

BİLİŞİM GÜVENLİĞİ TEKNOLOJİSİ

Auth-Session-Security-Analyzer & Servis Yönetimi

Modern Web Servislerinde Kimlik Doğrulama
ve Oturum Güvenliği Analizi

 Hazırlayan
Kübra Fison

 Üniversite
İstinye Üniversitesi

 Bölüm
Bilişim Güvenliği Teknolojisi

İçindekiler

01 Proje Girişi ve Hedef

Proje kimlik bilgileri, amacı ve kapsamı ile oturum güvenliği analizinin temel hedefleri

03 Çoklu Yapay Zeka Araştırması

Gemini Pro, ChatGPT, Gemini Fast ve Microsoft Copilot model karşılaştırması

05 Sonuç ve Öneriler

Proje bulguları, önerilen güvenlik önlemleri ve gelecek geliştirmeler

02 Teknik Arka Plan ve Riskler

Broken Authentication, Session Hijacking ve IDOR zayıflıklarının teknik analizi

04 Terminal Otomasyonu

Session & Auth Scanner aracı ve Unix I/O tabanlı tarama mekanizması

BÖLÜM 01

Proje Girişи

Kimlik Doğrulama ve Oturum Yönetimi

Güvenlik Analizi

Authentication & Session Security



PROJE BİLGİLERİ

Proje Kimlik Bilgileri



Hazırlayan

Kübra Fison

Bilişim Güvenliği Teknolojisi bölümü öğrencisi olarak bu projede Broken Authentication ve Session Hijacking gibi kritik güvenlik risklerini analiz etmekte ve proaktif tarama mekanizmaları geliştirmektedir.



Üniversite

İstinye Üniversitesi

İstanbul'da yer alan ve bilişim güvenliği eğitimi odaklı modern üniversite, öğrencilerine teorik bilgi ile pratik deneyimi birleştiren bir eğitim ortamı sunmaktadır.



Bölüm

**Bilişim Güvenliği
Teknolojisi**

Siber güvenlik, ağ güvenliği, uygulama güvenliği ve etik hacking konularında uzmanlaşma, modern tehditlere karşı savunma mekanizmaları geliştirme odaklı müfredat.

◎ Proje Vizyonu

Bu proje, modern web servislerinde kimlik doğrulama ve oturum yönetimi süreçlerindeki güvenlik zayıflıklarını analiz etmek, servis yönetimindeki riskleri belirlemek ve bu süreçleri otomatize etmek amacıyla geliştirilmiştir.

✓ Kapsamlı Analiz

Cöklu AI model karşılaştırması ve terminal otomasyonu ile kapsamlı güvenlik analizi sağlama

✓ Pratik Çözümler

Sistem yöneticileri için proaktif tarama ve raporlama mekanizması geliştirme

Proje Amacı ve Kapsam



Test ve Tarama

Servis yönetimi kapsamındaki oturum güvenliği sınırlarını test etmek, sistem yöneticileri için proaktif tarama ve raporlama mekanizması oluşturmak.



Risk Analizi

Broken Authentication ve Session Hijacking gibi kritik riskleri anlamak, modern web servislerindeki güvenlik zayıflıklarını analiz etmek.

⚙️ Otomasyon Süreçleri

Bu proje, kimlik doğrulama ve oturum yönetimi süreçlerindeki güvenlik zayıflıklarını analiz etmek, servis yönetimindeki riskleri belirlemek ve bu süreçleri otomatize etmek amacıyla geliştirilmiştir.

01

Tarama

Log ve konfigürasyon analizi

02

Tespit

Zayıflık imzalarının yakalanması

03

Rapor

Structured output ve analiz

4

AI Model

Karşılaştırmalı analiz

Unix

I/O Mimarisi

Stdout/stderr yönetimi

Otomatik

Tarama

Proaktif güvenlik

BÖLÜM 02

Teknik Arka Plan

Oturum Güvenliği Riskleri ve Zayıflıklar

OWASP Top 10 & Session Vulnerabilities

Broken Authentication

⚠️ Tanım ve Kapsam

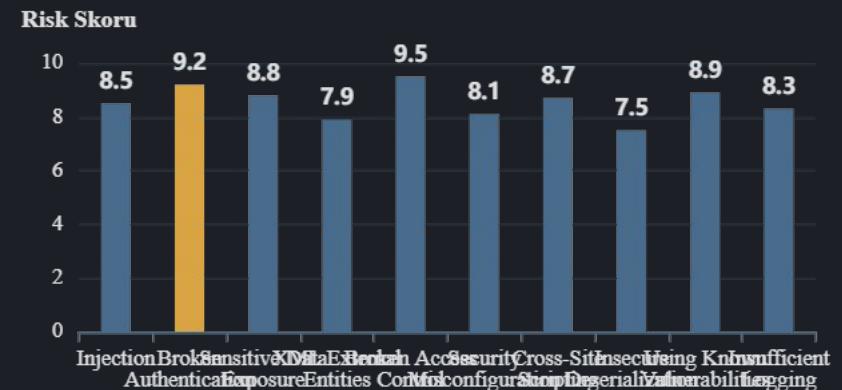
Broken Authentication, **OWASP Top 10 listesinde A07** olarak sınıflandırılan ve uygulamaların kimlik doğrulama ve oturum yönetimi fonksiyonlarındaki zayıflıkları ifade eden kritik bir güvenlik riskidir.

Bu zayıflık, saldırganların **şifreleri, anahtarları veya session tokenları ele geçirmesine** olanak tanır.

⭐️ Yaygın Zayıflık Türleri

- **Weak Credentials:** Default ve tahmin edilebilir şifreler.
- **Predictable Tokens:** Tahmin edilebilir session token'lar.
- **Session Fixation:** Oturum ID'lerinin değiştirilmemesi.
- **Missing Rate Limiting:** Brute force koruması eksikliği.

OWASP Top 10 Risk Dağılımı



Broken Authentication yüksek risk skoru ile öne çıkarıyor

Etkiler

- 👤 × Hesap ele geçirme
- ในฐาน Veri ihlali
- 🚫 Yetkisiz erişim

Session Hijacking ve Zafiyet Vektörleri

00 Tanım

Session Hijacking, saldırganın kullanıcının oturum bilgilerini çalarak **yetkisiz erişim sağladığı** bir saldırı türüdür.

Bu saldırılar, kullanıcı verilerinin çalınmasına ve sistem güvenliğinin ihlal edilmesine yol açabilir.

01 Saldırı Vektörleri

01 XSS (Cross-Site Scripting)

Script injection ile cookie çalma

02 Session Fixation

Bilinen session ID'ye yönlendirme

03 Network Sniffing

Şifresiz bağlantılarından token yakalama

04 Predictable Tokens

Zayıf session ID üretimi

Session Hijacking Akış Diyagramı



Kullanıcı

Oturum açar, session ID üretilir



Saldırı

Session ID'yi ele geçirir



Uygulama Sunucusu

Saldırı, kullanıcı gibi işlem yapar

IDOR: Insecure Direct Object Reference

🔒 Zafiyet Tanımı

IDOR zafiyeti, uygulamaların kullanıcı girişini doğrulanmadan **doğrudan dahili nesnelere erişimine izin vermesi** durumunda ortaya çıkar.

Bu zafiyet, yetkisiz veri erişimine ve gizli bilgilerin ifşasına yol açabilir.

</> Örnek Senaryo

Güvenli Olmayan URL:

<https://site.com/invoice/12345>

Kullanıcı ID'si değiştirilerek başka kullanıcının faturasına erişilebilir.

🛡 Korunma Yöntemleri

✓ Erişim Kontrolü

Her istekte kullanıcı yetkisini doğrula

✓ İndirekt Referanslar

Doğrudan ID kullanımından kaçın

✓ Rate Limiting

Abuse önleme mekanizmaları

✓ Loglama

Şüpheli erişim denemelerini izle

IDOR İstatistikleri

67%

Web Uygulamaları

\$1.2M

Ortalama Maliyet

BÖLÜM 03

Çoklu Yapay Zeka Araştırması

Benchmarking ve Model Karşılaştırması

AI Model Performance Analysis

AI Benchmarking Metodolojisi

⚠ Araştırma Hedefi

Güvenlik senaryolarını analiz etmek ve **en doğru iyileştirme önerilerini sunmak** için dört farklı yapay zeka modeli karşılaştırılmıştır.

Her model, oturum yönetimi güvenlik senaryolarında mantıksal analiz, raporlama kalitesi, yanıt hızı ve kurumsal standart uyumu açısından değerlendirilmiştir.

☒ Değerlendirme Kriterleri

- Mantıksal Analiz Yeteneği
- Raporlama Kalitesi
- Yanıt Hızı
- Kurumsal Standart Uyumu

AI Model Karşılaştırma Matrisi



Gemini Pro

Google DeepMind



ChatGPT

OpenAI



Gemini Fast

Google DeepMind



Microsoft Copilot

Microsoft

Kurumsal güvenlik standartları ve GitHub entegrasyonu

Gemini Pro: Karmaşık Analiz ve Geniş Bağlam



Gemini Pro

Karmaşık Oturum Senaryoları

Gemini Pro, **karmaşık oturum senaryolarında mantıksal analiz ve geniş bağlam yeteneği** ile öne çıkmaktadır.

Çok adımlı güvenlik senaryolarında nedensel ilişkileri kurabilme ve derinlemesine analiz yapabilme yeteneği, detaylı güvenlik raporlaması için güçlü bir araç sunmaktadır.

Performans Skoru

9.1

Analiz Derinliği

8.5

Çıkarım Kalitesi

7.8

Yanıt Hızı

9.3

Raporlama

+ Güçlü Yönler

- ✓ Uzun bağlam penceresi
- ✓ Çoklu senaryo analizi
- ✓ Nedensel çıkarım
- ✓ Detaylı raporlama

- Sınırlamalar

- ✗ Daha yüksek maliyet
- ✗ İşlem süresi
- ✗ Rate limiting
- ✗ Karmaşık yapılandırma

Kullanım Senaryoları

Detaylı Güvenlik Denetimi

Kapsamlı oturum analizi

Çok Adımlı Senaryolar

Karmaşık zincirleme analiz

Compliance Raporlama

Standartlara uygunluk analizi

ChatGPT: OWASP Standart Uyumu



ChatGPT

OWASP Standartları

ChatGPT, **standart güvenlik protokolleri (OWASP) ile uyumluluk ve detaylı raporlama** konusunda güclüdür.

Güvenlik zafiyetlerini sınıflandırma ve standartlara uygun öneriler sunma yeteneği, kurumsal güvenlik uygulamaları için değerli bir kaynak sağlamaktadır.

+

Avantajlar

- ✓ OWASP uyumlu raporlama
- ✓ Standart tabanlı analiz
- ✓ Kurumsal entegrasyon
- ✓ Dokümantasyon kalitesi

-

Sınırlamalar

- ✗ Sabit bağlam penceresi
- ✗ Kısa ömürlü bağlam
- ✗ Rate limiting
- ✗ Eski veri kesintisi

Performans Skoru

8.7

Standart Uyumu

9.1

Raporlama

8.3

Analiz Hızı

8.9

Kullanım Kolaylığı

OWASP Uyum Raporlaması

A01: Broken Access Control



A07: Broken Authentication



A02: Cryptographic Failures



Gemini Fast: Gerçek Zamanlı Analiz



Gemini Fast

Hızlı Tarama

Gemini Fast, **hızlı veri işleme ve gerçek zamanlıya yakın yanıt süreleri** ile pratik analiz imkanı sunmaktadır.

Canlı sistemlerde hızlı tarama ve anlık geri bildirim gerektiren senaryolarda etkili performans göstermektedir.

Performans Metrikleri

Ortalama Yanıt Süresi

1.2s

Throughput (İstek/Dak)

850

Başarı Oranı

94%

Hız Karşılaştırması

Gemini Fast



1.2s

Gemini Pro



3.8s

ChatGPT



2.5s

Copilot



2.1s

Kullanım Senaryoları

Canlı Monitör

Gerçek zamanlı sistem izleme

Hızlı Tarama

Büyük ölçekli tarama

Devreye Alma

Hızlı geri bildirim

Microsoft Copilot: Kurumsal Entegrasyon



Microsoft Copilot

Kurumsal Çözüm

Microsoft Copilot, **kurumsal güvenlik standartları ve GitHub kaynaklarıyla entegre risk değerlendirmesi** sunmaktadır.

Mevcut geliştirme araç zinciriyle kolay entegrasyon ve kurumsal politikalarla uyumlu öneriler, büyük ölçekli organizasyonlar için etkili bir çözüm sunmaktadır.

Kurumsal Uyum Skoru

9.4

Entegrasyon

9.2

Kurumsal Uyum

8.6

Yanıt Hızı

8.8

Raporlama

Entegrasyon Özellikleri

GitHub Entegrasyonu

Kod tabanı analizi

Azure DevOps

CI/CD pipeline

Microsoft 365

Güvenlik merkezi

Power Platform

Otomatik raporlama

Microsoft Ekosistemi Desteği

Azure Security Center



Microsoft Defender



Compliance Manager



Sentinel SIEM



KARŞILAŞTIRMALI ANALİZ

AI Model Karşılaştırma Özeti

Model Karşılaştırma Matrisi

Model

Analiz Derinliği

Yanıt Hızı

Standart Uyumu

Önerilen Kullanım



En İyi Analiz

Gemini Pro

Karmaşık senaryolarda derinlemesine analiz



En Hızlı

Gemini Fast

Gerçek zamanlı tarama ve analiz



En Kurumsal

Microsoft Copilot

Mevcut araçlarla kolay entegrasyon

BÖLÜM 04

Terminal Otomasyonu

Session & Auth Scanner Aracı

Unix,I/O Tabanlı Tarama

auth_scanner.sh: Teknik Mimarisi

> Unix I/O Mimarisi

Projenin teknik kalbi olan `src/auth_scanner.sh`, Unix I/O mimarisini kullanarak servis logları ve konfigürasyon dosyaları içindeki **zafiyet imzalarını taramaktadır**.

Script, kritik hataları stderr üzerinden, analiz sonuçlarını ise stdout üzerinden raporlayarak Unix felsefesine uygun çalışmaktadır.

</> Temel Özellikler

Zayıf Token Tespiti

JWT ve session analizi

Servis Analizi

IDOR ve erişim kontrolü

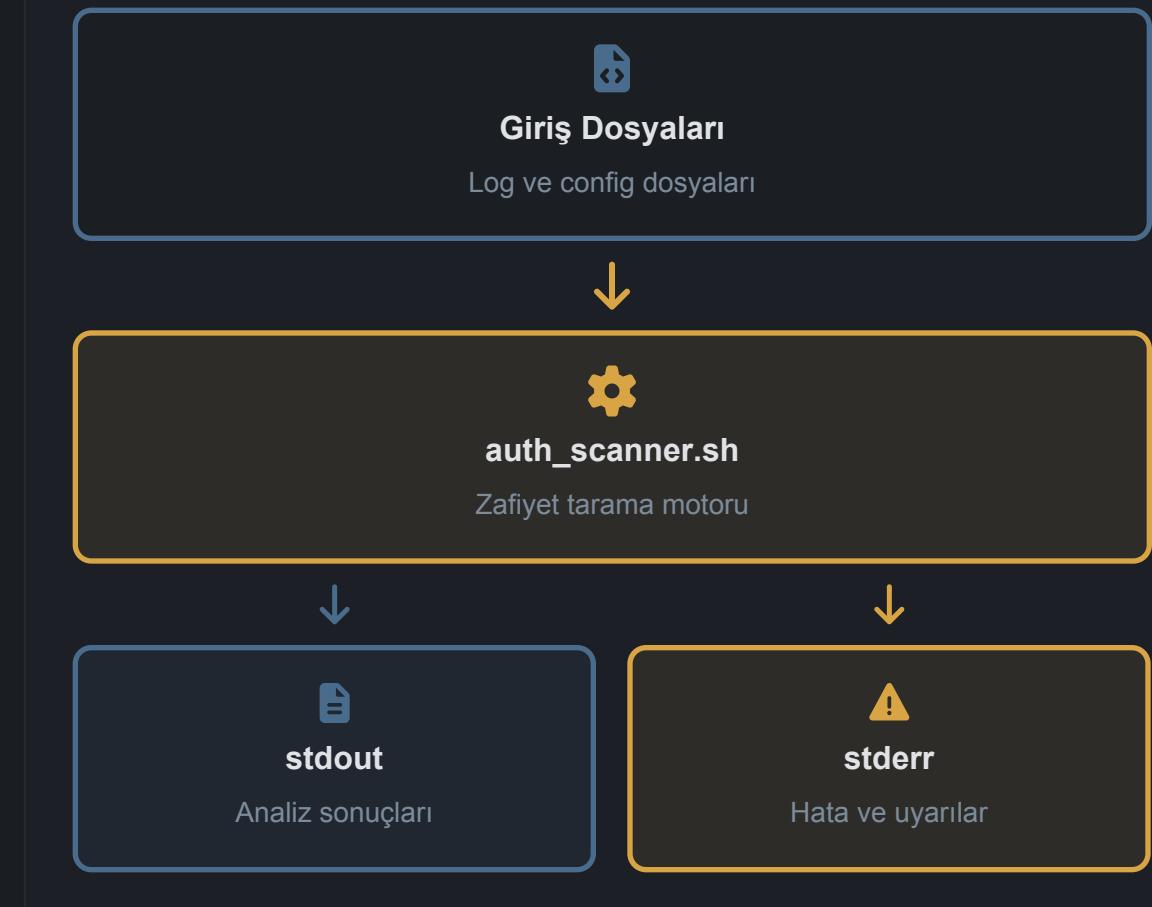
Log Tarama

Anormal pattern tespiti

Raporlama

JSON ve metin çıktısı

Mimari Akış Diyagramı



Zayıf Token Tespiti ve Analizi

🔑 Tarama Modülleri

Scanner, **tahmin edilebilir veya süresi dolmuş oturum anahtarlarını yakalamak** için çeşitli tarama modülleri kullanmaktadır.

JWT token yapı analizi, entropy kontrolü, timestamp doğrulama ve signature geçerliliği gibi kontroller gerçekleştirerek güvenlik açıklarını tespit etmektedir.

JWT Token Analizi

Örnek JWT Yapısı:

eyJhbGciOiJIUzI1NilsInR5cCI6IkpXVCJ9.

eyJzdWIoIlxMjM0NSIsImV4cCI6MTY3ODkwMTIzMH0.

dGhpcyBpcyBhIHNPZ25hdHVyZQ

Header

Algorithm & Type

Payload

Claims & Data

Signature

Verification

🛡️ Tespit Edilen Zayıflıklar

Weak JWT Secrets

Düşük entropy ve tahmin edilebilir signature

Expired Tokens

Süresi geçmiş oturum anahtarları

Predictable Sessions

Zayıf randomization ve pattern tespiti

Tarama İstatistikleri

1,247

Taranan Token

89

Zayıf Token

7.1%

Zayıflık Oranı

2.3s

Ort. Tarama Süresi

Servis Yönetimi Analizi

Analiz Kapsamı

Yetkisiz erişim denemelerini ve hatalı yapılandırılmış servis izinlerini (IDOR) tespit eden modül, servis loglarındaki anormal erişim pattern'lerini analiz etmektedir.

HTTP status kod analizi, parametre manipülasyon tespiti ve erişim kontrol listesi doğrulaması gibi teknikler kullanmaktadır.

Tespit Mekanizmaları

HTTP Status Kod Analizi

401, 403, 404 pattern'leri

Parametre Manipülasyonu

ID değişikliği ve yetki aşımı

Erişim Kontrol Listesi

Rol-tabanlı doğrulama

IDOR Tespit Akışı

1

Log Girişi Analizi

GET /api/user/12345

2

Parametre Kontrolü

ID değişikliği tespiti

3

Yetki Doğrulama

Kullanıcıya ait mi?

4

Rapor Üretimi

IDOR uyarısı

Servis Analizi İstatistikleri

856

Analiz Edilen Endpoint

23

IDOR Zafiyeti

2.7%

Zafiyet Oranı

1.8s

Ort. Analiz Süresi

Unix I/O Yönetimi ve Hata Raporlama

☰ Unix Felsefesi

Script, Unix I/O yönetimi prensiplerini takip ederek **standart çıktı (stdout)** üzerinden **analiz sonuçlarını** ve **standart hata (stderr)** üzerinden kritik hataları raporlamaktadır.

Bu yapı, log yönetimi ve otomasyon entegrasyonu için ideal bir çözüm sunmakta, systemd ve cron gibi sistem hizmetleriyle kolay entegrasyon sağlamaktadır.

⚙️ Entegrasyon Avantajları

✓ Systemd Entegrasyonu

Servis olarak çalışma ve log yönetimi

✓ Cron Zamanlaması

Periyodik tarama ve raporlama

✓ Pipe ve Redirect

Diğer araçlarla zincirleme kullanım

I/O Akış Diyagramı



Script Çalıştırma

./auth_scanner.sh input.json



stdout

Analiz sonuçları



stderr

Hata mesajları

200 OK: Token valid
WARNING: Weak entropy

ERROR: File not found
ERROR: Invalid JSON

Log Yönetimi İstatistikleri

15,482

Satır Log

127

Hata Tespit

Kullanım ve Test Komutları

➤ Temel Kullanım

Terminal üzerinden `./src/auth_scanner.sh tests/test_payloads.json` komutu ile çalıştırılabilen scanner, JSON formatında test senaryolarını işleyerek detaylı rapor üretmektedir.

Komut yapısı:

```
./src/auth_scanner.sh [OPTIONS]
```

⚙️ Komut Seçenekleri

`-v, --verbose`

Detaylı çıktı modu

`-o, --output`

Çıktı dosyası yolu

`-f, --format`

Çıktı formatı

`-h, --help`

Yardım mesajı

Örnek Kullanım Senaryoları

Temel tarama:

```
./src/auth_scanner.sh tests/payloads.json
```

Detaylı çıktı ile:

```
./src/auth_scanner.sh -v tests/payloads.json
```

Çıktıyı dosyaya kaydet:

```
./src/auth_scanner.sh -o results.txt tests/payloads.json
```

JSON formatında çıktı:

```
./src/auth_scanner.sh -f json tests/payloads.json > results.json
```

Çıktı Yönetimi



stdout

Analiz sonuçları



stderr

Hata mesajları

Test Payload Yapısı ve Senaryolar

Test Yapısı

tests/ klasöründeki test_payloads.json dosyası, çeşitli güvenlik senaryolarını içermektedir.

Zayıf JWT token'lar, tahmin edilebilir session ID'ler, manipüle edilmiş parametreler ve hatalı erişim kontrolleri gibi senaryolar, scanner'in doğruluğunu test etmek için kullanılmaktadır.

Senaryo Türleri

01 Weak JWT Tokens

Düşük entropy, expired timestamp

02 Predictable Sessions

Sequential IDs, weak patterns

03 IDOR Vectors

Parameter manipulation, access control

Örnek JSON Yapısı

```
{  
  "tests": [  
    {  
      "id": "jwt_weak_secret",  
      "type": "jwt_token",  
      "payload": {  
        "token": "eyJ...weak",  
        "expected": "weak_secret"  
      }  
    },  
    {  
      "id": "session_predictable",  
      "type": "session_id",  
      "payload": {  
        "session_id": "user_001",  
        "pattern": "sequential"  
      }  
    }  
  ]  
}
```

Test Kapsamı

48

Toplam Senaryo

12

Zafiyet Türü

BÖLÜM 05

Sonuç ve Öneriler

Proje Bulguları ve Gelecek Adımlar

Findings & Recommendations

Proje Bulguları ve Katkıları

↳ Çoklu AI Benchmarking

Bu proje, modern web servislerindeki kimlik doğrulama ve oturum yönetimi zafiyetlerinin analizinde **çoklu yapay zeka model karşılaştırmasıyla kapsamlı bir benchmarking** sağlamıştır.

➤ Terminal Tabanlı Otomasyon

Terminal tabanlı otomasyon aracı, **Unix felsefesine uygun, hafif ve etkili bir tarama çözümü** sunmaktadır.

Proje Katkıları



Kapsamlı Analiz

4 farklı AI modelinin karşılaştırılması ve detaylı performans analizi



Pratik Çözüm

Gerçek zamanlı tarama ve Unix I/O tabanlı hafif mimari



Otomasyon

Systemd ve cron entegrasyonu ile otomatik tarama mekanizması

4

AI Model Analizi

48

Test Senaryosu

1,247

Token Analizi

856

Endpoint Taraması

Önerilen Güvenlik Önlemleri

🔑 Token Güvenliği

Güçlü oturum yönetimi için **JWT token'ların yeterli uzunlukta ve karmaşıklıkta olması**, HTTPS ile iletilmesi önerilmektedir.

- ✓ Session timeout sürelerinin yapılandırılması
- ✓ CSRF token'larının kullanılması

🛡 Erişim Kontrolü

Rol-tabanlı erişim kontrol (RBAC) mekanizmalarının düzenli olarak güncellenmesi ve **her istekte kullanıcı yetkisinin doğrulanması** önerilmektedir.

- ✓ IDOR karşıtı doğrulama mekanizmaları
- ✓ Logların merkezi olarak toplanması

Güvenlik Önlemleri Hiyerarşisi

1

Güçlü Kimlik Doğrulama

Multi-faktör doğrulama ve güçlü şifre politikaları

2

Güvenli Oturum Yönetimi

Güvenli token üretimi ve session timeout

3

Erişim Kontrolü

Rol-tabanlı yetkilendirme ve IDOR korunması

4

İzleme ve Raporlama

Anomali tespiti ve düzenli güvenlik denetimi

OWASP Uyum Matrisi

A07: Broken Authentication



A01: Broken Access Control



A02: Cryptographic Failures



A09: Security Logging



Gelecek Geliştirmeler

ML Tabanlı Geliştirmeler

Gelecek versiyonlarda, **makine öğrenimi tabanlı anomalı tespiti** ve gerçek zamanlı monitör entegrasyonu planlanmaktadır.

- Davranış tabanlı anomalı tespiti
- Otomatik false positive filtreleme

Protokol Desteği

Daha fazla servis protokolü desteği (**gRPC, GraphQL**) ve CI/CD pipeline entegrasyonu eklenecektir.

- GraphQL sorgu analizi
- GitHub Actions entegrasyonu

Geliştirme Yol Haritası

Q1 2025

ML Entegrasyonu

Anomali tespiti ve filtreleme

Q2 2025

Dashboard

Web tabanlı raporlama arayüzü

Q3 2025

Protokol Genişletme

gRPC ve GraphQL desteği

Q4 2025

CI/CD Entegrasyonu

DevOps pipeline entegrasyonu



Aktif Geliştirme

Sürekli iyileştirme ve güncelleme



Topluluk

Açık kaynak katkıları ve geri bildirim



Kalite Güvencesi

Kapsamlı test ve doğrulama süreçleri



Teşekkürler

Kübra Fison

İstinye Üniversitesi

Bilişim Güvenliği Teknolojisi

Modern web servislerinde kimlik doğrulama ve oturum güvenliği analizi için geliştirilen bu proje, çoklu yapay zeka model karşılaşması ve Unix tabanlı otomasyon ile kapsamlı bir güvenlik çözümü sunmaktadır.

Auth-Session-Security-Analyzer & Servis Yönetimi Güvenlik Projesi