

KTÜ  
MÜHENDİSLİK FAKÜLTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ  
AĞ GÜVENLİĞİ DERSİ  
DÖNEM PROJESİ  
ÖN RAPORU

**Racihe Nisa Başoğlu 402564**

**Emine Sude Aslan 402505**

**Sema Nur Ersoy 440089**

# 1. Proje Amacı

Bu projenin amacı, kullanıcıların yerel ağlarındaki aktif cihazları ve açık portları kolay bir şekilde tespit edebilmesini sağlayan kullanıcı dostu bir grafiksel araç geliştirmektir. Uygulama ayrıca hedef IP adresinin işletim sistemini belirlemeye yönelik temel bir altyapı sunar.

## 2. Uygulama Özellikleri

### a. Cihaz Tarama (IP Taraması)

- Belirli bir alt ağda (192.168.1.0/24 gibi) IP taraması yapar.
- Ping ve belirli TCP portlarına (80, 443, 8080) bağlantı denemesi yaparak cihazın aktif olup olmadığını kontrol eder.
- Yaygın IP'leri (1, 100, 101, 254) önce hızlıca tarar.
- Taramayı paralel (çoklu iş parçacığı) olarak yapar.
- Kullanıcıya tarama ilerlemesini Progressbar ile gösterir.
- Sonuçları arayüzde liste halinde verir.

#### a.1. Cihaz Tarama Fonksiyonlarının açıklanması (network\_scanner.py)

**yerel\_agdaki\_ipler:** cihaz\_tara\_paralel fonksiyonu kullanılır. Bu fonksiyon TCP port ve ping kontrolü ile cihaz varlığını kontrol eder.

**total\_ips** değişkeninde toplam taranacak ip sayısı hesaplanır. Her ip tarandığında ilerleme yüzdesi kaydedilir. Bu ilerleme arayüzde gösterim için kullanılır.

### b. Port Tarama

- Belirli bir IP adresi ve port aralığı alarak, o IP'deki açık portları kontrol eder Bilinen portlara karşılık gelen servis isimleri (HTTP, FTP, SSH vb.) eşleştirme ile gösterilir.

#### a.2. Port Taraması Fonksiyonlarının açıklanması (port\_scanner.py)

**servis\_banner(hedef\_ip, port, timeout):** Belirtilen portta çalışan servisin karşılama bilgisini okur. Özel portlar için özel komutlar gönderir.

**port\_tara(hedef\_ip, port, servisler):** Verilen ipye göre taranan portların json dosyasında eşleştiği servisleri belirler.

**Port\_Taramasi():** Her port için bir thread oluşturulur ve port\_tara fonksiyonu çağrılır. Port açıksa servis\_banner\_oku fonksiyonu ile banner bilgisi okunur.

### c. İşletim Sistemi Tespiti

- Bu çalışmanın amacı, hedef IP adresindeki yaygın servis portlarına bağlantı kurarak servislerin açılış mesajlarında (banner) yer alan bilgileri toplamaktır. Bu bilgiler, hedef sistemin çalıştığı işletim sistemi, kullanılan servis sürümleri ve güvenlik açıklarına dair ipuçları sağlayabilir.

#### a.3.İşletim Sistemi Tespit Fonksiyonlarının açıklanması (os\_detector.py)

**os\_tespit:** Ip adresi girilen cihaza ping atılarak cihazın Time to Live (ttl) süresine bakılır. Kartar yapısı ile tahmin edilir.

**ttl\_al(ip):** İşletim sistemine göre farklı ping komutları kullanılarak ttl değeri döndürülür.

**os\_tahmin(ttl):** Dönen ttl değerine göre kara bloğu ile işletim sistemi belirlenir.

#### Çalışma Prensibi:

- Kullanıcıdan bir IP adresi alınır.
- Sık kullanılan bazı portlar (örn. 21, 22, 80, 443) üzerinden TCP bağlantısı kurulur.
- Bağlantı kurulan servis, genellikle bağlantı sırasında otomatik olarak bir banner (giriş mesajı) döner.
- Bu banner mesajı ekrana yazdırılır



