Resime Veri Gömme Nedir ?

Tarihçe : Steganografi İngilizce "Steganography" Adında Ve Yunanca "Steganos (Gizlenmiş,Örtlü,Korunmuş)" ve "Graphein(Yazı)" kelimelerinden oluşmaktadır .Eski Yunancada "gizlenmiş yazı" anlamına gelir ve bilgiyi gizleme bilimine verilen isimdir.

Temel Prensipleri : Değişimin Fark Edilememesi , Saklanabilecek Veri Miktarı , Dayanıklılık

Çalışma Süreci : Basitçe **,** Kullanılan algoritmaya göre şifrelenecek metin alınıp veri gömme işlemi yapılıcak metine algoritmaya bağlı olarak işlemler yaparak şifreleme işlemini gerçekleştirir.

LEAST SIGNIFICANT BIT (LSB) INSERTION YÖNTEMİ :

Steganografide en fazla kullanılan tekniklerden biri Least Siginificant (LSB-en az öneme sahip bit) Insertion yöntemidir. Uygulaması çok basit olan bir yöntem olmasına karşın dikkatsizce uygulanması durumunda **veri kayıpları ortaya çıkmaktadır**. Bu yöntemde, gizlenecek verinin her biti, resim verisinin bir baytının son bitine yazılır. Örnek olarak bir 24 bit resim dosyasının, üç pikselinin ilk sekiz baytına "A" harfinin gömme işlemi

Pikseller: (00100111 11101001 11001000) (00100111 11001000 11101001) (11001000 00100111

11101001)

A: 01000001

Sonuç: (00100110 11101001 11001000) (00100110 11001000 11101000) (11001000 00100111 11101001)

Avantajları	Dezavantajları(Eksikler)
Uygulaması Basit	Veri Kaybolma İhtimali
Renkli dijital Görüntüler Eklenebilir (8-24 Bit)	Veri Sınırı Kısıtlaması
Tanınmışlık	Güvenilirlik Seviyesi Düşüklüğü
	Esneklik (Geliştirmeye Açık Olmayan)

1. **Algoritması : A)İşleme :**
   1)Pikseller Alınır
   2) R,G,B'nin Her Bir Değeri İçin LSB Sıfıra Eşitlenir. Bu Bitler Karakteri Gizlemek İçin Kullanılacaktır.
   3)Her Karakter Tek Tek İşleneceği İçin Ele Alınan Karakter Tam Sayıya Çevrilir .
   4)8 Bit İşlendiğinde Bir Sonraki Karaktere Geçilir Ve Tüm İşlem Bitene Kadar Bu Süreç Devam Eder.
   B)Çıkarma :
   1) 8 Ardışık Sıfır Bulunana Kadar Pikseller Alınır.
   2) LSB Bitleri Bulunur. Bu Karakterler Bitene Kadar Devam Eder Ve Çıktı Verecek Olan Sonuç Metine Aktarılır.

Resmin İlk Piksellerine Gömme Yöntemi :

Bu yöntemde gizlenecek metin , veri şifrelemesi gerçekleştirilecek resimin ilk Piksellerine (First Pixels) Gömme işlemini doğrudan sıralı olarak yapmaktadır.

Avantajları	Dezavantajları(Eksikler)
İşleyişi Basit	Güvenlik Seviyesi
	Bilinme Derecesi
Anlaşılabilir Kolay Akış Şeması	Karakter Sınırı
	Esneklik

Algoritması :

İşleme ;

1. Resmin pikseller alınır
2. Şifrelenecek Metin Alınır.
3. Resimin İlk pikselinden başlıyarak sıralı bir biçimde veriyi gömer.

Çözme :

1) Resmin ilk pikselinden Başlıyarak şifrelenmiş metni sıralı bir biçimde çözer

DHY Algoritması :

Algoritmasından kod tasarımına herşeyi grupça geliştirilmiş bu algoritmada tanınmamışlığın verdiği Avantajdan doğan bir üstünlük bulunmaktadır. Resimlere özel olarak üretilen 'Key' Sistemi ile her resimde aşılması zorlaşan çözümleme işlemine ek olarak 'MD5' yöntemi kullanılarak güvenlik daha güçlü bir hale getirilmiştir. Kullanılan algoritmadan kaynaklı olarak veri sınırı sorunu baya aşılmıştır. Ve geliştirmeye esnek bir yapıdadır.

Avantajları	Dezavantajları(Eksikler)
Kullanımı Basit	**Tanınmışlık Derecesi**
Resim Formatı Esnekliği	
Esneklik (Geliştirmeye Açık)	
Veri Sınırı Olmaması	
Güvenlik Seviyesi Yüksekliği	

Algoritması :

İşleme ;

1) Şifrelenecek Resim Eklenir (Eğer Jpg İse Döngü İçerisinde Pngye Çevrilir)
2) Şifrelenecek Metin İşleme Alınır
3) Resmin pixellerini veriyi gömebilmek ve overflow hatasından kurutula bilmek için 6'ya göre mod'unu alır.Kalan Piksellerden Veriye gömme işlemi başlatılır.
4) N ve N+1 değerlerinin arasındaki blue değer farkını alır
5) Resmin pixel sayısına bölerek ortalama bir fark çıkartıyoruz bu farkda bizim key'imiz olmaktadır.
6) Key bulundukdan sonra resimde gömüceğimiz verinin uzunluğu kadar aralarında key kadar fark olan n ve n+1 pixellerini bozmadan bırakılır.
7) Geri kalan pixellerin N ve N+1 olanlarında key kadar fak bulunursa n.blue-1 ve N+1.Blue+1 şeklinde Blue değerlerinin arasındaki farkı bozuyoruz.
8) Veriyi Gizleme İşlemi Başlatılır.
9) Bir hata olmaması açısından tekrar aralarında key kadar fark olan N ve N+1 pixellerini buluyoruz
10) Veriyi karakter Karakter N+2.pixelin Blue değerine ASCII değerlerini işliyoruz
11) Verinin gömülmediği yerlerin n ve n+1 pixellerinin blue değerleri key'imizle eşleşmemesine rağmen yinede verinin sonuna belirlediğimiz 3 byte'lık sınır belirlemek için yerleştirdiğimiz karakterlerlede verinin sonunu doğrulamış oluyoruz.

12) Farklı bir key kullanarak şifreleme yapıyor ve veriyi resme gizlemeden önce md5 ile veriyi bir kez daha şifreliyor