# SafeGate

### Github Adresi

https://github.com/49720Orange/SafeGate

### Grubu oluşturan öğretmenlerin listesi

Neslihan GENÇ, Emine AYDOĞMUŞ, İsmail Ersin TURAN

# SafeGate Analiz Raporu

### Kısa Özet

Modern teknoloji sürekli olarak gelişirken, artan veri alışverişi siber tehditleri önemli bir gündem maddesi haline getiriyor. Siber tehditler arttıkça doğal olarak verilerimizi korumaya ve güvenli hale getirmeye daha da çok ihtiyaç duyuyoruz. Veri şifreleme bu açıdan en önemli yöntemlerden biri. Kullanımı kolay her kullanıcı tarafından kolaylıkla anlaşılabilen ve etkileşime izin veren bir görsel tasarım ile veri şifrelemeyi sağlayacağız. Ayrıca ağ güvenliği diğer önemli konulardan biridir ve bunu sağlamak için etik hackleme araçları kullanılmaktadır. Normalde bu araçlar bazı işletim sistemleri üzerinde yazılıp kullanılmasına rağmen biz bu projede web uygulaması üzerine birkaç etik hackleme aracı ekleyip kullanıcıların test etmesini sağlayacağız.

### Problem Tanımı

Gönderdiğiniz verinin güvenliğinden emin misiniz?

Verinin güvenilmeyen ağ yapılarından geçerken veya uygulama veri tabanları içinde durduğu durumlarda şifreli halde bulunmasını sağlayarak güvenliği ve gizliliği nasıl sağlarız?

### Analiz Süreci

#### İhtiyaç Analizi

Günümüz koşullarında artık herkes online olarak evden çalışıyor veya arkadaşları ile mesajlaşarak sosyalleşiyor. Firmalar için gizli sayılacak verileri veya sosyal hayatta sır olarak söyleyeceğimiz herşeyi mesajlar yoluyla gönderiyoruz. Verinin güvenilmeyen ağ yapılarından geçerken veya uygulama veri tabanları içinde durduğu durumlarda şifreli halde bulunmasını sağlayarak güvenliği ve gizliliğini sağlamalıyız.

#### İçerik Analizi

Hazırladığımız bir web arayüzü sayesinde mesajların istenilen algoritma ile şifrelenerek ya da deşifrelenerek gösterilmesi sağlanacaktır. Ağ içerisinde kullanılabilecek etik hackleme araçlarıyla ağın durumu gözlemlenecektir. Örneğin Port Tarama aracıyla açık olan portlar listelenecektir.

#### Durum Ortam Analizi

Planladığımız projeyi PyCharm 2020 2.3 Professional Edition üzerinde geliştireceğiz. Projemiz web tabanlı bir uygulama olduğu için Flask ile geliştirilecektir. Tasarım kısmında HTML, Bootstrap 5.0, Javascript ve CSS3 araçları kullanılacaktır. Şifreleme algoritmaları ile ilgili bölümde uygun kütüphaneler kullanılacaktır. Örneğin AES şifreleme için AES kütüphanesi kullanılması planlanmıştır. Mail gönderme için Flask\_Mail kütüphanesi kullanılacaktır. Etik hackleme araçları eklemek için de bunlara uygun kütüphanelerle proje geliştirilecektir.

#### Kullanıcı Analizi

Yapacağımız proje GitHub’a yükleneceği için her zaman açık kaynak bir özelliğe sahip olacaktır. Siber güvenlik ve kriptoloji ile ilgilenen herkesin kullanımına uygundur.

# SafeGate Tasarım Raporu

## Kısa Özet

Projemiz bir web uygulaması olduğu için Flask ve Django kütüphanelerini araştırıp Flask kullanarak geliştirmeye karar verdik. Web uygulamasının tek sayfa üzerinde görüntüleyip bazı uygulamalarda sonuçları göstermek için farklı sayfalar kullanılacak şekilde tasarlamaya karar verdik. Uygulamamız Kriptografi yani şifreleme ve bazı etik hackleme araçları konusunda olduğu herhangi bir veritabanına ihtiyaç olmayacağı için web sayfamızın hakkımızda kısmında gösterilecek bilgileri veri tabanından çekerek görüntülemeyi tasarladık. Ayrıca etik hackleme araçlarından birkaçını ekleyebileceğimizi kod tasarım kısmında gösterdik, ancak uygulamanın geliştirilme aşamasında farklı birkaç tane daha araç ekleyebiliriz.

## Veri Tasarımı

Geliştireceğimiz SafeGate isimli web uygulaması içerik ve geliştirilecek uygulamalar bazında veritabanına ihtiyaç duymamaktadır. Ancak uygulama içinde Hakkımızda kısmında projeyi yürüten bilgilerini veritabanından çekip görüntüleyeceğiz. Veritabanı için Sqlite3 kullanacağız.

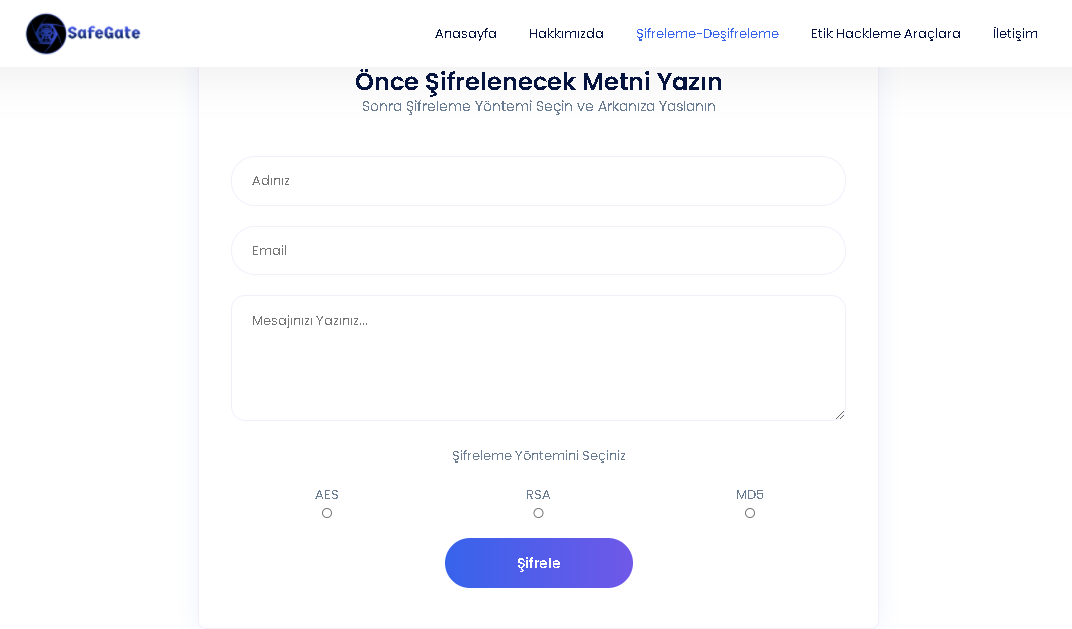
|  |  |
| --- | --- |
| **SafeGate (Veritabanı Adı)** | |
| **hocaBilgileri (Tablo Adı)** | |
| **id** | int, primary key |
| **ad** | varchar |
| **soyad** | varchar |
| **okuladi** | varchar |
| **email** | varchar |

## Ara yüz Tasarımı

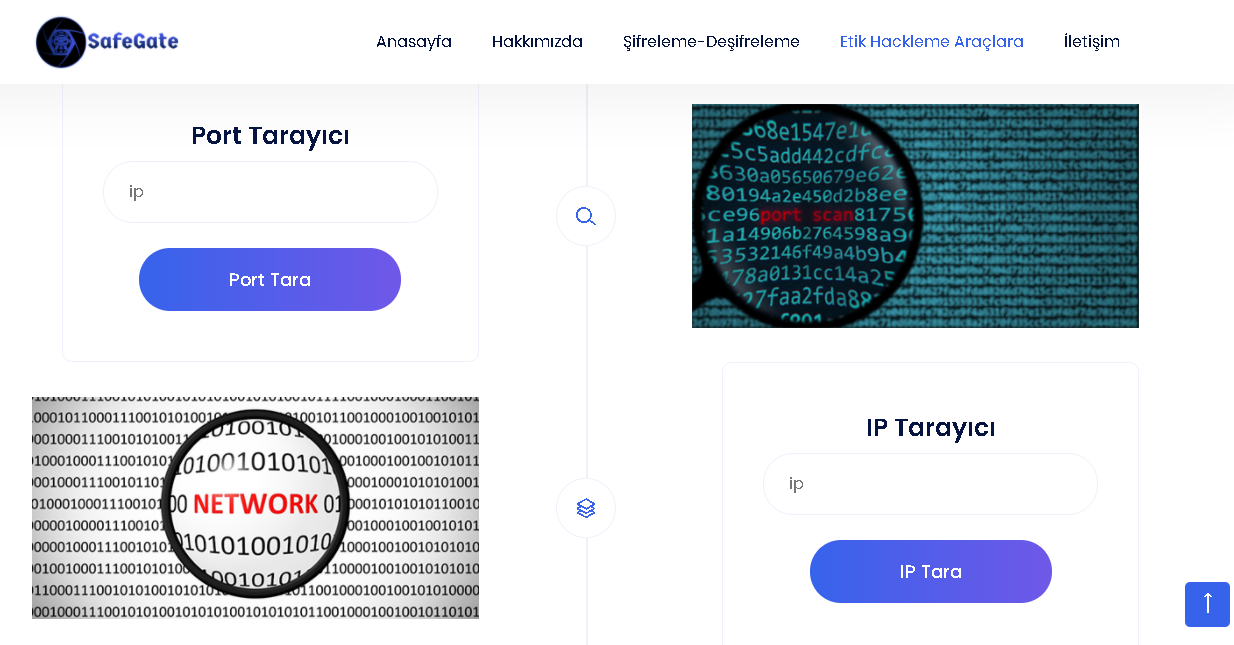
Web uygulaması geliştirirken HTML, CSS3, Bootstrap, JavaScript dilleri kullanıldı. Hazır bir tema üzerinden gerekli gerekli değişiklikler yapılarak bir web sayfası tasarladık. Projemizin adına ve web uygulamamıza uygun bir logo tasarladık ve kullandık. Tek sayfalı olarak tasarladığımız web sayfasında flask ile geliştirdiğimiz uygulamalarının sonuçları farklı sayfalar kullanılarak gösterilecektir.



**Şekil 1- SafeGate Anasayfa ve Kullanılan Linkler**



**Şekil 2- Şifreleme ve Deşifreleme uygulamasının yapılacağı kısım. Benzer şekilde deşifreleme için de ayrı bir form olacak. Sonuçlar farklı sayfalar da gösterilecek ve sonuçlar mail atılacak.**



**Şekil 3- Etik Hackleme Araçları uygulamalarının bulunduğu bölüm. Sonuçlar ya pop-up üzerinde ya da farklı bir sayfada gösterilecek.**

## Kod Tasarımı

## Zaman Çizelgesi

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Süre (gün) | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Bölümler | 22.Kas.20 | 23.Kas.20 | 24.Kas.20 | 25.Kas.20 | 26.Kas.20 | 27.Kas.20 | 28.Kas.20 | 29.Kas.20 | 30.Kas.20 | 1.Ara.20 | 2.Ara.20 | 3.Ara.20 | 4.Ara.20 | 5.Ara.20 | 6.Ara.20 | 7.Ara.20 | 8.Ara.20 | 9.Ara.20 | 10.Ara.20 | 11.Ara.20 | 12.Ara.20 | 13.Ara.20 | 14.Ara.20 | 15.Ara.20 | 16.Ara.20 | 17.Ara.20 | 18.Ara.20 | 19.Ara.20 | 20.Ara.20 |
| Grup üyeleri ile iletişim kanallarının kurulması |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Proje konusunun tartışılması |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Proje ile ilgili planlama yapılması |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| İş bölümü yapılması |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Proje tamamlanması test süreci |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Github a yüklenmesi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Projenin tamamlanması, rapor teslimi ve sunum. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# Proje ismi Gerçekleştirme Raporu

## Karşılaşılan Sorunlar ve Uygulanan Çözümler

Projenin gerçekleşme sürecinde bazı kütüphaneleri kullanırken sorun yaşadık. Ama grup içinde iletişime geçerek alternatif kütüphaneler araştırıldı ve onlar uygulandı. Örneğin simetrik şifreleme yöntemi olan AES şifrelemesi için Pycryptodome kütüphanesi içerisindeki AES kütüphanesini kullanmayı denedik. Şifrelemede sorun çıkmadı ama deşifrelemede problem yaşadık. Bu yüzden daha önce de kullanmayı planladığımız Cyrptography kütüphanesindeki Fernet sınıfını kullanarak sorunu çözdük. Henüz daha tamamlanmayan kısımlar var. Grup içinde işbölümü yaparak eksik kısımlar da tamamlanmaya devam etmektedir.

## Proje Bileşenleri ve Görevleri

Projemiz Flask kütüphanesi kullanılarak geliştirilen bir web uygulamasıdır. Web sayfasının tasarımında, hazır template üzerinde değişikler yapıldı ve HTML5, CSS3, Bootstrap, Javascript kullanıldı. Web uygulaması genelinde veritabanına ihtiyaç duyulmadığından sadece Hakkımızda kısmındaki bilgiler veritabanından çekilerek görüntülenmektedir. Veritabanı için SQLite3 kullanıldı. Bunlar dışında hem şifreleme hem de etik hackleme araçları kısmında geliştirilmek istenen uygulamaya uygun kütüphaneler kullanıldı. Veritabanı, şifreleme, etik hackleme araçlarının bileşenleri için ayrı ayrı sınıflar tanımlandı.

## Github Yükleme Süreci

Proje için ortak bir mail hesabı alınarak GitHub’da hesap oluşturuldu. Sonra oluşturulan projeye GitHub masaüstü uygulamasıyla proje dosyaları gönderildi.

# Proje ismi Test Raporu

## Karşılaşılan Sorunlar ve Uygulanan Çözümler

Projenin test edilmesi sürecinde RSA ile deşifrelemede sorun yaşadık. Web uygulaması dışında çalışan fonksiyonu, web arayüzünde formdan aldığı verilerle çalıştıramadık ve hata mesajı aldık. Onun dışında açık olan portları taramak için kullandığımız kısımda tüm portları taramak uzun zaman aldığından sadece çok kullanılan portlardan bir dizi oluşturularak bu dizideki portlarla tarama gerçekleştiriyoruz.

## Test Sürecinde Kullanılan Modüller (Varsa)

## Değerlendirme Kriterleri

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **KRİTERLER** | Ekip Üyeleri | Eğitmen 1 | Eğitmen 2 | Eğitmen 3 | Ortalama |
| 1. Analiz Raporunun Tamamlanması | Emine Aydoğmuş |  |  |  |  |
| 1. Tasarım Raporunun Tamamlanması | Neslihan Genç |  |  |  |  |
| 1. Gerçekleştirim Raporunun Tamamlanması | İsmail Ersin Turan |  |  |  |  |
| 1. Gantt Diagramı | Emine Aydoğmuş |  |  |  |  |
| 1. Arayüz tasarımı | İsmail Ersin Turan |  |  |  |  |
| 1. Veri Tasarımı-Sınıf Tasarımı | Neslihan Genç |  |  |  |  |
| 1. Kullanıcı Yardım Dökümanı | İsmail Ersin Turan |  |  |  |  |
| 1. Programın Çalıştırılması | Neslihan Genç |  |  |  |  |
| 1. Yazılım Test Çalışması | Emine Aydoğmuş |  |  |  |  |