

CLASSIFICATION: TECHNICAL WHITEPAPER

Log Sensitivity Analyzer (LSA)

Yüksek Performanslı Adli Denetim ve Veri
Sızıntısı Önleme (DLP) Çerçeve

Compliance: KVKK Madde 12 & GDPR Recital 49

Mikroservis mimarileri ve dağıtık sistemler için 'Data in Motion' analiz çözümü.

Kaotik Log Akışlarında Adli Düzen: Yönetici Özeti

Modern dağıtık sistemlerde (Microservices) loglar, yanlışlıkla sızan verilerle doludur. LSA, bu kaosu denetleyen, **PII** (Kişisel Veri) ve **Secret** (Şifre TokenName) tespiti yapan bir güvenlik katmanıdır.

Compliance-First

KVKK Madde 12 ve GDPR Recital 49 uyumluluğu için 'Privacy by Design' yaklaşımı.

JSON-Centric

Maksimum otomasyon ve SIEM entegrasyonu için tüm çıktılar ve konfigürasyonlar JSON tabanlıdır.

Unix-Native

Yüksek işlem hacmi için File Descriptors (FD 0/1/2) ve Pipe mimarisi kullanımı.

Beklenen Özellikler

- PII Tespiti:** Log satırlarında regex ile kredi kartı, e-posta, telefon numarası gibi kişisel verileri yakalama.
- Secret Scanning:** Loglara yanlışlıkla basılan API anahtarlarını ve şifreleri bulma.
- Tehlike Analizi:** Hangi log dosyasının ne kadar risk taşıdığını raporlama.

Veri Güvenliği Bir Tercih Değil, Yasal Bir Emirdir

KVKK Madde 12 (Türkiye)

- **Zorunluluk:** Veri sorumlusu, "her türlü teknik ve idari tedbiri" almak zorundadır.
- **Risk:** 1.000.000 TL'ye varan idari para cezası ve 4 yıla varan hapis.
- **Yükümlülük:** İhlal durumunda "En kısa sürede" bildirim (Kurul Kararı 2019/10).

GDPR Recital 49 (AB)

- **Şeffaflık:** Veri işleme süreçlerinde "Accountability" (Hesap Verilebilirlik).
- **Bildirim:** İhlal durumunda 72 saat içinde bildirim şartı.

LSA, PII tespitini ve maskelenmesini sağlayarak yasal 'Due Diligence' (Gerekli Özən) yükümlülüğünü yerine getirir.

Mimari Karar: Neden Batch Değil, Stream Processing?

Batch Processing (Eski Yöntem)



Latency: Dakikalar/Saatler

Risk: Sızıntı diske yazıldı (KVKK Saklama İhlali)

Karmaşıklık: Veri büyütükçe artan yük

Stream Processing (LSA)



$O(1)$

Karmaşıklık: $O(1)$ Hafıza (Sliding Window)

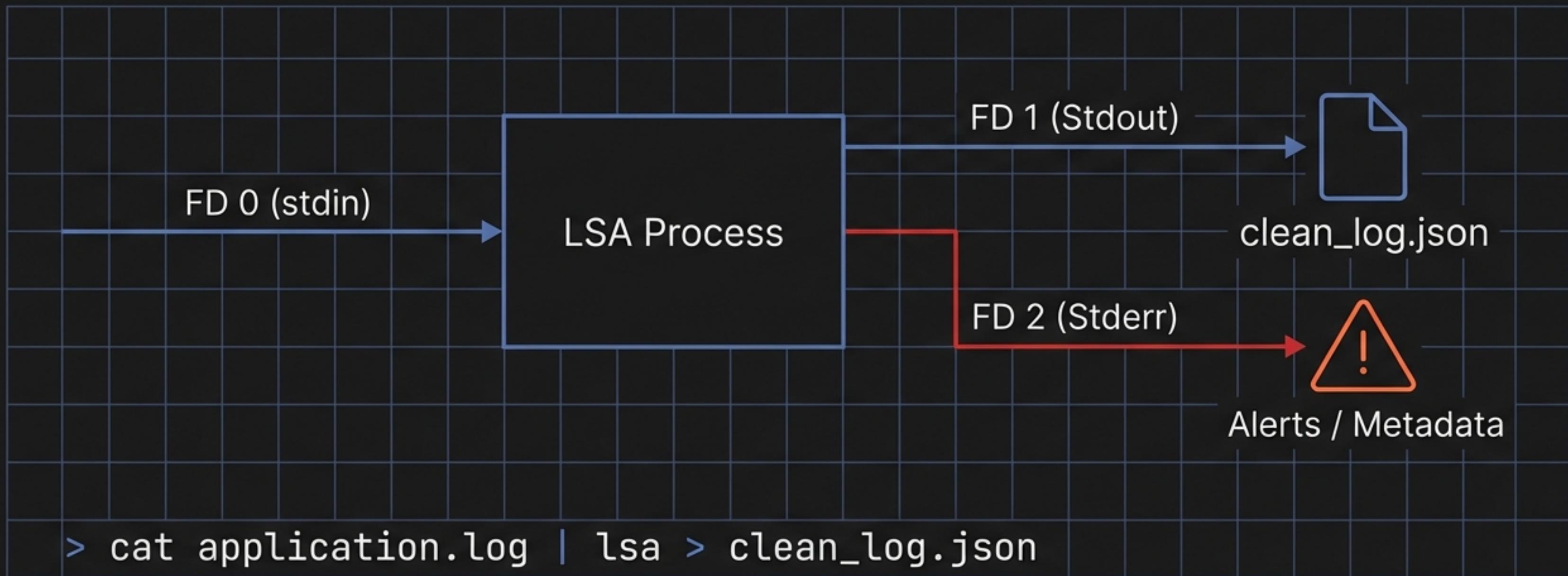
Anlık Müdahale: Veri diske yazılmadan yakalanır

Durum: Data in Motion

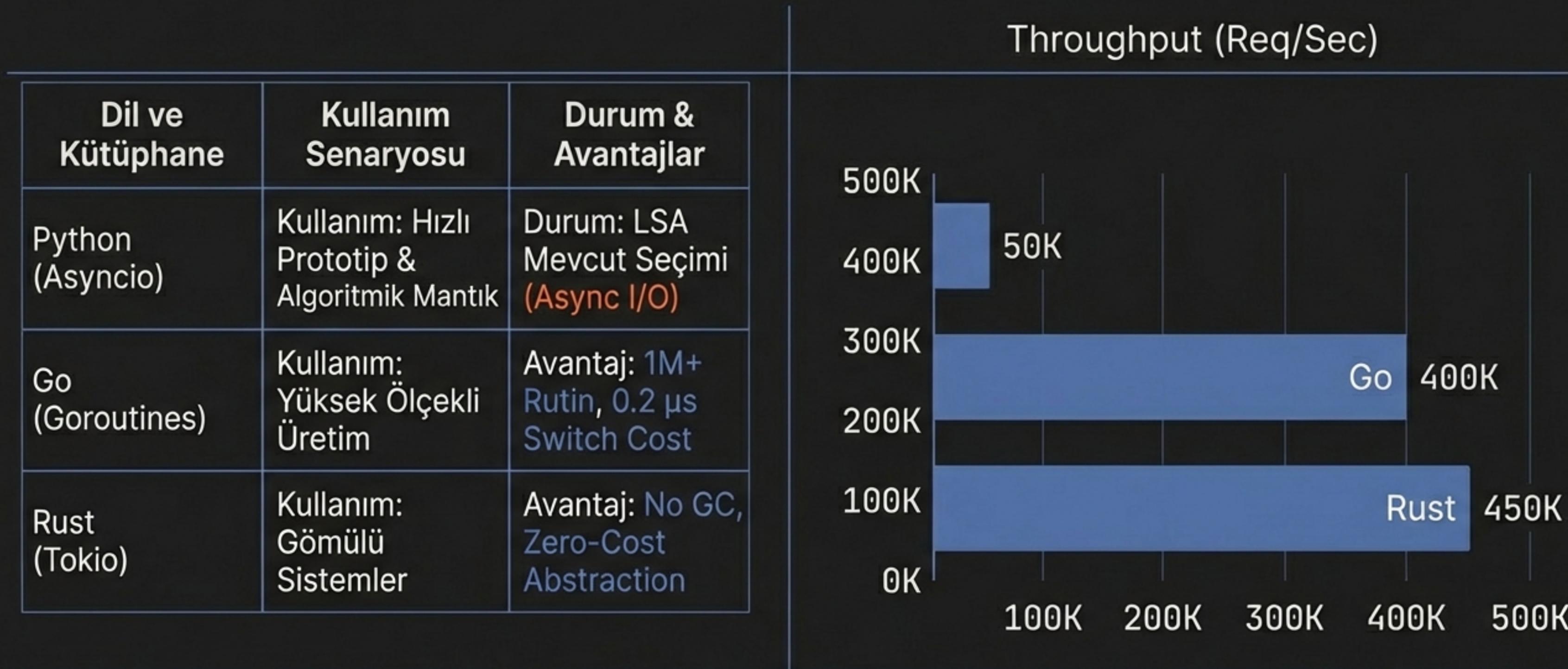
Sızıntı diske temas etmeden engellenir, risk penceresi milisaniyelere indirilir.

Unix I/O Mimarisi ve Terminal Otomasyonu

Non-Logging Audit Paradox: Denetim aracı veri sızdırmamalıdır.

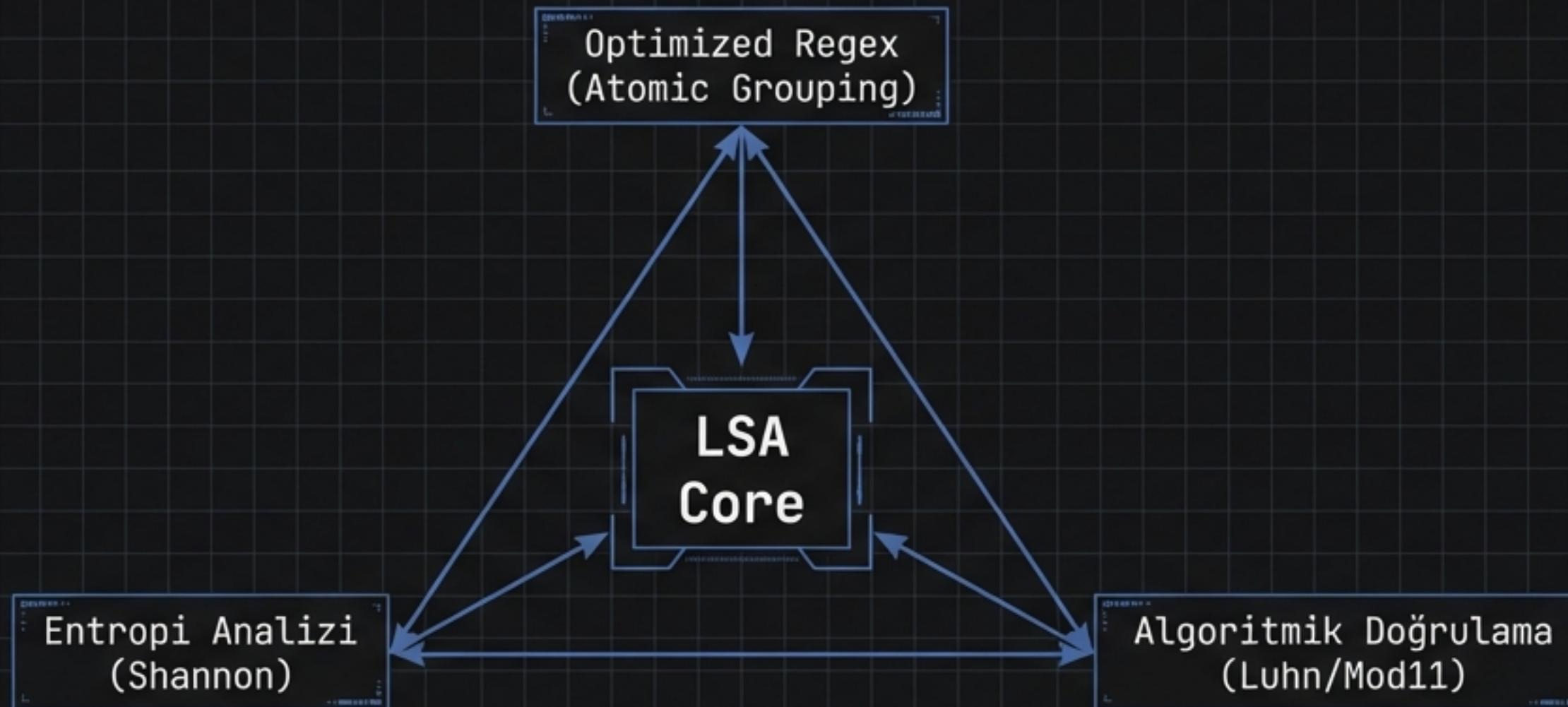


Eşzamanlılık (Concurrency) ve Performans Analizi



LSA, zengin kütüphane desteği için Python'u, I/O darboğazlarını aşmak için asenkron mimariyi kullanır.

Tespit Metodolojisi: Hibrit Yaklaşım



Karşılaştırma Tablosu:

- **GitLeaks:** Sadece Regex (Yüksek False Positive).
- **TruffLeHog:** Entropi Odaklı (TCKN Desteği Zayıf).
- **LSA:** Hibrit + Yerel Mevzuat (KVKK) Desteği.



Rastgele 11 haneli bir sayı TC Kimlik değildir. Matematiksel doğrulama şarttır.

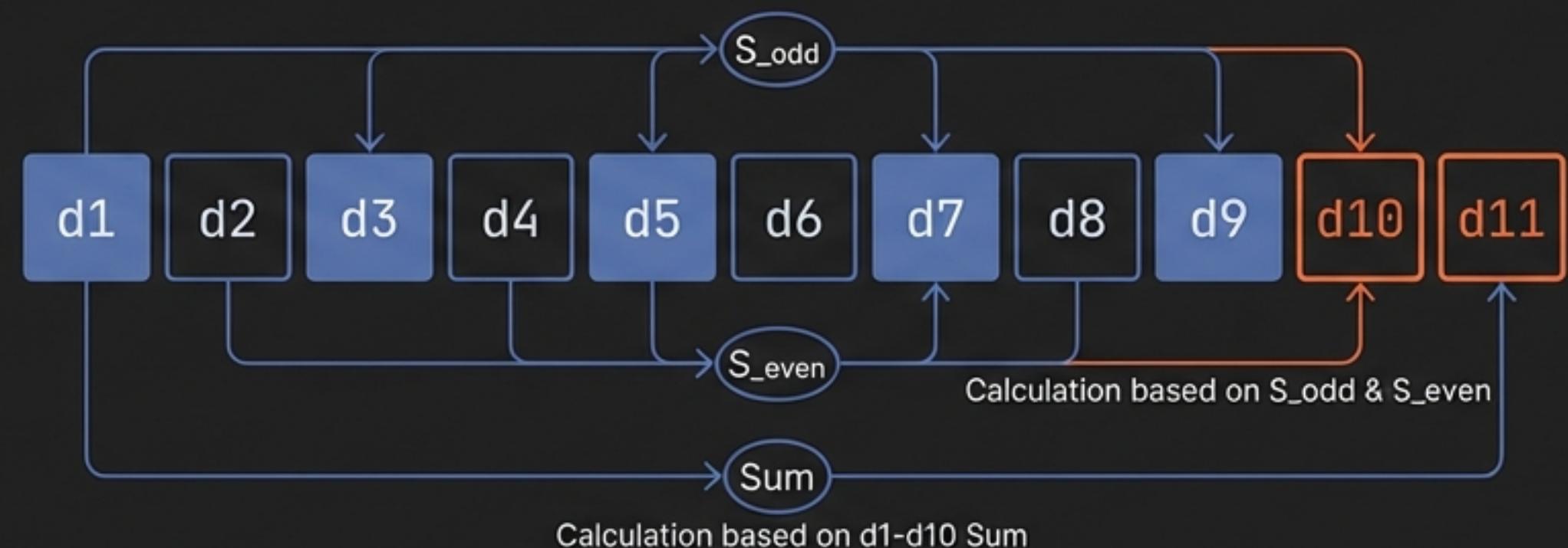
Algoritmik Doğrulama 1: TC Kimlik (Modulo 11)

Doğrulama Kuralları

- Uzunluk:** 11 Hane
- Kural:** İlk hane $\neq 0$
- Doğrulama:** 10. ve 11. hane hesaplanabilir olmalıdır.

$$d_{10} = (7 \cdot S_{odd} - S_{even}) \{ \text{mod } 10 \}$$

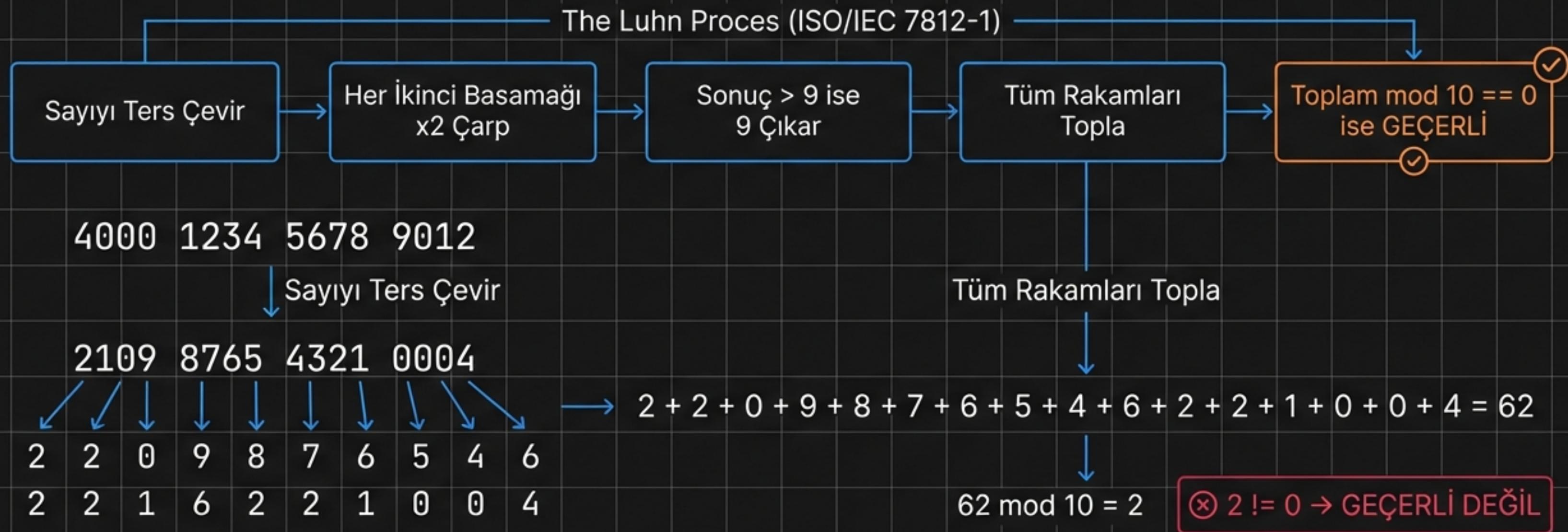
$$d_{11} = \left(\sum_{i=1}^{10} d_i \right) \{ \text{mod } 10 \}$$



! Bu algoritma, rastgele sayı dizilerini eleyerek yanlış alarmları (False Positives) %99 oranında düşürür.

Algoritmik Doğrulama 2: Kredi Kartı (Luhn Algoritması)

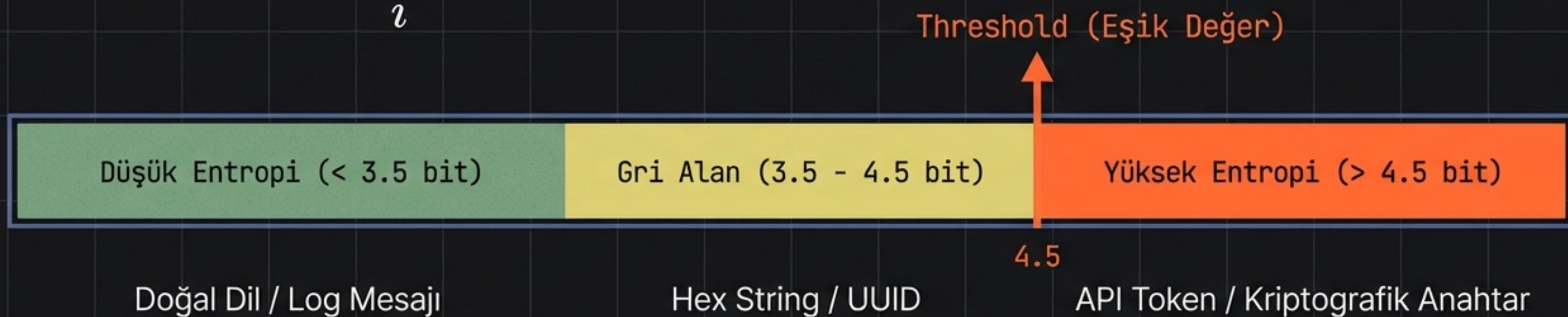
“16 haneli bir zaman damgası (timestamp) kredi kartı değildir.”



! Not: Regex ile tespit edilen 4xxx (Visa) veya 5xxx (Mastercard) desenleri bu matematiksel süzgeçten geçirilir.

Shannon Entropisi ile Secret Scanning

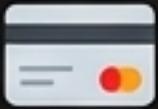
$$H(X) = - \sum_i p_i \log_2(p_i)$$



Kelimeler öngörülebilir, şifreler rastgeledir. LSA bu rastgeleliği ölçer.

Çok Faktörlü Risk Skorlama Çerçevesi

$$\text{Risk Score} = (f_{type} \times f_{exposure} \times f_{density})$$

Veri Tipi (Type)	Bağlam (Context)	Yoğunluk (Density)
TCKN (10 Puan) 		
Kredi Kartı (9 Puan) 	'Password' kelimesi yakınında mı? → (x2.0 Çarpan)	Tekil sizıntı vs. Veritabanı dökümü.
Email (4 Puan) 		

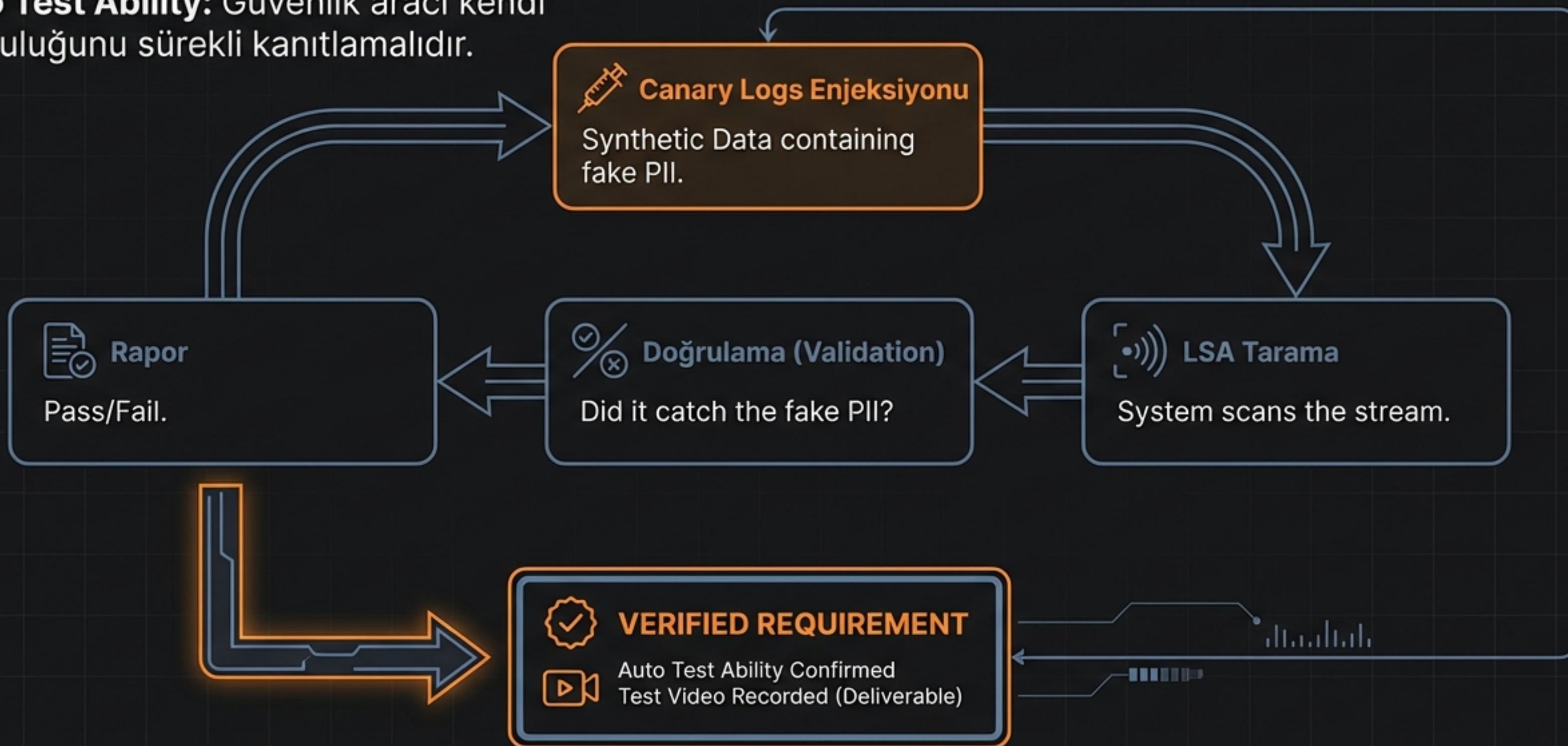
Risk Meter



(100+ → Circuit Breaker)

Otomasyon ve Kendi Kendini Test (Self-Check)

Auto Test Ability: Güvenlik aracı kendi doğruluğunu sürekli kanıtlamalıdır.



Proje Mimarisi ve Standartlar

Klasör Yapısı

└ proje-root/	
└ researchs/	Derin araştırma çıktıları
└ specs/	Gereksinimler ve analiz
└ src/	Kaynak kodlar
└ docs/	Dokümantasyon
└ README.md	Proje özeti

Base Info Format (JSON)

project_info.json

```
{  
  "projectName": "My OS Project",  
  "category": "Service Management",  
  "author": "Student Name",  
  "version": "1.0.0",  
  "description": "Systemd service manager wrapper",  
  "features": [  
    "List services",  
    "Auto-restart crashed services"  
,  
  "requirements": {  
    "os": "Ubuntu 22.04",  
    "language": "Python 3.10"  
  }  
}
```



⚠ Git Workflow: Branching ve Pull Request standartları ile sürdürülebilir gelişim.

UI Standartları ve Raporlama (Streamlit)



Canlı Dashboard

Mobil Uyumlu (Responsive)

Vibrant Colors for Alerting

UI Standard Concept: Based on Module 6: Web Dashboard specifications, emphasizing modern, responsive design and functional, vibrant color schemes for critical data visualization.

Teknik Özeti ve Teslimat

Final Kontrol Listesi

- Yasal:** KVKK Md. 12 & GDPR tam uyumluluk sağlandı.
- Matematiksel:** Luhn, Mod10 ve Shannon Entropisi entegre edildi.
- Mimari:** Stream Processing ile O(1) bellek performansı.
- Güvenlik:** Mlock (RAM güvenliği) ve Read-Only dosya sistemi.



GitHub Reposu ve Dokümantasyonu İnceleyin