PENETRATION TESTING FRAMEWORK

LAM-Action Guard

Sızma Testi Otomasyon Platformu

Penetration Testing Automation Framework

Geliştirici
Aziz Efe Çırak

Versiyon
v1.0.0

Kategori
Penetration Testing

AGENDA

İçindekiler

PROJECT OVERVIEW

Proje Künyesi ve Tanıtım

Hazırlayan

Aziz Efe Çırak

Cybersecurity Researcher & Developer

Versiyon

v1.0.0

Initial Release

Kategori

Penetration Testing

Automation Framework

Proje Misyonu

LAM-Action-Guard, kapsamlı bir **Sızma Testi Otomasyon Platformudur** (Pentest Automation Framework). Güvenlik test süreçlerini hızlandırmak, standartlaştırmak ve manuel hata payını azaltmak için tasarlanmıştır.

> **"Güvenlik, bir ürün değil, bir süreçtir."**
>
> Security is a process, not a product.

Hız

Test süreçlerini optimize eder

Standart

Kaliteyi ve tutarlılığı sağlar

Hata Azaltma

İnsan hatasını minimize eder

CORE CAPABILITIES

Temel Özellikler ve Yetenekler

Modern sızma testi ihtiyaçlarını karşılamak için tasarlanmış kapsamlı özellikler

Güvenlik Açığı Tarama

Vulnerability Assessment

Kritik zafiyetlerin otomatik tespiti için gelişmiş tarama mekanizmaları. Yaygın güvenlik açıklarını yüksek doğrulukla tespit eder.

XSS | SQLi | LFI | RCE

Keşif Otomasyonu

Recon Automation

Hedef sistem hakkında otomatik bilgi toplama. Subdomain keşfi, port tarama ve servis tanımlama işlemlerini entegre eder.

- ✓ Subdomain enumeration
- ✓ Port ve servis tarama
- ✓ Teknoloji tespiti

Yüksek Performans

High Performance

Go dilinin eşzamanlılık (concurrency) özelliği ile ultra hızlı port tarayıcı. 65.535 portu saniyeler içinde tarar.

Port Tarama Hızı	~2dk
65.535 port için	

Çoklu Dil Desteği

Multi-Language Support

Bash, Python ve Go'nun güçlü yönlerinin hibrit kullanımı. Her modül için en uygun dili seçer.

Bash	Python	Go
Recon	Scanner	Port Scan

JSON Raporlama

JSON-Based Reporting

İşlenebilir, makine dostu rapor formatı. CI/CD entegrasyonu ve otomasyon için optimize edilmiştir.

- ✓ Makine tarafından okunabilir

Self-Check

System Health Check

Sistemin kendi sağlığını kontrol etme mekanizması. Bağımlılık kontrolü ve ortam doğrulaması.

- ✓ Otomatik sistem kontrolü

SYSTEM ARCHITECTURE

Modüler Mimari Yapı

Üç temel modülden oluşan, esnek ve ölçeklenebilir sistem mimarisi

01 recon_automation.sh

Bash Script

İlk keşif ve bilgi toplama işlemlerini yönetir. Hedef sistem hakkında kapsamlı bilgi toplar.

- Subdomain keşfi
- DNS sorgulama
- Teknoloji tespiti

Bash v4.0+

02 vuln_scanner.py

Python Script

Detaylı güvenlik açığı analizlerini gerçekleştirir. Esnek ve genişletilebilir yapıda tasarlanmıştır.

- XSS, SQLi taraması
- LFI, RCE kontrolü
- Çoklu thread desteği

Python v3.10+

03 port_scanner.go

Go Program

Eşzamanlılık (concurrency) özelliği ile çok hızlı port taraması yapar.

- Goroutines desteği
- Ultra düşük latency
- Yüksek performans

Go v1.19+

Modüler Yapının Avantajları

Esneklik

Modüller bağımsız çalışabilir veya kombine edilebilir

Bakım

Her modül ayrı güncellenebilir ve geliştirilebilir

Ölçeklenebilirlik

Yeni modüller kolayca eklenebilir

Performans

Her görev için en uygun dil ve araç kullanılır

INSTALLATION & SETUP

Teknik Gereksinimler ve Kurulum

Platformu çalıştırmak için gereken minimum ortam ve kurulum adımları

Sistem Gereksinimleri

İşletim Sistemi

Windows	Linux	macOS
10/11	Ubuntu/Debian	10.15+

Donanım Önerileri

Minimum RAM	4 GB	Önerilen RAM	8 GB+
İşlemci	2+ Çekirdek	Depolama	1 GB+

Yazılım Gereksinimleri

Python — v3.10+

Vulnerability Scanner

- ✓ requests, beautifulsoup4, colorama

Go — v1.19+

Port Scanner

- ✓ net, sync, time (standart kütüphane)

Bash — v4.0+

Recon Scripts

- ✓ dig, nmap, curl, whois

Kurulum Adımları

1. **Depoyu Klonlayın**
   ```
   git clone [repo-url]
   ```
2. **Bağımlılıkları Yükleyin**
   ```
   pip install -r requirements.txt
   ```
3. **Çalıştırın**
   ```
   python main.py --help
   ```

PERFORMANCE METRICS

Performans ve Ölçeklenebilirlik

Go, Python ve Bash'in güçlü kombinasyonu ile yüksek performanslı otomasyon

Port Tarama

~2dk

65.535 port için toplam süre

Go'nun goroutines özelliği ile eşzamanlı port taraması, geleneksel yöntemlere göre 10x daha hızlı performans.

Eşzamanlılık	1000+ goroutines
Timeout	1 saniye

Vulnerability Tarama

95%+

Tespit doğruluğu oranı

Python'ın çoklu thread desteği ile aynı anda birden fazla hedefde güvenlik açığı taraması. ZAP ve Nuclei ile karşılaştırılabilir performans.

Thread Sayısı	10-50
False Positive	%5

Kaynak Kullanımı

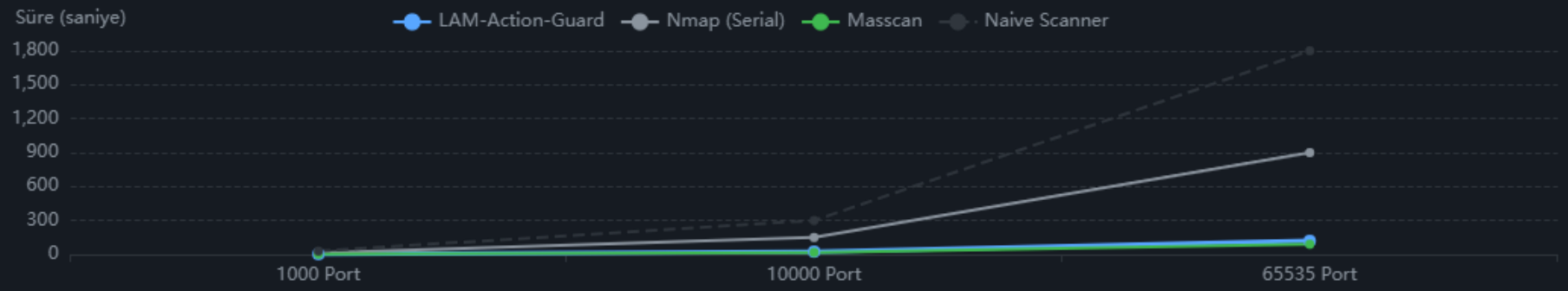
<500MB

Ortalama RAM tüketimi

Bash scriptlerinin düşük kaynak tüketimi ile hafif ve hızlı keşif işlemleri. Arka planda sürekli çalışmaya uygun.

CPU Kullanımı	%10-20
Disk Alanı	<100MB

Karşılaştırmalı Performans Analizi



VULNERABILITY DETECTION

Desteklenen Güvenlik Vektörleri

Modern web uygulamalarında en yaygın kritik zafiyetlerin tespiti

XSS

Cross-Site Scripting

Otomatik payload enjeksiyonu ve tespit mekanizmaları ile reflected, stored ve DOM tabanlı XSS zafiyetlerini tespit eder. Filtre aşımı teknikleri ve WAF bypass yöntemleri uygular.

CVSS Skoru 6.1 - 8.8

SQL Injection

SQLi

Blind ve error-based tarama teknikleri ile veritabanı güvenlik açıklarını tespit eder. Union-based, boolean-based ve time-based SQL enjeksiyonlarını otomatik olarak test eder.

CVSS Skoru 8.8 - 9.8

LFI

Local File Inclusion

Dosya erişim kontrolleri ve directory traversal zafiyetlerini tespit eder. /etc/passwd, log dosyaları ve hassas sistem dosyalarına yetkisiz erişim denemelerini kontrol eder.

CVSS Skoru 7.5 - 9.1

RCE

Remote Code Execution

Uzaktan kod çalıştırma zafiyetlerini tespit eder. Komut enjeksiyonu, deserialization ve file upload güvenlik açıklarını otomatik olarak test eder.

CVSS Skoru 9.8 - 10.0

OUTPUT FORMAT

Raporlama ve Çıktı Formatları

JSON tabanlı, işlenebilir ve makine dostu raporlama sistemi

JSON Rapor Yapısı

Makine tarafından kolayca işlenebilen, standart JSON formatında raporlar. CI/CD entegrasyonu ve otomasyon için optimize edilmiştir.

```
{
  "scan_id": "scan_001",
  "timestamp": "2025-01-20T10:30:00Z",
  "target": "example.com",
  "vulnerabilities": [{
    "type": "xss",
    "severity": "high",
    "url": "https://example.com/search",
    "parameter": "q",
    "cvss_score": 8.8,
    "description": "Reflected XSS found",
    "evidence": "payload: "
  }],
  "summary": {
    "total_found": 5,
    "critical": 1,
    "high": 2,
    "medium": 2
  }
}
```

Özellikler ve Avantajlar

Makine Dostu

JSON formatı, raporların programatik olarak işlenmesini ve diğer araçlara entegrasyonunu kolaylaştırır.

Standart Uyumlu

CVSS skorlama sistemi, CVE referansları ve OWASP kategorizasyonu ile endüstri standartlarına uygun.

Detaylı Kanıt

Her zafiyet için HTTP istek/yanıt örnekleri, payload bilgileri ve ekran görüntüleri gibi detaylı kanıtlar.

Özelleştirilebilir

Çıktı formatı, filtreleme seçenekleri ve rapor şablonları kolayca özelleştirilebilir.

Entegrasyon Senaryoları

- CI/CD Pipeline
- Dashboard
- Alerting
- SIEM Integration

ROADMAP

Gelecek Planları ve Yol Haritası

Platformun gelecek gelişim planları ve yenilikçi özellikler

WEB Arayüzü

Dashboard Integration

Modern, responsive web arayüzü ile kullanıcı dostu dashboard. Gerçek zamanlı izleme ve yönetim.

- ✓ Responsive tasarım
- ✓ Gerçek zamanlı izleme
- ✓ Rol bazlı erişim
- ✓ Etkileşimli raporlar

Hedef: Q2 2025

CVE Desteği

Vulnerability Database

NVD API entegrasyonu ile otomatik CVE güncelleme. Zafiyet veritabanı yönetimi ve korelasyon.

- ✓ NVD API entegrasyonu
- ✓ Otomatik güncelleme
- ✓ CVSS skorlama
- ✓ Exploit entegrasyonu

Hedef: Q3 2025

Yapay Zeka

AI Anomaly Detection

Isolation Forest ML algoritması ile davranış analizi. False positive azaltma ve akıllı tarama.

- ✓ Isolation Forest
- ✓ Davranış analizi
- ✓ False positive azaltma
- ✓ Otomatik öğrenme

Hedef: Q4 2025

FREQUENTLY ASKED QUESTIONS

Sık Sorulan Sorular

Teknik sorular ve platform hakkında merak edilenler

Hangi işletim sistemlerinde çalışır?

LAM-Action-Guard, Windows 10/11, Linux (Ubuntu, Debian, CentOS) ve macOS 10.15+ üzerinde test edilmiştir. Python, Go ve Bash'in platformlar arası uyumluluğu sayesinde tüm major işletim sistemlerinde sorunsuz çalışır.

Güvenlik açığı tespiti doğruluğu nedir?

Platformumuz, yaygın zafiyet türlerinde (XSS, SQLi) %95+ doğruluk oranı sağlar. False positive oranı %5'in altındadır. Her bulgu için kanıt ve payload bilgisi sunulur.

False positive'leri nasıl azaltıyorsunuz?

Çift doğrulama mekanizmaları, response analizi ve pattern matching kullanıyoruz. Yakında AI destekli anomali tespiti ile false positive oranını daha da düşürmeyi hedefliyoruz.

Raporlar nasıl özelleştirilebilir?

JSON çıktısı, Python scriptleri ve Jinja2 template'leri ile özelleştirilebilir. Filtreleme, format dönüştürme (HTML, PDF, Markdown) ve CI/CD entegrasyonu için örnek scriptler sağlıyoruz.

Manuel müdahale gerektiren durumlar?

Otomasyon, tekrar eden görevler için tasarlanmıştır. Karmaşık business logic hataları, oturum yönetimi ve çok adımlı exploit senaryoları için manuel kontrol önerilir. Raporlar bu durumları işaretler.

Performans ayarlamaları nasıl yapılır?

Her modül için yapılandırma dosyaları (YAML/JSON) kullanılır. Thread sayısı, timeout değerleri, concurrency limitleri ve rate limiting ayarları kolayca özelleştirilebilir. CLI parametreleri ile hızlı ayar yapılabilir.

OPEN SOURCE COMMUNITY

İletişim ve Katkıda Bulunma

Güvenlik topluluğu için, topluluk tarafından geliştirilen bir araç

GitHub Deposu

Projeye katkıda bulunmak için GitHub deposunu ziyaret edin. Pull request'ler, issue'lar ve feature istekleri memnuniyetle karşılanır.

github.com/azizefecirak/LAM-Action-Guard

Fork | Star | Issue

Geliştirici

Aziz Efe Çırak

Cybersecurity Researcher

Siber güvenlik araştırmacısı ve açık kaynak yazılım geliştiricisi. Penetration testing, otomasyon ve yapay zeka üzerine çalışmalar yapıyor.

azizefecirak@example.com

linkedin.com/in/azizefecirak

@azizefecirak

Nasıl Katkı Sağlayabilirsiniz?

Kod Katkısı

Yeni modüller, bug fix'leri ve performans iyileştirmeleri

Dokümantasyon

README, wiki ve kullanım kılavuzları

Test & Bug

Test geri bildirimleri ve bug raporları