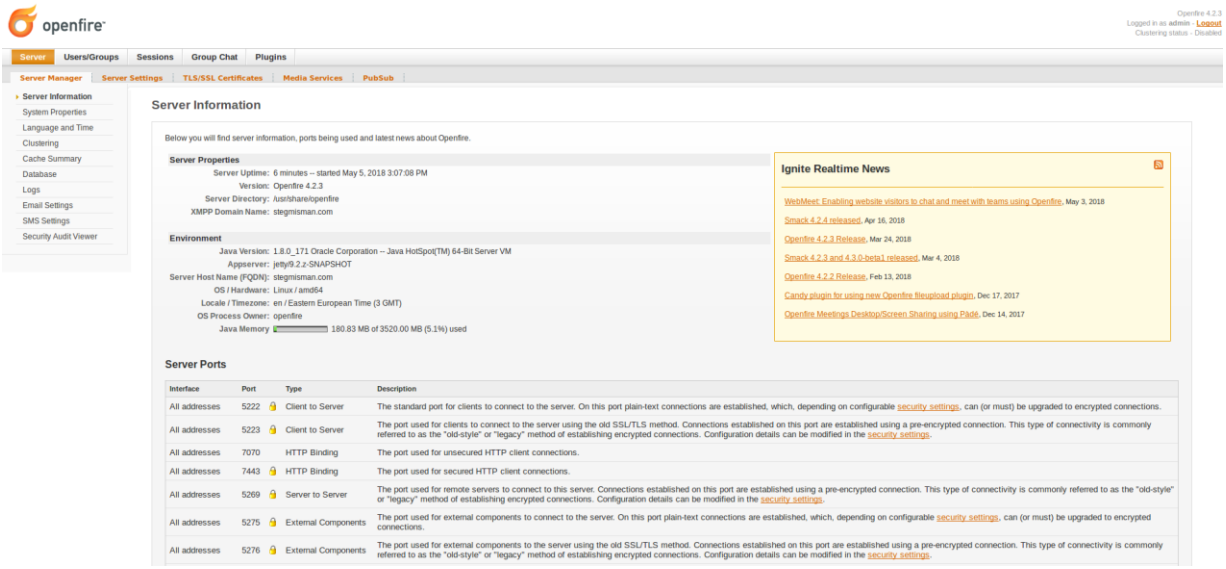
Açık kaynaklı kod yapısına sahip olan MySQL güvenilir, hızlı ve kolay bir kullanım sunmaktadır. MySQL veritabanı yazılımı sunucu/istemci mimarisinde tasarlanmış, iş parçacıklarından oluşmaktadır. Bununla birlikte farklı istemci programlarını, kütüphanelerini, yönetim araçlarını ve uygulama ara yüzlerini desteklemektedir.

C++, J 5.1, ODBC, Python, Net, PHP gibi bağlantı araçları mevcut olup farklı programlama dilleri ile uygulama geliştirme işlemine olanak sağlamaktadır.

3.5. Openfire Server

Openfire Server XMPP protokolü ile anlık mesajlaşma imkânı sağlayan, Java programlama dili ile geliştirilmiş Apache lisansı dahilinde açık kaynaklı bir sunucudur. Eklenti mimarisi içeren sunucu grup sohbetini destekler nitelikte geliştirilmiş ve eklentiler sayesinde diğer gerekli ihtiyaçların kolayca tedarik edilmesi ve uygulanmasına olanak sağlayan bir yapıdadır.

1. Web tabanlı yönetim paneli
2. SSL/TLS desteği
3. Kullanıcı dostu
4. Platform bağımsız olması da genel özellikleri arasındadır.



Şekil 3.5.1. Openfire Server yönetim paneli arayüzü

Yönetici paneline giriş işleminden sonra bizi yukarıdaki ekran karşılayacaktır. Burada Openfire Server ile ilgili versiyonu, alan adı, ne süredir açık olduğu gibi bilgilerin yanı sıra; Java versiyonu, Java hafıza durumu, sunucu zaman dilimi gibi yararlı bilgiler de gösterilmektedir. “Server Port” isimli alanda ise sunucunu kullandığı portlar ve açıklamaları mevcuttur. En üstteki panel ana ayar gruplarını içermektedir. Bir altındaki panel ise üst gruba ait diğer alt grup ayarlarını içermekte, sol panel ise bu gruba ait ayarları içermektedir. Genel olarak; “Server” ayar tabında sunucu genel ayarları ve kurulum ayarlarını görüntüleme ve düzenleme işlemleri yapılmaktadır. “User/Groups” ayar tabında ise kayıtlı istemcileri ve grupları görüntüleme ve düzenleme işlemleri yer almaktadır. Kullanıcıların ne zamandır aktif oldukları ve bağlantı durumları ile ilgili bilgiler “Sessions” ayar tabında görüntülenebilir. “Group Chat” kısmında oluşturulan grupların görüntüleme, düzenleme ve yönetim işlemleri bulunur. Daha önce bahsettiğimiz uygulama eklenti yüklenerek çalışma performans ve özelliklerini geliştirme gibi işlemler “Plugins” ayar tabında düzenlenir ve görüntülenir.

3.6. Smack

Smack açık kaynaklı XMPP istemci kütüphanesidir. Bir Java kütüphanesi olan Smack, eklendiği uygulamalara XMPP sunucular ile kolaylıkla haberleşmesini sağlayacak fonksiyon, kütüphane ve metotları içermektedir.

3.7. AES

Modern şifreleme yöntemleri genel olarak ikiye ayrılmaktadır. Bunlar simetrik ve asimetrik şifreleme yöntemleridir. Asimetrik şifrelemede gizlenmek istenen bilgi herkesin bildiği bir anahtar ile şifrelenir ve sadece gizli anahtar kullanılarak çözülebilir. Simetrik algoritmalarda ise tek bir gizli anahtar bulunur; şifreleme ve şifre çözme işlemleri bu anahtarlar vasıtasıyla yapılır.

Simetrik şifreleme yöntemlerinden biri olan AES, Ocak 1997'de NIST'in mevcut standart olan DES'in yerini alması düşünülerek üretme süreci başlatılmıştır. Böyle bir sürece girmeye neden olan şey ise DES'in özel bir amaç için tasarlanmış olmasında dolayı 64 bitlik anahtar uzayının gelişen teknoloji ve artan işlemci hızları karşısında yetersiz kalmasıdır. 4 yıl süren algoritma yarışmasının ve değerlendirme sürecinin sonunda Ekim 2000'de NIST, Joan Daemen ve Vincent Rijmen tarafından tasarlanan, Rijndael algoritmasının Gelişmiş Şifreleme Standardı (AES) olarak kullanılacağını açıklanmıştır. AES algoritması, sabit 16 baytlık veri bloklarını 128, 192 veya 256 bit anahtar seçenekleri ile şifreleyen bir blok şifre algoritmasıdır.

3.8. SHA-2

Özet algoritması girilen veriyi sabit bir uzunluğa indirgeyen matematik fonksiyonudur. Özet algoritmalarının tek yönlü olması hasebiyle sonuç verisinden giriş verisine ulaşmak mümkün değildir. Fakat aynı özet değeri farklı verilerle elde edilebilir. Özet algoritmaları dijital imzalarda, mesaj ve kimlik doğrulama işlemlerinde oldukça yoğun bir şekilde kullanılmaktadır. Veri bütünlüğündeki eksikliği, hatayı, yanlışı anlamada basit, ucuz fakat güçlü bir yöntemdir. MD5, MD6, SHA-0, SHA-1, SHA-2 algoritmaları günümüzde en çok bilinen ve kullanılan özet algoritmalardır. İşlemci hızları ve teknolojilerde gelişmelerin artması ile birlikte, zamanlar bu algoritmalarda güvenlik açıkları oluşmakta ve özet değerleri çözülebilmekte ve üretilebilmektedir. Bu algoritmalara günümüzde kullanımdan kaldırılmış SHA-0, SHA-1, MD4, MD5 özet algoritmaları örnek olarak verilebilir.

SHA-0 özet algoritması 1993'te Amerika tasarlanmıştır. Yetersizliği sebebiyle SHA-0 algoritmasını, 1995 yılında girilen verileri 160bit (20 bayt) özet değerine çeviren SHA-1 takip etti. Bu özet fonksiyonunda da güvenlik açıklarının bulunması üzerine, NSA ve NIST aileye 2002 yılında SHA-2 standardını getirdi. SHA-2 standardı birçok

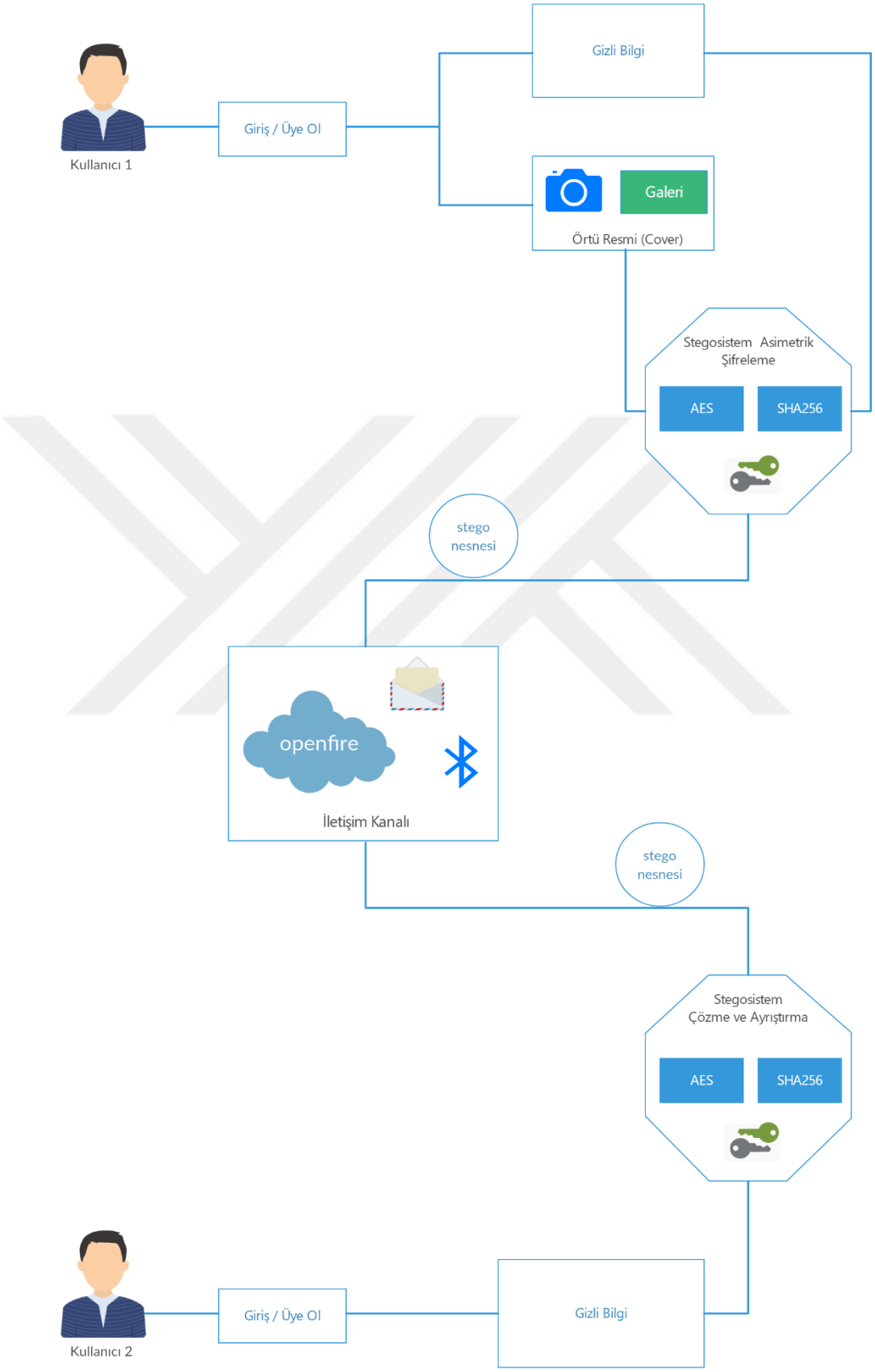
yeni özellikle beraber; 224, 256, 384, 512 bitlik özet değerleri üretebilecek şekilde tasarlanmıştır. Bu durum SHA-2 ailesini diğer özet algoritma standartlarına karşı daha cazip kılmaktadır. Günümüzde SHA-2 algoritması güvenli kabul edilmektedir.

4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

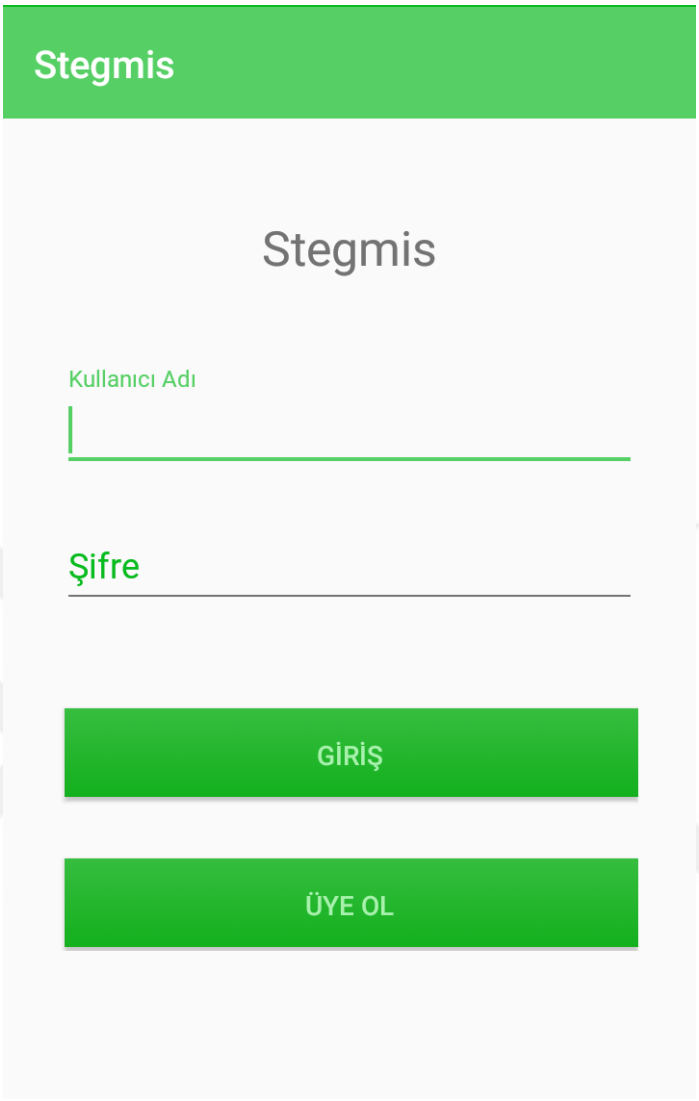
Bu çalışmamız kapsamında STEGMIS isimli, istemci-sunucu mimarisine sahip android mobil uygulama öneriyoruz. Uygulamamızın istemci tarafı Android Studio geliştirme ortamında Java programlama dili ile geliştirilmiştir. Uygulama geliştirme aşamasında Git versiyon kontrol sistemi kullanılarak sürümlendirme ve takip işlemleri sağlanmıştır.

Mobil uygulamalar açık kaynaklı bir XMPP sunucusu olan Openfire üzerinden haberleşmektedir. Openfire sunucusunun veri tabanı yönetim sistemi tarafı MySQL ile yapılandırılmıştır. Gizli bilginin resim içerisine saklanması aşamasında tek taraflı şifreleme algoritması olan AES ve 256-bitlik gizli anahtar kullanılarak şifrelenmiştir. Gizli anahtar her kullanıcıya ait eşsiz bir bilginin SHA-256 özet fonksiyonu ve bir de vektörel anahtar kullanılarak elde edilir. Gizli anahtarın her kullanıcıya ait, özel bir bilgi ile oluşturulmasıyla, resimdeki gizli mesajın asimetrik şifrelenmesi sağlanmıştır. Bu sayede resme ulaşan üçüncü kişiler yalnızca uygulama kullanarak gizli mesaja ulaşamayacaklardır. Gizli mesaja ulaşabilmek için bir de doğru kişi ile kimlik doğrulaması yapılması gerekmektedir. Şifrelenen gizli bilgi resim piksellerinin bayt değerlerinin son 3 bitine LSB yöntemi ile saklanarak stego-resim oluşturulmuştur. Uygulamamız oluşturulan stego-resmin kişisel veri saklama amacıyla kullanılmasına da olanak sağlamaktadır. Bu sayede kullanıcılar resimleri farklı kişilere mesaj iletimi için kullanmanın yanında, kişisel özel bilgilerini de resimler içerisinde saklayabilirler.

Verilen bilgiler ve kullanılan teknolojiler doğrultusunda geliştirilen STEGMIS mobil uygulamamızın mimarisi ve ekranları aşağıda verildiği şekildedir.



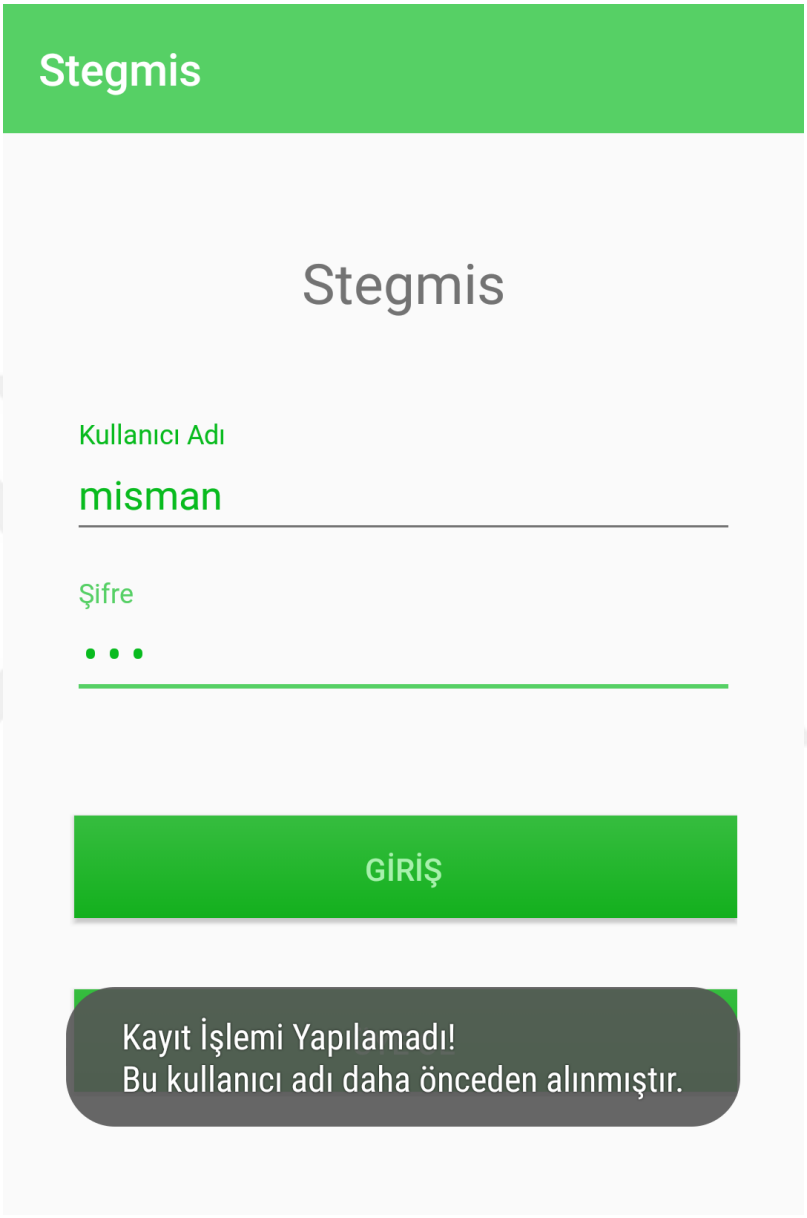
Şekil 4.1. Uygulama akış şeması



Şekil 4.2. Uygulama giriş ekranı

Uygulamaya ilk giren kullanıcı bu ekran ile karşılaşmaktadır. Buradan daha önce seçilmemiş bir kullanıcı adı girip, şifresini belirledikten sonra ÜYE OL diyerek bir hesap oluşturur.

Daha önce seçilmiş bir kullanıcı adı ile sisteme kayıt olmaya çalışıldığında aşağıda görüldüğü üzere ‘Kayıt İşlemi Yapılmadı. Bu kullanıcı adı daha önceden alınmıştır.’ Uyarısı ile karşılaşılacaktır.



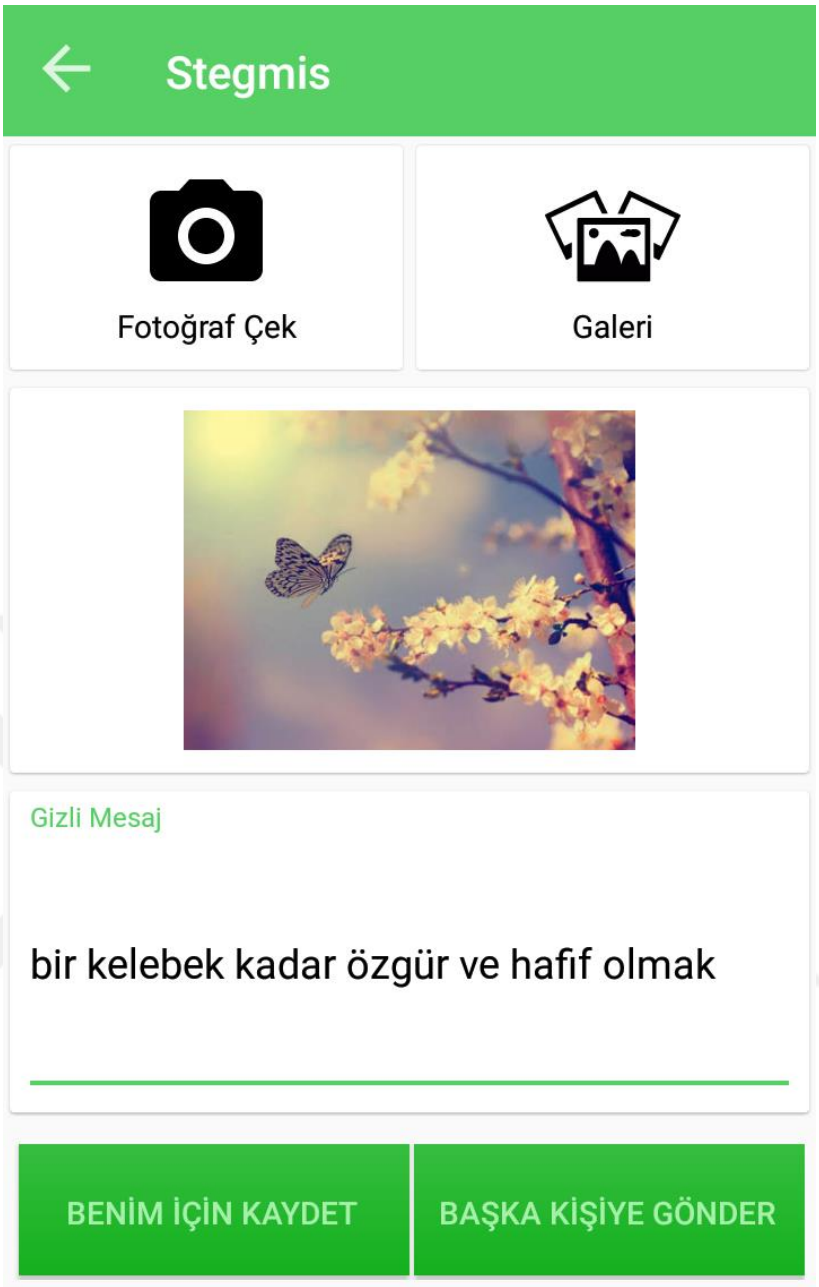
Şekil 4.3. Aynı kullanıcı adı hatası

Daha önce kullanılmamış, özgün bir kullanıcı adı seçilerek hesap oluşturma işlemi başarılı olduktan sonra; yeni kullanıcı GİRİŞ yaparak uygulamayı kullanmaya başlar.

Kullanıcı giriş işlemi tamamlandıktan sonra aşağıdaki ekranla karşılaşmaktadır. Bu ekranda kullanıcı yapmak istediği işlemi seçer. Bu işlemler;

1- Resme mesaj gizleme: Bu bölüm stego resmini oluşturulması üretilmesi, göderilmesi işlemlerini içerir.
2- Gizlenen mesajı gösterme: Bu bölümde yeni gelen stego-resim mesajının veya daha önce kaydedilmiş olan stego-resminin çözülüp gizli mesajın gösterilmesi işlemi yapılır.
3- Uygulamadan çıkış işlemi yapılır.

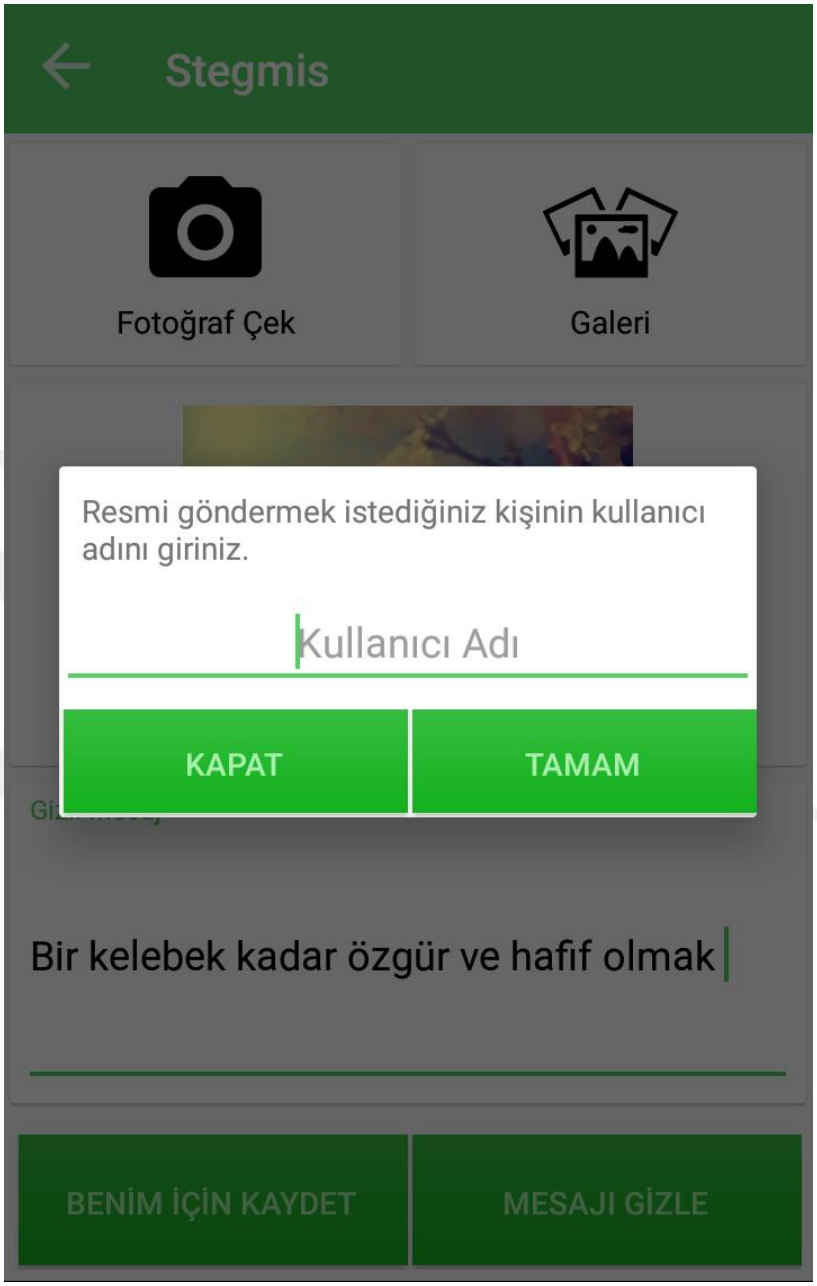
Şekil 4.4. İşlem seçme ekranı



Şekil 4.5. Resme mesaj gizleme

Resme mesaj gizleme işlemi seçildiğinde Şekil 4.5 de gösterilen ekran karşımıza çıkar. Bu ekranda sıradan bağımsız olarak örtü resmi seçme işlemi ve örtü resminin içerisine saklanacak yazı girişi yapılır. Örtü resmi seçme işlemi şekilde görüldüğü üzere anlık fotoğraf çekme ve galeriden ekleme olarak iki farklı yöntemle yapılabilir. Gizli mesajın belirlenmesi ve girilmesinden sonra kullanıcı stego-resim nesnesini kişisel kullanım için "BENİM İÇİN KAYDET", başka bir kişiye iletim için "MESAJI GİZLE" demelidir. "BENİM İÇİN KAYDET" seçimi yapıldığında stego-resim içerisine kendi kullanıcı adı ile şifrelenerek gizli mesaj saklanır ve kaydedilir. Bu sayede başka bir kişi resmi ele geçirse bile gizli bilgiye ulaşamaz.

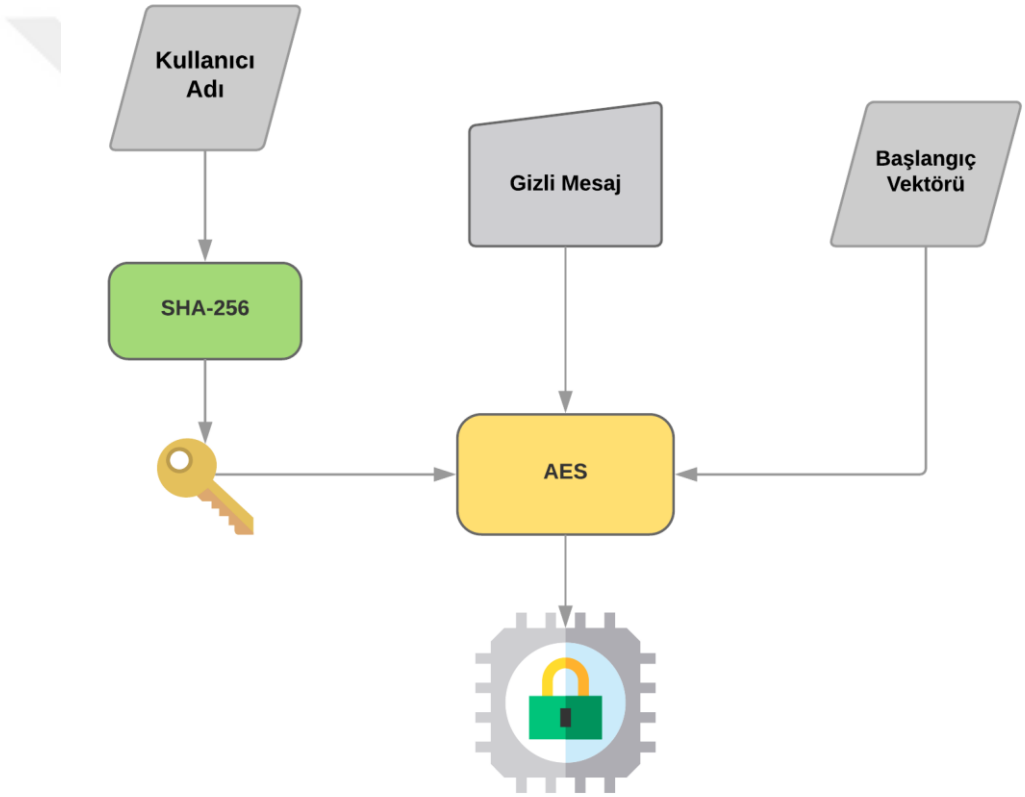
Gizli mesajın ve örtü resminin belirlenmesinden sonra kullanıcının stego-resmi diğer kullanıcıya gönderme işlemi Şekil 4.6 ile devam eder.



Şekil 4.6. Alıcı seçim işlemi

Bu ekranda kullanıcı gizli bilgi gizlenmiş olan resmi göndermek istediği kişinin kullanıcı adını girer. Kullanıcı adını bilmiyor ise resim gönderme işlemi yapılamayacaktır. Bununla birlikte kendi kullanıcı adını yazması da sistem tarafından engellenmiştir. Eğer kendi kişisel bilgilerini saklamak istiyorsa bir önceki ekrandan "BENİM İÇİN KAYDET" butonuna tıklamalıdır.

Bu aşamalardan sonra gizli bilginin şifrelenerek resim içerisine kodlama işlemine geçilir. İlk olarak gönderilecek kişinin kullanıcı adı ile SHA-2 özet algoritması kullanılarak gönderilecek kişinin 256 bitlik anahtarı elde edilir. Anahtarın elde edilmesinden sonra gizli bilgi AES simetrik şifreleme algoritması, özet anahtarı ve başlangıç vektörü ile şifrelenir. Elde edilen şifrelenmiş veri önce baytlarına sonra bitlik verilere dönüştürülür. Bitlik verilerin uzunluğu hesaplanarak 32 bit veri şeklinde resim içerisine saklanacak bitler kısmının başına eklenir. Dönüştürülen bitler resmin ilk pikselinden başlanarak her baytın üç biti değiştirilecek şekilde LSB yöntemi ile kodlanır.



Şekil 4.7. Gizli mesajın şifrelenmesi

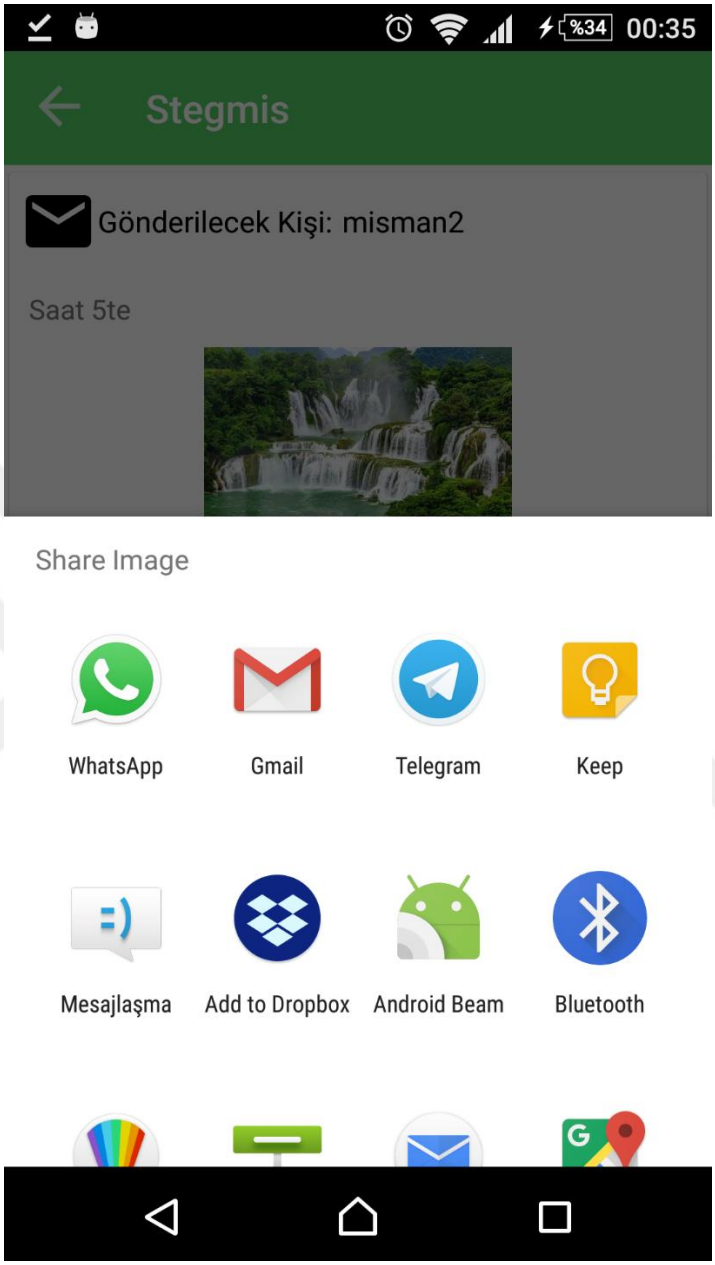
Şekil 4.7. Gizli mesajın bölümleri

Şekil 4.8. Stego-resim ön izleme

Kullanıcı mesaj göndereceği kişinin kullanıcı adını da girdikten sonra, bir sonraki ekran olan mesajın ön izlenmesine geçer. Bu ekran; gönderilecek kişinin ya da bir başka ifade ile mesajı ayrıştırıp görebilecek kişinin kullanıcı adının, mesaj saklanmış stego-resminin son halinin, gönderme ve paylaşma butonlarının bulunduğu Şekil 4.8. da gösterilen ekrandır. Kullanıcı "PAYLAŞ" diyerek farklı bir uygulama ile paylaşma işlemine geçebilir, ya da "GÖNDER" diyerek stego-resim iletimini tamamlar.

Mesaj gömülü resmin ve kime iletileceğinin ön izlemesi esnasında, kullanıcı bu stego-resmi Openfire sunucu üzerinden değil de farklı bir yöntemle paylaşmak

isteyebilir. Bu durumda “PAYLAŞ” butonuna tıklama ve iletimde kullanılacak uygulamayı seçmesi yeterli olacaktır.

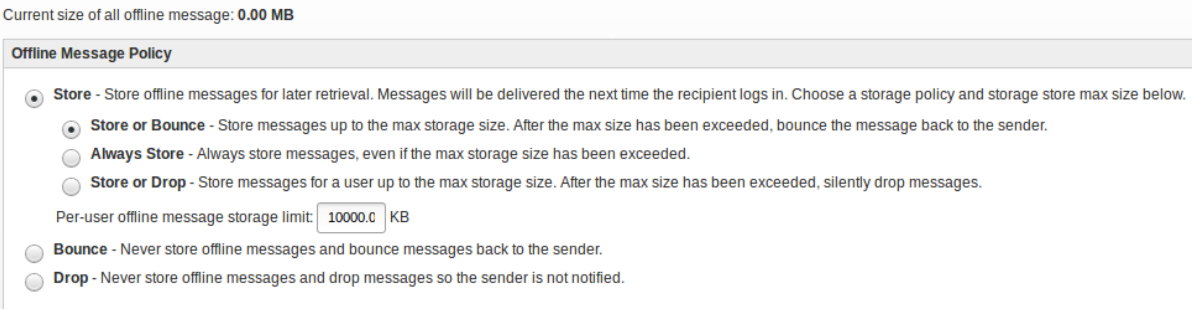


Şekil 4.9. Stego-resmi paylaşma işlemi

Kullanıcının stego-resim paylaşımı esnasında dikkat etmesi gereken husus, seçeceği uygulamanın stego-resim üzerinde sıkıştırma, kırpma, ölçeklendirme gibi işlemleri yapmamasıdır. Bu işlemlerin stego-resim üzerinde yapılması halinde stego-resim içerisindeki gizli bilginin bozulması veya kaybolması kaçınılmazdır.

Kullanıcı “GÖNDER” butonuna basıp resim gönderim işleminin tamamladıktan sonra, resim Openfire sunucusu üzerinden diğer kullanıcıya iletilir. Mesaj gönderilen

kullanıcı eğer o anda giriş işlemi yapmadıysa ve herhangi bir sebeple (internet, bağlantı eksikliği vb.) mesajı alacak durumda değil ise Openfire sunucusunun "Offline Messages" çevrimdışı mesaj işleyiş politikalarından birisi uygulanır.



Şekil 4.10. Openfire çevrimdışı mesaj iletim politikaları

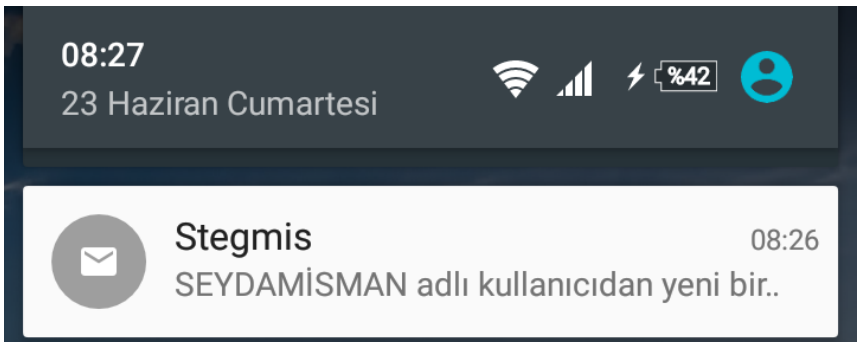
Bu politikalar 3 ana başlık altında tanımlanmıştır.

Depolama: Bu politikada çevrimdışı mesajlar depolanır. Bu politika kapsamında kullanıcı bazlı depolama limiti belirlenmiştir. Çevrimdışı mesajların kapladığı alan da hemen ayar menüsünün üstünde gösterilmiştir. Bu depolama limitinin dolmasına binaen 3 farklı politika ortaya konulmuştur. Bunlar depolama limiti aşıldığında mesajın kullanıcıya geri iletilmesi, depolama limiti aşılsa bile mesajın depolanmaya devam edilmesi ve depolama limiti aşılırsa gönderilen mesajın silinmesi süreçleridir.

Geri iletme: Bu politikada çevrimdışı mesaj geri gönderen kullanıcı gönderilir.

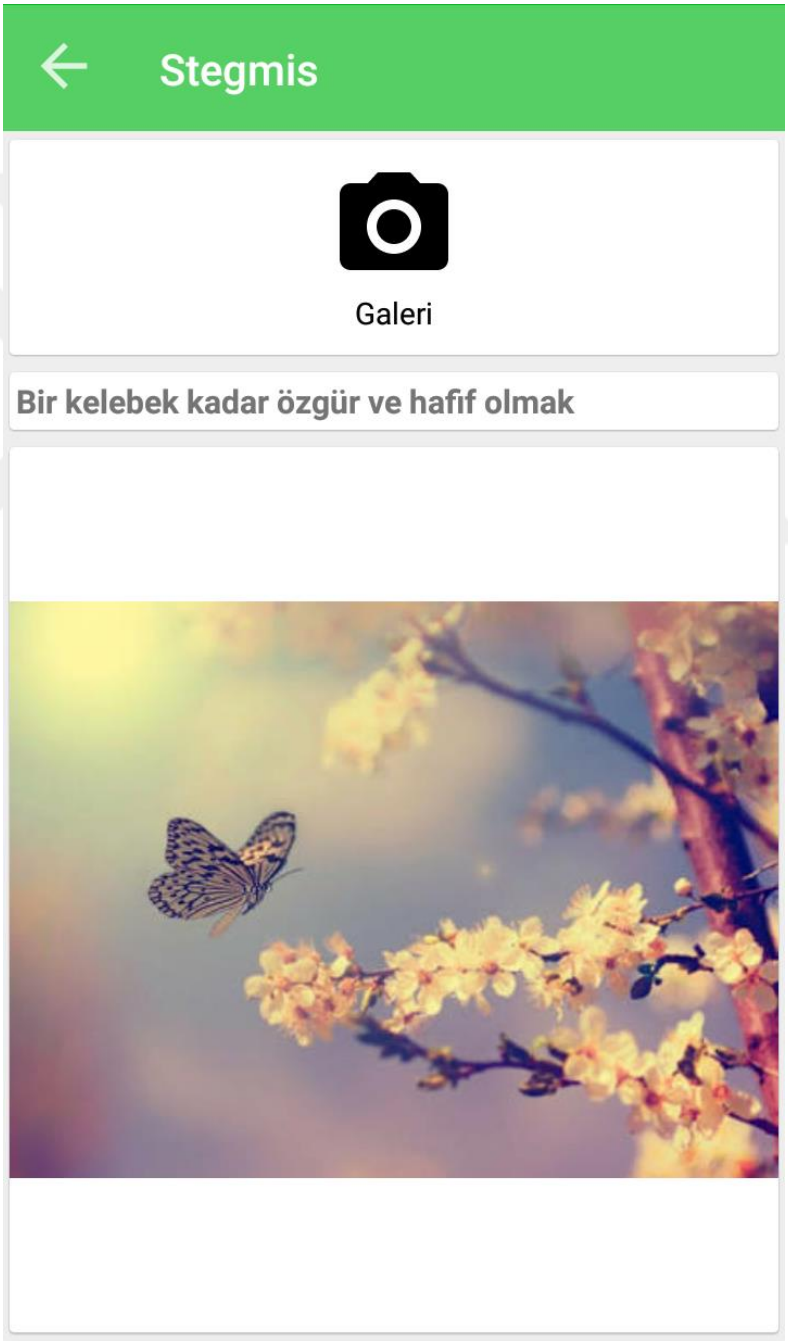
Silme: Bu politikada çevrimdışı mesaj geri gönderilmeden silinir.

Kullanıcıya stego-resim mesajı geldiği bildirim ile Şekil 4.11. deki gibi gösterilir.



Şekil 4.11. Bildirim Gelmesi

Kullanıcı bildirime tıkladığında, uygulama anlık olarak mesajın piksel baytlarının en önemsiz 3 er bitlerinin oluşturduğu 32 bitlik değeri tamamlayarak gömülen mesaj uzunluğunu belirler. Daha sonra bu sayı nispetinde bit okuma işlemine devam edip şifrelenmiş gizli mesajın oluşturulması sağlanır. Oluşturulan şifreli gizli mesaj kişinin kendi kullanıcı adının SHA-256 özet değeri ile elde edilmiş anahtar ile çözülür. Bu sırada uygulamadaki resme gizlenen mesajı göster ekranı açılarak çözülen gizli mesaj ve stego-resim Şekil 4.12 deki gibi ekranda gösterilir.



Şekil 4.12. Stego-resim içerisindeki gizli bilginin ortaya çıkarılması

5. SONUÇLAR VE ÖNERİLER

Steganografi günümüzün popüler ve ilgi çekici konularından biridir. Aslında bakıldığında steganografinin geçmişteki askeri, siyasi, sosyal, iletişim, güvenliğin sağlanması ve zafiyetinin ortadan kalkması gibi amaçları günümüzde de devam etmekte olup değişen yalnızca araçlardır. Teknoloji ve bilgisayar sistemleri, her konuda olduğu gibi steganografiye de geniş bir araç havuzu sağlamış, bu araç havuzlarıyla da kendi özelliklerine göre yeni yöntem havuzları önerilmiş ve üretilmiştir. Bu yöntem havuzları da bir zorunluluk ve tamamlanmış bir çalışma olmaktan ziyade bu alanda çalışan insanlar için bir ilham ve insan ufkunu genişleten mühendislik yapıları olarak tasavvur edilmelidir ve öyledir. Bu açıdan bakıldığında steganografinin bir sanat olması yani kendini tekrar eden ve sabit yöntem ve formülleri olan determinizm anlayışından çıkarıp, bireye has bir takım iletişim yöntemlerinin tasarlanabilir ve üretilebilir olması sonucuna ulaşılmaktadır.

Steganografi algoritmalarında genel olarak üç temel özellik öne çıkmaktadır. Bunlar güvenlik, anlaşılmazlık ve mesaj saklama kapasitesidir. Bununla beraber geliştirilen birçok yöntem mesaj saklandığının ortaya çıkması ve saklanan bilginin bir şekilde kaybolması veya zarar görmesi konularında yetersiz ve güçsüz kalmaktadır. Bir sistemin güçlü olarak telakki edilebilmesi için şu özelliklere sahip olması gerekmektedir.

- Gizli mesaj saklama işleminden sonra dijital örtü nesnesinin kalitesinde algılanabilir herhangi bir değişiklik olmamalıdır.
- Gizli bilgi bir anahtar olmadan anlaşılmamalıdır.
- Birden fazla mesaj saklama işlemi yapıldığından birbirine karışmamalıdır.
- Stego-nesnesi kalitesi üzerinde değişiklik yapmayan saldırılara karşı dayanıklı olmalıdır.

5.1 Sonuçlar

Bu çalışmamızda steganografinin tarihsel süreçte nasıl tezahürlerinin olduğu örneklerle açıklanmıştır. Daha sonra günümüzde dijital dünyadaki steganografi alanında yapılan bazı çalışmalardan bahsedilmiştir. Dijital alandaki örtü verisine göre

sınıflandırılan steganografi tipleri (ses, video, yazı, resim) ve bu alanlarda kullanılan ve geliştirilen algoritma ve yöntemler açıklanmıştır.

Günümüzde mobil telefonların başını iki mobil işletim sistemi çekmektedir. Bunlar Google'ın destekleyip geliştirdiği açık kaynak kodlu yazılım olan android ve Apple şirketinin destekleyip geliştirdiği iOS'tur. Bu tez çalışması kapsamında bir android mobil uygulama geliştirilmiştir. Sunucu ve istemci mimarisi ile geliştirilen bu uygulamada kullanılan teknolojiler ve özellikleri açıklanmıştır. Bu uygulama ile sisteme üye olan kullanıcılar arasında güvenli bir iletişim sağlanması amaçlanmıştır. Gerçek zamanlı çalışan bu uygulama ile kullanıcıya örtü resmini ve iletmek istediği gizli mesajı seçmesine imkân verilir. Seçilen bu mesaj, mesaj iletilecek kişinin kullanıcı adının SHA-256 özet değeri ile oluşturulan bir anahtar vasıtasıyla AES algoritması kullanılarak şifrelenir. Şifrelenen bu veri bitlerine ayrıştırılarak örtü verisinin piksellerine her baytına 3 bit olacak şekilde LSB yöntemi kullanılarak saklanmıştır. Bununla iki dereceli güvenlik sağlanmıştır;

- Üçüncü istenmeyen şahısların gizli bilgiye ulaşması, şifrelenmiş gizli bilgiye ulaşması anlamını taşımaktadır. Bu da gizli bilgiye ulaşmak için yeterli değildir.
- Gizli bilgiye yalnızca o kullanıcı adı ile sisteme giriş yapabilen ve stego-resme sahip olan kişiler ulaşabilmektedir.

Gizli bilgi içeren stego-resim Openfire Sunucusu üzerinden veya resim boyutunu değiştirmeyecek diğer uygulamalar ile diğer kullanıcıya iletilebilir.

Aynı zamanda bu uygulamayla kullanıcıya kişisel bilgilerinin saklanması ve depolanması özelliği de sunulmaktadır.

5.2 Öneriler

Uygulamamızda mesajın örtü resmi içerisine saklama aşamasında herhangi bir dağılım algoritması kullanılmadan basit LSB yöntemi kullanılmaktadır. Gelecek çalışmalarda; kullanıcının tercihine binaen F5, DCT ve PVD algoritmaları da desteklenebilir.

Mesajın resim örtü verisine gömülmesi aşamasında standart olarak her baytın en önemsiz 3 biti kullanılmaktadır. Kullanıcının tercihine bırakılıp, sekize kadar bit sayısı desteklenebilir.

Mesajın şifrelenmesinde simetrik anahtar kullanan AES algoritması yerine Diffie Hellman, RSA gibi asimetrik anahtar kullanan algoritmalar kullanılarak güvenliğin geliştirilebileceği düşünülmektedir.

Uygulamanın kapsamı geliştirilerek, stego resimlere, stego resim olduklarını belirleyici bazı bayt değerleri atanır. Uygulamayı kullanan her bir kullanıcının galeri resim ekleme işlemleri takip edilir ve her resim taranarak gelen resmin stego resim olup olmadığı kontrol edilir. Eğer mesaj saklanmış bir resim ise otomatik olarak içindeki mesaj ortaya çıkarılarak kullanıcıya gösterilebilir.

KAYNAKLAR

Al-Frajat, A. K., Jalab, H. A., Kasirun, Z. M., Zaidan, A. A., & Zaidan, B. B., 2010, Hiding data in video file: An overview. Journal of Applied Sciences(Faisalabad), 10(15), 1644-1649.

Bateman, P., & Schaathun, H. G., 2008, Image steganography and steganalysis. *Department Of Computing, Faculty of Engineering and Physical Sciences, University of Surrey, Guildford, Surrey, United Kingdom, 4th August*.

Bender, W., Gruhl, D., Morimoto, N., & Lu, A., 1996, Techniques for data hiding. *IBM systems journal*, *35*(3.4), 313-336.

Bhaumik, A. K., Choi, M., Robles, R. J., & Balitanas, M. O., 2009, Data hiding in video. International Journal of Database Theory and Application, 2(2), 9-16.

Brassil, J. T., Low, S., Maxemchuk, N. F., & O'Gorman, L., 1995, Electronic marking and identification techniques to discourage document copying. IEEE Journal on Selected Areas in Communications, 13(8), 1495-1504.

Brewster D. Microscope, 1857, volume XIV. Encyclopedia Britannica or the Dictionary of Arts, Sciences, and General Literature, Edinburgh, IX-Application of photography to the microscope, 8th edition.

Channalli, S., & Jadhav, A., 2009, Steganography an Art of Hiding Data, International Journal on Computer Science and Engineering, IJCSE, vol. 1, no. 3.

Chaumont, M., & Puech, W., 2006 September, A DCT-based data-hiding method to embed the color information in a JPEG grey level image. In Signal Processing Conference, 2006 14th European (pp. 1-5). IEEE.

Distribution dashboard [online], https://developer.android.com/about/dashboards/, Haziran 2018.

Dun Hu, Sheng & Tak U, Kin., 2011, A Novel Video Steganography Based on Non-uniform Rectangular Partition. 57-61. 10.1109/CSE.2011.24.

E. H. Wilkins, 1954, A History of Italian Literature, Oxford University Press, London. Corliss, R., 1993, *Pacific Overtures Times*, 142 (11), 68-70.

Eltahir, M. E., Kiah, L. M., Zaidan, B. B., & Zaidan, A. A., 2009 April, High rate video streaming steganography. In *Information Management and Engineering, 2009. ICIME'09. International Conference on* (pp. 550-553). IEEE.

F. Bacon, 1640, Of the advancement and proficiencies of learning or the partitions of sciences, volume VI. Leon Lichfield, Oxford, for R. Young and E. Forest.

Gang, L., Akansu, A. N., & Ramkumar, M., 2001, MP3 resistant oblivious steganography. In Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on (Vol. 3, pp. 1365-1368). IEEE.

Gülcü, Ş., Gülcü S., 2013, Improving of the LSB Based Steganography Technique using Multiple LSBs and Discrete Logarithm, *International Symposium on Digital Forensics and Security*

Gürkan, A. T., Şubat 2016, gerçek zamanlı mobil steganografi uygulaması geliştirilmesi, Yüksek Lisans Tezi, *Gazi Üniversitesi Fen Bilimleri Enstitüsü,* Ankara

Hemalatha, S., Acharya, U. D., & Renuka, A., 2013, Comparison of secure and high capacity color image steganography techniques in RGB and YCbCr domains, International Journal of Advanced Information Technology, Vol.3, No.3, pp.1-9.

Herodotus, 1996, The Histories. Penguin Books, London, Translated by Aubrey de S´elincourt.

Hussain, M., & Hussain, M., 2013 May, A survey of image steganography techniques, International Journal of Advanced Science and Technology, Vol. 54.

Jayaram, P., Ranganatha, H. R., & Anupama, H. S. (2011). Information hiding using audio steganography–a survey. The International Journal of Multimedia & Its Applications (IJMA) Vol, 3, 86-96.

Johnson, N. F., & Jajodia, S., 1998, Exploring steganography: Seeing the unseen. *Computer*, pp.26 − 34.

Juneja, M., & Sandhu, P. S., 2013, A new approach for information security using an improved steganography technique. Journal of Information Processing Systems, 9(3), 405-424.

Kabetta, H., B. Y. Dwiandiyanta, and Suyoto, 2011, "Information hiding in CSS: a secure scheme text steganography using public key cryptosystem," Int. Journal on Cryptography and Information Security, vol.1, pp. 13-22.

Kahn, D., 1984, Cryptology and the origins of spread spectrum: Engineers during World War II developed an unbreakable scrambler to guarantee secure communications between Allied leaders; actress Hedy Lamarr played a role in the technology. IEEE spectrum, 21(9), 70-80.

Kapoor, V., & Mirza, A., 2015, "An Enhanced LSB based Video Steganographic System for Secure and Efficient Data Transmission", International Journal of Computer Applications (0975 – 8887) Volume 121 – No.10, July 2015

Kapotas, S. K., Varsaki, E. E., & Skodras, A. N, 2007 October, Data hiding in H. 264 encoded video sequences. In Multimedia Signal Processing, 2007. MMSP 2007. IEEE 9th Workshop on (pp. 373-376). IEEE.

Katzenbeisser, S., & Petitcolas, F., 2000, A Survey of steganographic techniques in Information Hiding Techniques for Steganography and Digital Watermarking, Ed. London: Artech House, pp. 43-78.

Khan, M. S., Bhaskar, M. V., & Nagaraju, M. S., 2011, An optimized method for concealing data using audio steganography. International Journal of Computer Applications (0975–8887) Volume, 25-30.

Kruus, P., Scace, C., Heyman, M., & Mundy, M., 2003, A survey of steganography techniques for image files. Advanced Security Research Journal. [Online], 5(1), 41-52.

Laskar, S. A., & Hemachandran, K., 2013, Steganography Based On Random Pixel Selection For Efficient Data Hiding, International Journal of Computer Engineering and Technology, Vol.4, Issue 2, pp.31-44.

Meet Android Studio [online], https://developer.android.com/studio/intro [Ziyaret Tarihi: Haziran 2018]

Monika Agarwal, January 2013, International Journal of Network Security & Its Applications (IJNSA), Vol.5, No.1, Text Steganographıc Approaches: A Comparison

Mohamed Amin, Muhalim & Salleh, Mazleena & Ibrahim, Subariah & Rozi b. Katmin, Mohd. (2003). Stenography: Random LSB Insertion Using Discrete Logarithm 234-238.

Nasereddin, Hebah & Abdallah, Yaman, 2016, Enhancing Open Space Method in Data Hiding Technique. International Journal of Computer Applications. 155. 5-17.

Raja, K. B., Chowdary, C. R., Venugopal, K. R., & Patnaik, L. M., 2005 December, A secure image steganography using LSB, DCT and compression techniques on raw images. In *Intelligent Sensing and Information Processing, 2005. ICISIP 2005. Third International Conference on* (pp. 170-176). IEEE.

Reddy, H. M., & Raja, K. B., 2009, High capacity and security steganography using discrete wavelet transform. International Journal of Computer Science and Security (IJCSS), 3(6), 462.

Sarangpure, V., Talmale, R., & Domke, M., 2014, Survey paper - Audio-Video Steganography Using Anti Forensics Technique. International Journal of Research [online],1(9),661-665.
https://edupediapublications.org/journals/index.php/IJR/article/view/678/309
[Ziyaret Tarihi: Haziran 2018]

Shanableh, Tamer., 2012, Matrix encoding for data hiding using multilayer video coding and transcoding solutions. Signal Processing: Image Communication. 27. 1025–1034. 10.1016/j.image.2012.06.003.

Shanthakumari, R., & Malliga, D. S., 2014, Video Steganography using LSB matching revisited algorithm", IOSR Journal of Computer Engineering, Volume 16, Issue 6, Ver. IV, PP 01-06

Sharmila, B., & Shanthakumari, R., 2012, Efficient Adaptive Steganography for color images based on LSBMR algorithm. ICTACT Journal on image and video processing, 2(03), 387-392.

Sherly, A. P., & Amritha, P. P., 2010, A compressed video steganography using TPVD. International Journal of Database Management Systems (IJDMS) Vol, 2(3).

Shirali-Shahreza, M. H., & Shirali-Shahreza, M., 2006 July, A new approach to Persian/Arabic text steganography. In Computer and Information Science, 2006 and 2006 1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse. ICIS-COMSAR 2006. 5th IEEE/ACIS International Conference on (pp. 310-315). IEEE.

Tacticius, A., 1990, How to Survive Under Siege/Aineas the Tactician, Clarendon ancient history series.

Upham, D., 1997, Jsteg, [online] ftp://ftp.funet.fi/pub/crypt/steganography

Westfeld, A., 2001 April, F5—a steganographic algorithm. In *International workshop on information hiding* (pp. 289-302). Springer, Berlin, Heidelberg.

Wu, D. C., & Tsai, W. H., 2013, A steganographic method for images by pixel-value differencing, Pattern Recognition Letters 24 (9–10), pp.1613-1626.

Yang, H., Sun, X., & Sun, G., 2009, A High-Capacity Image Data Hiding Scheme Using Adaptive LSB Substitution, Journal: Radioengineering, vol. 18, no. 4, pp. 09-516

EKLER

EK-1 SMACK KÜTÜPHANESİ İŞLEMLERİ

Smack Kütüphanesinin STEGMIS Uygulamamıza Tanımlanması

Smack kütüphanesini STEGMIS projemizde kullanabilmemiz için “app” modülüne ait klasörün içindeki “build.gradle” dosyamızın içerisindeki “dependencies” alanına aşağıdaki kütüphane adreslerini eklememiz gerekmektedir.

implementation "org.igniterealtime.smack:smack-android-extensions:4.2.0"
implementation "org.igniterealtime.smack:smack-tcp:4.2.0"

Bu adresler eklendikten sonra Android Studio bizden gradle dosyasını yeniden inşa edilmesini talep edecektir. Kabul ettiğimiz takdirde, Android Studio gerekli kütüphaneleri indirerek düzenleyecek ve artık bu sınıflar ve kütüphaneler bizim için kullanılabilir durumda olacaktır.

Openfire Sunucusu ile Bağlantı Kurulması

Gerekli kütüphaneler eklendikten sonra android istemcisinin sunucu ile nasıl bağlantı kurması gerektiği aşağıda verilen kod örneğinde gösterilmiştir.

```
AbstractXMPPConnection connection=null;
try {
    XMPPTCPConnectionConfiguration connConfig =
            XMPPTCPConnectionConfiguration.builder()
                    .setHost(GeneralUtil.XMPP_HOST)
                    .setXmppDomain(GeneralUtil.XMPP_DOMAIN)
                    .setSecurityMode(ConnectionConfiguration.SecurityMode.disabled)
                    .setPort(GeneralUtil.XMPP_PORT)
                    .setDebuggerEnabled(true)
                    .build();
    connection = new XMPPTCPConnection(connConfig);
    connection.connect();
    AccountManager accountManager = AccountManager.getInstance(connection);
    if (accountManager.supportsAccountCreation()) {
        accountManager.sensitiveOperationOverInsecureConnection( allow: true);
        accountManager.createAccount(Localpart.from(prms[0]), prms[1]);
    } else {
        result="-1";
    }
}catch (SmackException | IOException | XMPPException e) {
    Log.e( tag: "TAG", e.getMessage());
    result="0";
} catch (InterruptedException e) {
    e.printStackTrace();
    result="0";
}
```

Şekil 7.1.1. Android istemcisinin Openfire sunucusu ile bağlantısı

Şekil 7.1.1 de gösterilen kod örneğindeki parametreler ve açıklamaları aşağıdaki tabloda verilmiştir.

Çizelge 7.1.1 Openfire sunucu bağlantı parametre tablosu

GeneralUtil.XMPP_HOST	Sunucunu ip adresini (Örn:192.168.2.179)
GeneralUtil.XMPP_DOMAIN	Openfire sunucusu kurulumunda atamış olduğumuz alan adını (Örn: stegmisman.com)
GeneralUtil.XMPP_PORT	Openfire sunucu portunu, standart: 5222

Yeni Kullanıcı Oluşturulması

Şekil 7.1.1 de bağlantı kurulmasından sonra (connection.connect()), 'AccountManager' sınıfı ile yeni kullanıcı oluşturulması gösterilmektedir. Şekilde de görüldüğü üzere 'AccountManager' sınıfının 'createAccount' fonksiyonu, prms[0] ve prms[1] alanları kullanıcı adı ve şifre ile çağırılarak yeni kullanıcı oluşturma işlemi gerçekleştirilir.

Kullanıcının Oturum Açma İşlemi

```
XMPPTCPConnectionConfiguration connConfig =
        XMPPTCPConnectionConfiguration.builder()
                .setHost(GeneralUtil.XMPP_HOST)  // Name of your Host
                .setXmppDomain(GeneralUtil.XMPP_DOMAIN)
                .setSecurityMode(ConnectionConfiguration.SecurityMode.disabled)
                .setPort(GeneralUtil.XMPP_PORT)
                .setUsernameAndPassword(prms[0],prms[1])
                .setDebuggerEnabled(true)
                .build();
connection = new XMPPTCPConnection(connConfig);
connection.connect();
StanzaCollector sc = connection.createStanzaCollector(new AndFilter(StanzaTypeFilter.MESSAGE));
StanzaListener stanza = (packet) -> {
        listener.afterLoginMessageCome(packet);
};
connection.addAsyncStanzaListener(stanza,sc.getStanzaFilter());
connection.login();
```

Şekil 7.1.2. Oturum açma işlemi

Yeni kullanıcı oluşturma işlemi tamamlandıktan sonra, yukarıdaki kod örneğinde görüldüğü üzere prms[0] ve prms[1] alanlarına kullanıcı adı ve şifre girilecek

kullanıcının oturum açması sağlanır. StanzaListener sınıfı bize sunucu tarafından gönderilen her mesaj paketini dinleme imkânı verir ve bu sayede gelen mesajların ihtiyaç kapsamına göre ayrıştırılmasına olanak sağlar.

Mesaj Gönderme İşlemi

Oturum açma işlemi tamamlandıktan sonra istemci tarafından Smack kütüphanesi ile Openfire sunucusuna mesaj gönderilme işleminin nasıl yapılacağından bahsedeceğiz.

```
boolean sendMessage(byte[] messagebytes, String touser) {
    ChatManager cm = ChatManager.getInstanceFor(connection);
    try {
        EntityBareJid jid = JidCreate.entityBareFrom( jid: touser + "@" + GeneralUtil.XMPP_DOMAIN);
        Chat chat1 = cm.chatWith(jid);
        Message m = new Message();
        String encodedAndCrypted = Base64.encodeToString(messagebytes, Base64.NO_WRAP);
        m.setBody(encodedAndCrypted);
        chat1.send(m);
        return true;
    } catch (XmppStringprepException e) {
        e.printStackTrace();
    } catch (SmackException.NotConnectedException e) {
        e.printStackTrace();
    } catch (InterruptedException e) {
        e.printStackTrace();
    }

    return false;

}
```

Şekil 7.1.3. Mesaj gönderme işlemi

Şekil 7.1.3'te görüldüğü üzere, herhangi bir kullanıcıya mesaj gönderebilmek için o kişinin kullanıcı adının bilinmesi gereklidir. Bu kullanıcı adı ile EntityBareJid sınıfından örnek model oluşturularak, daha önce Openfire sunucusu ile kurmuş olduğumuz bağlantı nesnesi ile ChatManager sınıfından yeni bir örnek nesne oluşturulur. ChatManager sınıfının 'chatWith' fonksiyonu ile Chat sınıf nesnesi üretilir. 'sendMessage' fonksiyonumuza gelen mesaj parametremiz bayt dizisi türünde olduğu için bu bayt dizi Base64 sınıfı yardımıyla kodlanır. Yazı veri tipine dönüştürülen mesaj Chat sınıfı 'send' fonksiyonu ile gönderilir. Buradan da anlaşılacağı üzere veri tipi yazı olan mesajlarda herhangi bir dönüştürme işlemine gerek yoktur.

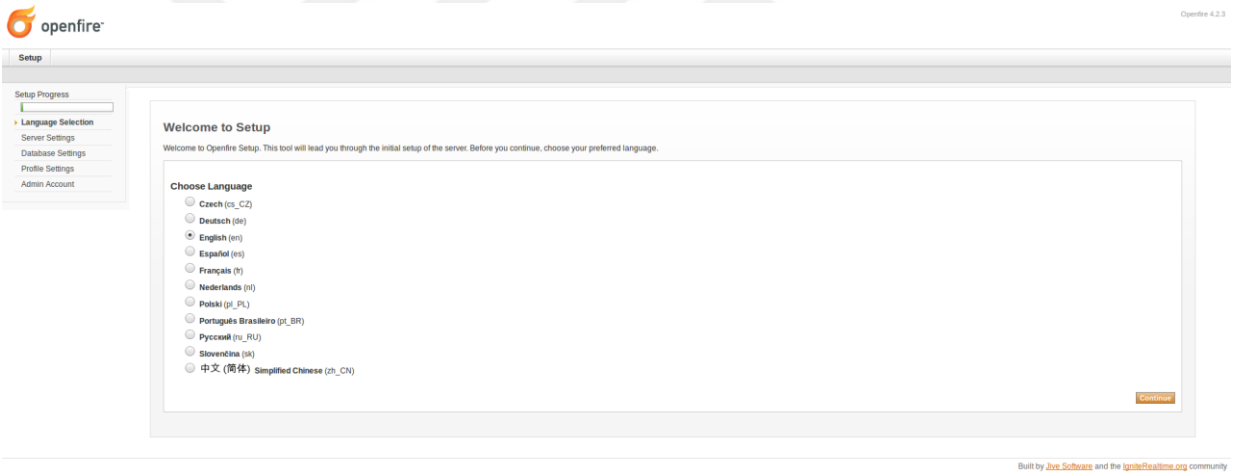
EK-2 OPENFIRE SUNUCU KURULUMU

1- Sistem terminalinden
"wget http://download.igniterealtime.org/openfire/openfire_4.2.3_all.deb" komutu ile yükleme dosyası indirilir.

2- "dpkg -i openfire_4.2.3_all.deb" komutu ile indirilen dosya çalıştırılarak Openfire Server yükleme işlemi yapılır.

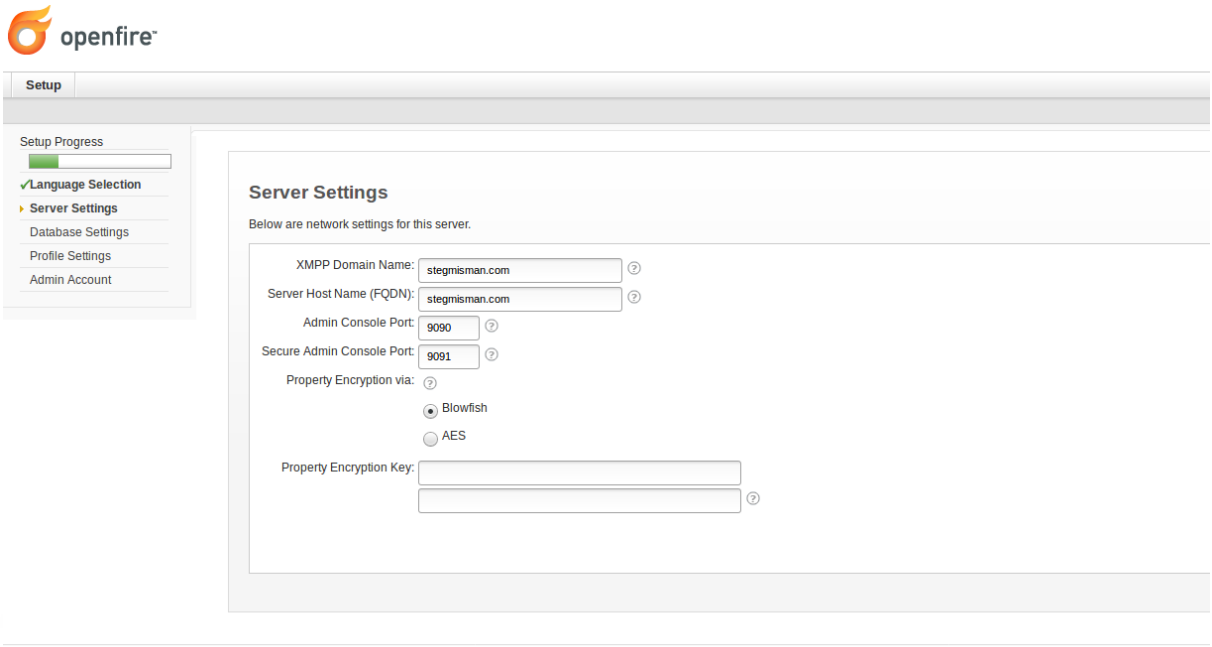
3- Yükleme işlemi tamamlandıktan sonra "service openfire start" komutu ile sunucu çalıştırılır.

4- Web tarayıcı ile "http://localhost:9090" adresine gidilir.



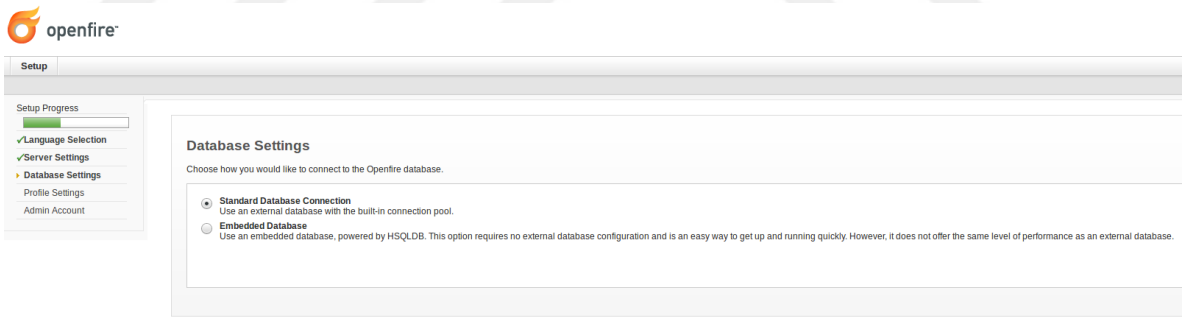
Şekil 7.2.1. Openfire Server kurulumu dil seçimi

5- Önümüze gelen ekranda tercih edilen sunucu dilini seçmemiz isteniyor. "English" seçerek "Continue" diyoruz.



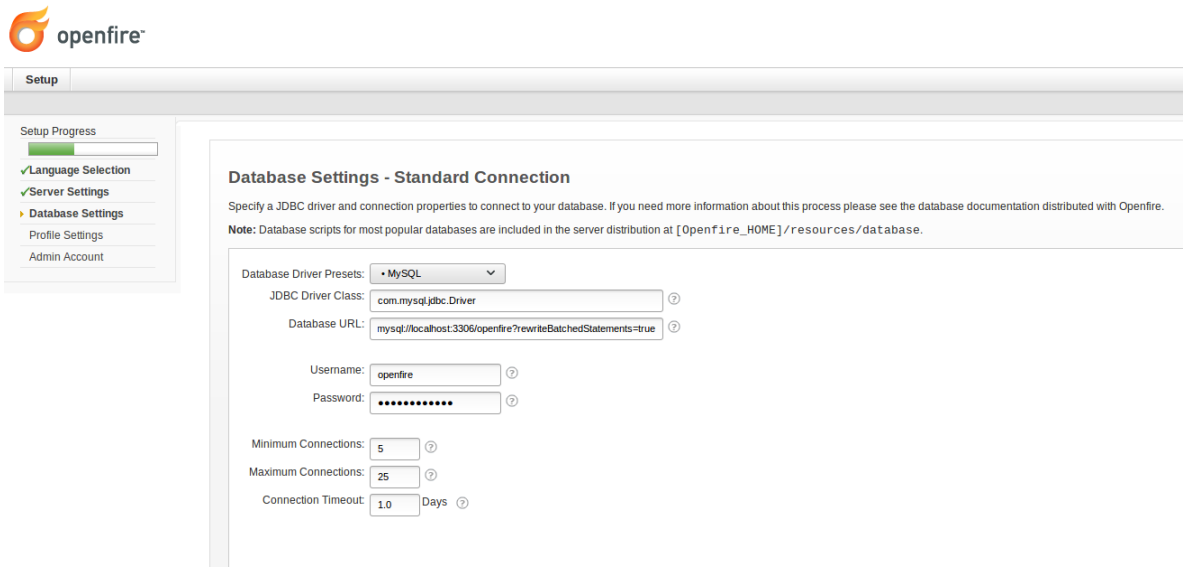
Şekil 7.2.2. Openfire Server kurulumu alan adı seçimi

6- Bu ekranda ise alan ismini ve sunucu ismi kısımlarını kullanacağımız alan adı ile doldurmalıyız. Biz "stegmisman.com" olarak doldurarak bir sonraki aşamaya geçiyoruz.



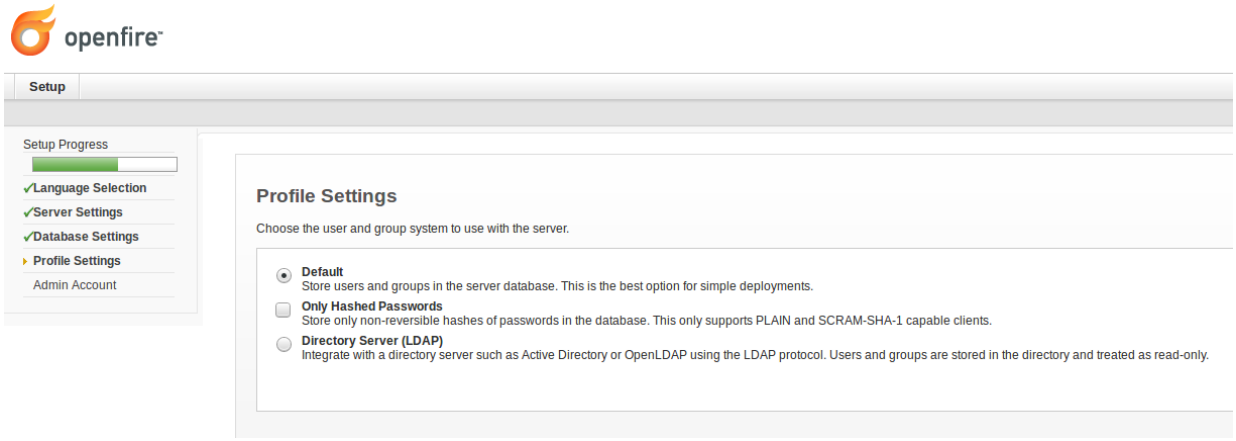
Şekil 7.2.3. Openfire Server veritabanı seçimi

7- Önümüze gelen veri tabanı ayarları kısmında; Openfire Server ile gömülü olarak gelen HSQLDB veritabanını mı kullanacağımızı yoksa başka bir veritabanı mı kullanacağımızı seçiyoruz. Biz MySQL veritabanı kullanacağımız için "Standart Database Connection" seçeneği ile devam ediyoruz.



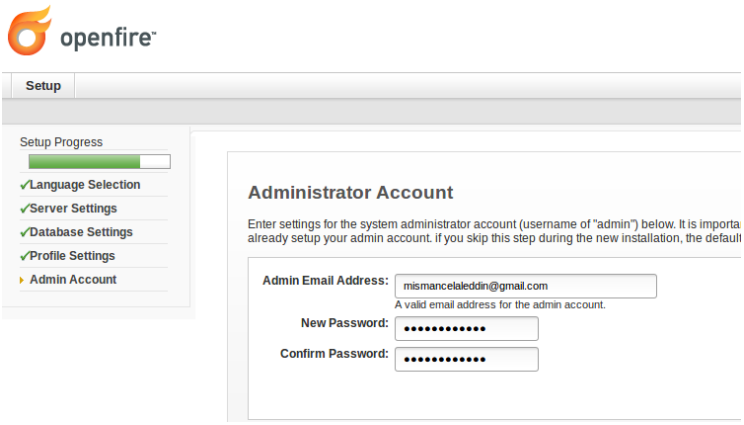
Şekil 7.2.4. Openfire Server veritabanı ayarları

8- Bu ekranda "Database Driver Presets" kısmına MySQL seçiyoruz. "JDBC Driver Class" alanı varsayılan kullanılarak, daha önceden "Openfire Server Veritabanı Bağlantısı İçin Ön Hazırlık" kısmında oluşturduğumuz veri tabanı ismini kullanıcı adını ve şifreyi dolduruyoruz. "Continue" dediğimiz anda Openfire Server veritabanı, kullanıcı adı ve şifre biligileri ile MySQL veri tabanımıza bağlanarak kendi sistemi için gerekli tablo ve verileri oluşturmaya başlayacaktır. Bu yüzden bu adım biraz zaman alabilir.



Şekil 7.2.5. Openfire Server kurulumu veri tutma yöntemi seçimi

9- Bu adımda kullanıcı ve grupların nasıl ve nerede tutulacağını seçiyoruz. "Default" seçeneği ile devam ediyoruz.



Şekil 7.2.6. Openfire Server kurulumu yönetim kullanıcısı şifre belirleme

10- Son olarak Openfire Server yönetici hesabımıza ait email adresimizi ve şifremizi belirleyerek işlemi tamamlıyoruz.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Celalettin Misman
Uyruğu : Türkiye Cumhuriyeti
Doğum Yeri ve Tarihi : KONYA – 15.05.1991
Telefon : +905416845558
Faks : -
e-mail : mismancelaleddin@gmail.com

EĞİTİM

Derece		**Adı, İlçe, İl**	**Bitirme Yılı**
Lise	:	**Meram Anadolu Lisesi, Meram, Konya**	**2009**
Üniversite	:	**İzmir Yüksek Teknoloji Enstitüsü, Urla, İzmir**	**2014**
Yüksek Lisans	:		
Doktora	:		

İŞ DENEYİMLERİ

Yıl	**Kurum**	**Görevi**
2014	**Mepsan A. Ş**	**Proje Yöneticisi**

YABANCI DİLLER

Arapça **Giriş Seviyesi**

Fransızca **Giriş Seviyesi**

İngilizce **İleri Seviye**

YAYINLAR

Increasing the security of Mobile Communication with Steganography, *International Conference on Advanced Technologies, Computer Engineering and Science (ICATCES 2018), Safronbulu, Turkey, 11-13 Mayıs 2018*